# Government MCA College

## Maninagar (East), Ahmedabad

## VISION:

Provide value-based quality education for computer science applications which enable students to solve real-life problems of society.

## MISSION:

- To equip our students with good knowledge, skills and attitude to solve real-life problems in the domain of computer applications.
- To establish industry-academia interaction to facilitate the students to work proficiently in the industrial environment.
- To imbibe high moral values and professional ethics.
- To provide a conducive environment so as to achieve excellence in teaching-learning, and research and development activities.

# Government MCA College

## Maninagar (East), Ahmedabad

## Program Objectives (POs):

- PO1 (Computational Knowledge): Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
- PO2 (Problem analysis): Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
- PO3 (Design/development of solutions): Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- PO4 (Conduct investigations of complex problems): Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5 (Modern tool usage): Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
- PO6 (Professional Ethics): Understand and commit to professional

**ethics and cyber regulations, responsibilities, and norms of professional computing practices.**

- **PO7 (Life-long Learning): Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.**
- **PO8 (Project management and finance): Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.**
- **PO9 (Communication Efficacy): Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.**
- **PO10 (Societal and Environmental Concern): Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practices.**
- **PO11 (Individual and Team Work): Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.**
- **PO12 (Innovation and Entrepreneurship): Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.**

# Government MCA College

## Maninagar (East), Ahmedabad

## Program Educational Objectives (PEOs):

- Identify and analyze real-life problems and design computing systems appropriate to their solutions that are technically sound, economically feasible, and socially acceptable.
- Exhibit professionalism, ethical attitude, good communication skills, and teamwork in their profession.
- Adapt to current trends by engaging in life-long learning.

## Program Specific Outcomes (PSOs):

- Meet the needs of users within an organizational and societal context through the selection, creation, application, integration, and administration of computing technologies.
- Apply concepts, probability, statistics, and mathematics through calculus (differential and integral), numerical methods, and sciences, including applications appropriate to the field of computing problems.
- Use algorithms, data structures, database management, software design, concepts of programming languages, and computer organization and architecture in computer applications.

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION

## Year – I (Semester – I) (W.E.F. JULY 2017)

**Subject Name: Fundamentals of Programming – 1**
**Subject Code: 3610001**

**1. Objectives:**

- To learn about the data types, operators and functions in C programming language.
- To be able to write code in C programming language for simple problems

**2. Prerequisites:** Basic Mathematics and knowledge about number systems

**3. Course Contents:**

| Sr. No. | Course Content | No. of Sessions |
|---|---|---|
| 1 | **Unit 1: Introduction to C**<br><br>Structure of a C Program, First C Program, Files used in a C Program, Compiling and executing C Program, Compiling and executing C Programs, Using comments, keywords, identifiers, Basic data types in C, Variables, Constants, Input/OutputStatement in C, Operators in C, Programming examples, Type conversion and Typecasting. | 07 |
| 2 | **Unit 2: Decision Control and Looping Statements**<br><br>Introduction to Decision Control Statements, Conditional branching statements, Iterative Statements, Nested Loops, break and continue statements, goto statement | 07 |
| 3 | **Unit 3: Functions**<br><br>Introduction, Using Functions, Function Declaration/Function Prototype, Function Definition, Function call, return statement, Passing Parameters to the function, scope of variables, Storage classes, Recursive Functions, Types of recursions, Tower of Hanoi, Recursion versus Iteration | 10 |
| 4 | **Unit 4: Arrays**<br><br>Introduction, Declaration of arrays, Accessing elements of the Array, Storing values in Arrays, Calculating the length of the array, Operations that can be performed on Arrays, ***Introduction of Pointers***, One-dimensional arrays for inter-function communication, two-dimensional arrays, Operations on two-dimensional arrays, Passing two-dimensional arrays to functions, multidimensional arrays, Sparse matrices , Applications of Arrays | 10 |

| | | |
|---|---|---|
| **5** | **Unit 5: Strings**<br><br>Introduction, Supressing input, String taxonomy, Operations on Strings, Miscellaneous String and Character functions, Array of Strings | **6** |

**4. Text Book(s):**

1. Reema Thareja. "Programming in C",2nd Edition, Oxford University Press

**5. Other Reference Books:**

1. Programming in C, by Pradip Dey & Manas Ghosh, Publisher – Oxford
2. Programming in ANSI C, by Balagurusamy, Publisher - Tata McGraw Hill.
3. Programming with ANSI and Turbo C, by Ashok N Kamthane, Publisher – Pearson Education.

**6. Unit wise coverage from Text book(s):**

| Unit 1 | Topics |
|---|---|
| **I** | Chapter 2 |
| **II** | Chapter 3 |
| **III** | Chapter 4 |
| **IV** | Chapter 5 |
| **V** | Chapter 6 |

**7. Accomplishments of the student after completing the course:**

After completion of the course students should become capable of solving problems using computers through C programming language.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
## Year – I (Semester – I) (W.E.F. JULY 2017)

## Subject Name: Fundamentals of Programming – I

## Subject Name: 3610001

### Practical List

| | |
|---|---|
| 1 | Write a Program to Print "Hello World!" |
| 2 | Write a program to print Name , Address and Birth Date. |
| 3 | Write a program to add, multiply and divide two integers and float numbers. |
| 4 | Write a program to convert Rupees(float) to paisa(int). |
| 5 | Write a Program to swap two numbers using temporary variable. |
| 6 | Write a Program to swap two numbers without temporary variable. |
| 7 | Write a Program to check whether given Number is an odd or Even Number |
| 8 | Write a Program tofind maximum number from user entered 3 numbers. |
| 9 | Write a Program to Implement Simple Calculator operation on given two input integers. |
| 10 | Write a Program to check whether given Number is a Prime Number or not? |
| 11 | Write a Program to check whether given Number is a Perfect Number or Not? |
| 12 | Write a Program to implement Number system conversion<br>Create Menu<br>1 Decimal to Binary<br>2 Decimal to Octal<br>3 Decimal to Hexa<br>4 Exit |
| 13 | Write aprogram to convert octal to binarynumber. |
| 14 | Write a Program to display series 1 4 9 16 …n ( using all looping controls) |
| 15 | Write a Program to display multiplication table for given number. |
| 16 | Write a Program tofind first N Fibonacci numbers. |
| 17 | Write a Program tofind Factorial for given Number |
| 18 | Write a Program toconvertfromFahrenheittocentigrade.(Theformulafor conversion: C= (5/9) * (F -32)) |
| 19 | Writeaprogramtoprintthedigitpyramid(reverse) ofnlinesasshownbelow.<br>Exampleis forn=5.<br>123454321<br>1234321<br>12321<br>121<br>1 |
| 20 | Write a program to find sum of first N odd numbers |
| 21 | Write a program to find first N prime numbers |
| 22 | Write aprogram to perform addition and subtraction on mXn matrix. |
| 23 | Writeaprogramto.Writeafunctionthatcalculatesthesumandaverageofthe arraysent as anargument. |

| | |
|---|---|
| 24 | Write a program to add two matrices. |
| 25 | Write a program to find string length |
| 26 | Writeaprogramthatsortthearraysentasanargument.Providemenutosort arrayinascendingand descending. |
| 27 | Write a program thatreadsandmergetwosortedarraystogiveasinglesorted array. |
| 28 | Write a programthattakesfewarrayelementsfromuserandthendisplaythe menu to find smallest, largest, average and summation of allthe numbers. |
| 29 | Write a programtoupdateanddeleteanelementatdesiredpositioninan array. |
| 30 | Write a program to print size of int, float, double variable |
| 31 | Write a program that will read a text and count all occurrences of a particular Word |
| 32 | Write a program that appends the one string to another string |
| 33 | Writeaprogramtoreverseagiven String(oneword)andfindfrequencyof Characters found in it. |
| 34 | Write a program that finds a given word in a string. |
| 35 | Write a function prime that returns 1 if its argument is a prime no. and returns 0 Otherwise |
| 36 | Anumberisspecialifitisdivisibleby15.Anumberisbigifitisgreaterthan 999.Anumberisweirdifitisdivisibleby5and6butnot18.Anumberis scaryifitisbigorweird.Writeaprogramtocheckwhichofthenumbers450, 540, 600 and 675 arespecial but notscary. |
| 37 | WriteaCprogramtoreadinathreedigitnumber,andproducefollowing output(assumingthat theinputis 347).<br>3 hundreds<br>4 tens<br>7 units |
| 38 | Write a Program tofind sum of digitof 3digitnumber. |
| 39 | Write a program to reverse given number and check whether it is a Palindrome or not? |
| 40 | Write a program to reverse given String and check whether it is a Palindrome or not? |
| 41 | Write a program to count number of spaces, comma, lines, words and characters in a given string. |
| 42 | Write a program to count number of vowels in a given string. |
| 43 | Writeaprogramtoprintalldigitsofanintegernumberintowords.Usean arrayof10stringstostore10digitsinwords.Ex.Fori/p1985,o/pshouldbe OneNineEight Five. |
| 44 | WriteaprogramthatacceptName,Department,basic,HRA,andconveyance from the userand calculate total salary.Print salaryslip.<br>DA=35%ofBasic<br>Deductions: Medical :Rs.150 , PF: 10% ofBasic+DA,IT : 20% |
| 45 | Write an algorithm toaccept string ( oneword)from user and displaytriangle. e.g.INDIAwilldisplayTriangle as below<br>I<br>IN<br>IND<br>INDI<br>INDIA |

Note: Implement Practical#20 onwards using functions and different types of calling parameters ( if any).

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION

## Year – I (Semester – I) (W.E.F. JULY 2017)

**Subject Name: Fundamental of Web (FoW)**
**Subject Code: 3610002**

**1. Objectives:**

1. Students will learn about the opportunities, challenges and techniques for developing websites built with the new resources provided by HTML5.
2. Students will learn about the evolving principles and standards for constructing accessible websites; will understand different classes of disabilities and the available techniques for rendering websites useful to those with disabilities.

**2. Prerequisites:** Working knowledge of Internet

**3. Course Contents:**

| Unit | Course Content | No. of Sessions |
|---|---|---|
| 1 | **Unit 1: Introduction to Web and HTML 5**<br><br>**History of internet and Web**<br><br>**Key Terminology:** Internet Protocols, The Client-server Model, Domain Name System, Uniform Resource Locator, Hyper Text Transfer protocol, Web Servers<br><br>**Introduction to HTML5;** New structural elements of HTML5 (Building an HTML5 Starter document, Using header Element to create a site Header, Using the hgroup element to group headings, Creating navigation with nav element, Using the new article element, Grouping content with section element, Creating a side bar with the aside element, Using the footer element, Using the HTML5 outliner to ensure the correct structure,<br><br>**Grouping text level and redefined Semantics:** making up figures and Captions with the figure and figcaption elements, Marking up the date and time with the time element, making the Native Toggle Widget with the details element, using the address element for contact information, Highlighting text with mark element, using s element to show inaccurate or irrelevant element., changes to existing elements, wrapping links around elements, Adding semantic information with Microdata | 10 |
| 2 | **Unit 2: Introduction to CSS and Web Forms**<br><br>Creating a Responsive Design with CSS3 media query, Using custom fonts with @font-face, Making buttons with css Gradients and multiple backgrounds, Enhancing a site with transformations and Transitions, creating animations with CSS<br><br>**HTML5 Web Forms:** HTML4 input types, Creating a form to collect contact information, creating a search form with input type=search, creating calendar and time controls, Creating a number picker, Creating a slider ( without javascript), Creating a color picker, Displaying results with output | 10 |

| | | |
|---|---|---|
| | element, Using Form Placeholder Text, creating an autocomplete feature with list and datalist, Tracking the completion of a task with the progress element, measuring with meter element, Jumping ro a form element when the page loads, Allowing multiple entries, Basic validations with required attribute, writing your own validation rules, limiting user input, customizing and styling the form, error messages, | |
| 3 | **Unit 3: Drawing with Canvas**<br><br>Canvas overview, Laying a Grid on canvas, Canvas tools, Drawing polygons with a Path, drawing Arcs and Circles, Canvas transformations, | 6 |
| 4 | **Unit 4: Introduction to JavaScript**<br><br>Overview, JavaScript design principles, Embedding JavaScript into a web page, JavaScript Syntax, JavaScript Objects, DOM, JavaScript Events, Form validations, | 6 |
| 5 | **Unit 5: Web Media**<br><br>**Embedding Video with HTML5:** including video with video element, enabling video for all browsers, creating a video with subtitles and captions, media API, Making your own custom controls<br><br>**Embedding Audio with HTML5:** including audio with audio element, element, enabling audio for all browsers, media API, creating a beat mixer, adding streaming video | 8 |

**4. Text Book(s):**


1. Randy Connolly, Ricardo Hoar, Fundamentals of Web Development, ISBN-978-93-325-7527-1,Pearson
2. HTML5 Developer's cookbook, Chuck Hudson, Tom Leadbetter, ISBN-978-81-317-8690-1, Pearson


**5. Other Reference Books:**


1. Steven Holzner ,HTML Black Book , Dreamtech Press
2. Jacob Seidelin, HTML5 Games, creating Fun with HTML5,CSS3 and WebGL, Wiley
3. Faithe Wempen, "Step by Step HTML 5", South Asian Edition, Microsoft Press and PHI Learning
3. Wendy Willard, "HTML: A Beginner's Guide 5/E", 5th Edition, McGraw Hill
4. HTML Complete Reference by Thomas A. Powell, Publisher Tata McGraw Hill
5. Teach yourself Java Script in 24 by Michael Moncur Publisher: Pearson Education


**6. Unit wise coverage from Text book(s):**

| **Unit 1** | Book# | Topics |
|---|---|---|
| **I** | 1 | Chapter 1 |
| | 2 | Chapter 1,2 |
| **II** | 2 | Chapter 4,5 |
| **III** | 2 | Chapter 6 |
| **IV** | 1 | Chapter 6 |
| **V** | 2 | Chapter 7,8 |

**Suggested Tutorial**

1) **Location Awareness ( Book 2 Chapter 10): Geolocations Overview:** determining and mapping your location with getcurrentposition,, Determining distance with Position, Options, Following a moving location with watchPosition,

2) **Front end framework :** Overview of any front end framework for project like BootStrap

**Case study:**

1. Using all the new elements to Build a news Page
2. Using all the new elements to Build a search results Page
3. Marking up an article page with comments
4. Create a web form using all HTML5 input types
5. Using HTML5 and Bootstrap prepare your website

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION

## Year – I (Semester – I) (W.E.F. JULY 2017)

## Subject Name: Fundamental of Web (FoW)

## Subject Code: 3610002

### Practical List

| | |
|---|---|
| 1 | Write HTML code to display your bio-data using different types of lists and tables |
| 2 | Create a web page with appropriate content and insert an image towards the left hand side of the page when user clicks on the image, it should open another web page |
| 3 | Create a web page showing an ordered list of names of the subjects, with nested list if any subject has been selected it should display the content of each subject |
| 4 | Write the HTML to make it possible for someone clicking the words "About the authors" at the top of the page to skip down to a list of credits at bottom of the page |
| 5 | Suppose your company has three employees and you want to create a company "directory page" listing some information about each of them. Write the HTML for that page and link 1 employee to another employee. |
| 6 | Write a HTML to create a "guestbook" from that asks someone for his/her name, sex, age, email address |
| 7 | Write html to list the names in a frame taking up the left 25% of browser window. If clicking each name brings up a corresponding web page in right 75% of web browser window |
| 8 | Develop an Html application which accepts registration from the user and it should display the details of the products available in the Warehouse<br>(a) Item Number<br>(b) Item Name<br>(c) Total Quantity available<br>(d) Price/unit<br>- Use Form tag to display the registration form<br>- Use Table tag to represent data<br>- Cellspacing and cellpading attributes should be used in table |
| 9 | Develop a Html application which displays the dishes available in a particular restaurant and also mention the rates for each dishes Give a name for your restaurant which is common for all the web pages<br><br>The details of the dishes are given below<br>▪ Category of dishes<br>▪ Chinese<br>▪ Indian<br>▪ Sub Category<br>▪ Starter<br>▪ Main Items<br>▪ Desserts<br>- Use frames to display item available in restaurant and any other extra facilities given in restaurant<br>- Use list tag to display sub categories |
| 10 | Develop an Html application which displays the information of all trains:<br>a. Based on the day (Monday, Tuesday etc) selected<br>b. Train Number c. Train Name<br>d. Departure<br>e. Arrival<br>f. Departure Time g. Arrival Time |

| | |
|---|---|
| | - Use form to display the details<br>- Also display approximate railway chart for a particular zone by using tables. |
| 11 | Develop an HTML application which displays the information similar to W3SCHOOLS. |
| 12 | Develop an HTML application which accepts railway reservation from the user. |
| 13 | Create a web page with appropriate content and insert an image towards the left hand side of the page when user clicks on the image, it should open another web page with enlarged image |
| 14 | Develop a Html application for Library Management which displays the following details<br>▪ Different areas in the library<br>▪ Books available in different areas<br>▪ Total number of books available in the library<br>▪ Journals available<br>▪ National<br>▪ International<br>Use frames to display the contents |
| 15 | Write Html code which gives information of different cities in Gujarat, when user click on any of the cities form left frame, the information about respective cities should<br>appear on right frame<br>a. Use frame tag Split web page with frames taking the left 25% of browser window, If clicking each name corresponding web page in right 75% of web browser window<br>b. Also display the tourist spot of Gujarat state |
| 16 | Write an Html application similar to photo gallery for any city. |
| 17 | Employ Cascading Style Sheet (inline, internal and external) in HTML tags. |
| 18 | Write a program in Java Script which allows certain fields like Name, Age, Gender, Age, Cite, State and Country. Perform certain validations like name should accept only alphabets, Gender should accept only 1 character, Age should be only in numeric between 1 and 100. |
| 19 | Write a program in Java Script which allows certain fields like First Name, Last name, email address, comments. Perform certain validations like first name and last name should not be empty and email should be valid. If user clicks the submit button it should open a new window and contents should be displayed. If reset contents should be cleared. |
| 20 | Write a program in Java Script which contains 3 functions which are invoked on clicking the Red, Blue and green buttons. The function should contain changing the background, foreground to respective color and to display corresponding status messages. |
| 21 | Develop a webpage using java Script which has following fields like Source,<br>Destination; train no, Date and Number of tickets.<br>a. Source and destination should allow only place code in 3 character<br>b. Date should be in the format DD/ MM / YY<br>c. Number of tickets should allow only numeric |
| 22 | Write a program in Java Script that allows user to enter the text. It also allows the user to accept size and font name that has to be applied on the text entered by the user. |
| 23 | Write a program in Java Script which accepts names in a text box, if a button is clicked names should be sorted and added in another one text area. |
| 24 | Write Java Script code to represent Document object |
| 25 | Represent all properties and methods of Location object in Java Script |
| 26 | Write an Html application implementing Website navigation similar to Flipkart.com, amazon.com. |
| 27 | Develop a webpage for student registration for World Yoga Day event. Implement validation using Java script |

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION
## Year – I (Semester – I) (W.E.F. JULY 2017)

### Subject Name: Program Design Techniques (PDT)
### Subject Code: 3610003

**1. Objectives:**

1. To acquire logical reasoning ability along with various processes and techniques to develop logic and algorithms for solving variety of problems
2. To get orientation for writing efficient programs (codes)
3. To learn some approaches for debugging of programs

**2. Prerequisites:** Attitude of Enjoying Logical Reasoning and Thought Process

**3. Course Contents:**

| Sr. No. | Course Content | No. of sessions |
|---|---|---|
| 1 | **Unit 1: Introduction To Computer Problem Solving**<br>Introduction, The Problem-Solving Aspect, Top-Down Design, Implementation of Algorithms, The Efficiency of Algorithms | **04** |
| 2 | **Unit 2: Fundamental Algorithms**<br>Exchanging The Values of Two Variables, Supplementary Problems, Counting, Supplementary Problems, Summation of a Set of Numbers, Supplementary Problems, Factorial Computation, Supplementary Problems, Sine Function Computation, Supplementary Problems, Generation of The Fibonacci Sequence, Supplementary Problems, Reversing The Digits of an Integer, Supplementary Problems, Base Conversion, Supplementary Problems, Character To Number Conversion, Supplementary Problems | **12** |
| 3 | **Unit 3: Factoring**<br>Finding a Square Root of a Number, Supplementary Problems, The Smallest Divisor of an Integer, Supplementary Problems, Greatest Common Divisor of Two Integers, Supplementary Problems, Generating Prime Numbers, Supplementary Problems, Computing Prime Factors of an Integer, Supplementary Problems, Generation of Pseudo-Random Numbers, Supplementary Problems, Raising a Number To a Large Power, Supplementary Problems, Computing The n-th Fibonacci Number, Supplementary Problems | **12** |
| 4 | **Unit 4: Array Techniques**<br>Array Order Reversal, Supplementary Problems, Array Counting Or Histogramming, Supplementary Problems, Finding The Maximum Number in a Set, Supplementary Problems, Removal of Duplicate Numbers From an Ordered Array, Supplementary Problems, Partitioning an Array, Supplementary Problems, Finding The k-th Smallest Element, Supplementary Problems, Longest Monotone Subsequence, Supplementary Problems | **12** |

| | | |
|---|---|---|
| 5 | **Unit-5: Searching And Recursive Algorithms**<br>**Searching:** Sequential Search: (a) Search in an Unordered Array, (b) Search in an Ordered Array; Binary Search, Hash Searching<br>**Recursive Algorithms**: Introduction, Linear Recursion, Binary Recursion, Implementation of Recursion in (a) Factorial, (b) Fibonacci Sequence, (c) Summation of N Terms, (d) Greatest Common Divisor (GCD), (e) Binary Search | **06** |

**4. Text Book(s):**

1. R.G.Dromey "How to Solve it by Computer ", PHI , 1998

**5. Other Reference Books:**

1. Brian W.Kernighan & Dennis Ritchie "C Programming Language", PHI, 1990
2. Jeri R. Hanly and Eliot B. Koffman “Problem Solving and Program Design in C” Pearson Education, VII Edition, 2012
3. Deitel and Deitel "C How to Program ", Addisson Wesley , 2001
4. E.Balagurusamy " Programming in ANSI C " , Tata McGraw Hill, 2004
5. Byron.S.Gottfried "Schaum's Outline of Programming with C", 2nd Edition, 1996

**6. Unit-wise Coverage from Text book(s):**

**NOTE**: Faculty members are expected to generate similar examples on their own in addition to the examples presented in the book. Some algorithms (for example, Sequential Search, and Recursion examples may not be directly available in the book but these are simple ones, which can easily be handled by the faculty members). Secondly, these exercises should be implemented in FOP-1 subject.

| **Unit 1** | **Topics** |
|---|---|
| **I** | Chapter 1 |
| **II** | Chapter 2 |
| **III** | Chapter 3 |
| **IV** | Chapter 4 |
| **V** | Chapter 5: 5.7, 5.8; Chapter 8: Introduction |

**7. Accomplishment**

Students will acquire ability to develop logic and algorithm for solving variety of problems. They will be able to identify portions of the code (program) which can be made more efficient. They will also be able to debug the programs faster.

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION

## Year – I (Semester – I) (W.E.F. JULY 2017)

**Subject Name: Fundamentals of Computer organization (FCO)**
**Subject Code: 3610004**

**1. Objectives:**

- To be able to understand the elements of Computer Organization and Architecture
- To understand the hardware operation of digital computers

**2. Prerequisites:** Basic Mathematics and knowledge about number systems

**3. Course Contents:**

| Sr. No. | Course Content | No. of Sessions |
|---|---|---|
| 1 | **Unit 1: Basic Components of a digital computer.** | 01 |
| 2 | **Unit 2: Basic Working of Peripheral devices**<br><br>**(Circuit Diagrams not necessary)**<br>Key board , Mouse, Display Unit ,Printer, Multimedia Projector, Scanner, ***USB Ports, Network Adapters*** | 04 |
| 3 | **Unit 3: Introduction to Number System**<br><br>Decimal System, ***Two-state*** Devices, Counting in Binary System, Binary Addition and Subtraction, Converting Decimal Number to Binary Negative Numbers, Use of Complements to represent negative numbers ***in binary and*** other number systems, Binary Number Complements Weighted Code, BCD Code, Octal and Hexadecimal Number System | 09 |
| 4 | **Unit 4: Boolean Algebra and Logic Gates**<br><br>Fundamental Concepts of Boolean Algebra, Logic Gates, Logical Multiplication, AND Gate and OR Gate, Complementation and Inverts Evaluation of logical Expression, Evaluation of an Expression containing Parenthesis, Basic Laws of Boolean Algebra, Proof by Perfect Induction Simplification of Expressions, De Morgan's Theorems, Basic Duality of Boolean Algebra, Derivation of a Boolean Algebra, Interconnecting Gates Sum of Products And Product of Sums, Derivation of POS Expression Derivation of 3 input variables expression, NAND Gates and NOR Gates K-Map Method for Simplifying Boolean Expressions, Subcubes and Covering, POS Expression and Don't Care, Design Using NAND Gates Only, Design Using NOR Gates | 09 |
| 5 | Unit 5: **Basic Concepts of Sequential Logic**<br><br>***Overview of Synchronous and Asynchronous circuits***<br>RS Flip Flop, A Basic Shift Register, ***Binary Counter*** | 03 |

| | | |
|---|---|---|
| 6 | **Unit 6: Basic Concepts of Combinational Logic**<br><br>Construction of ALU, Integer Representation, 1 bit Binary Half Adder 1 bit Binary Full Adder, Positive and Negative Number, Addition in 1's Complement System, Addition in 2's Complement System, Shift Operation Logical and Modulo Operations (Circuit Diagrams not necessary), Basic working and application of Multiplexer | 04 |
| 7 | **Unit 7: Introduction to Memory and Storage Devices**<br><br>Random Access Memories, Basic Memory Cell, Static RAM (Circuit Diagrams not necessary), Dynamic RAM (Circuit Diagrams not necessary) ROM, Magnetic Disk Memories | 04 |
| 8 | **Unit 8: Introduction to Buses**<br><br>Interfacing Buses (Circuit Diagrams not necessary), Concepts of Address Bus, Data Bus and Control Bus, Bus Width (Circuit Diagrams not necessary) | 01 |
| 9 | **Unit 9: Introduction to Control Unit**<br><br>Construction of Instruction Word, Instruction Cycle and Execution Cycle organization of Control Registers | 02 |
| 10 | **Unit 10: Basic Concepts of Computer Organization**<br><br>Instruction Word Formats-Number of Addresses, Representation of Instruction and Data, Addressing Techniques, Direct Addressing, Immediate Addressing, Relative Addressing, Indirect Addressing, Indexed Addressing | 06 |
| 11 | **Unit 11: Introduction to Intel 8086 Architecture**<br><br>Introduction, Bus Interface Unit, Execution Unit, Introduction to Instruction Set, Data Addressing Modes, Instruction Format, Working of MOV, ADD, SUB, MUL, DIV,CMP, IMC, DEC, NEG, AND, OR, NOT, XOR, instructions | 07 |

**4. Text Book(s):**

1. A. Digital Computer Fundamentals, Tata McGraw Hill, 6th Edition, Thomas C. Bartee
2. B. Microprocessor 8086 – Architecture, Programming and Interfacing, Prentice Hall India (PHI), Sunil Mathur

**5. Other Reference Books:**

1. Computer System Architecture, PHI/Pearson Education, 3rd Edition, M. Morris Mano

**6. Unit wise coverage from Text book(s):**

| **Unit 1** | **Book#** | **Topics** |
|---|---|---|
| **I** | **1** | Chapter – 1: 1.7 |
| **II** | | **To be covered from Internet/latest books** |

| | | |
|---|---|---|
| **III** | **1** | Chapter – 2: 2.1 to 2.13 |
| **IV** | **1** | Chapter – 3: 3.1 to 3.22 |
| **V** | **1** | Chapter – 4: 4.1, 4.7, 4.8 |
| **VI** | **1** | Chapter – 5: 5.1 to 5.4, 5.6 to 5.8, 5.14, 5.15, 5.19, 5.20 |
| **VII** | **1** | Chapter – 6: 6.1, 6.2, 6.7 to 6.10 |
| **VIII** | **1** | Chapter – 8: 8.2, 8.3 |
| **IX** | **1** | Chapter – 9: 9.1, 9.2 |
| **X** | **1** | Chapter – 10: 10.1 to 10.9 (Except 10.6) |
| **XI** | **2** | Chapter – 2(2.1, 2.2), Chapter – 4(4.1, 4.2.1, 4.3, 4.5) |

## 7. Accomplishments of the student after completing the course:

After completion of the course students will get the knowledge of computer organization and architecture and will know the actual working and organization of digital computer system.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
## Year – I (Semester – I) (W.E.F. JULY 2017)

**Subject Name: Communication Skills (CS)**
**Subject Code: 3610005**

**1. Objectives:**

- Development of verbal and written communication skills.
- Development of skills for interviews, group communication, and effective presentation
- Learning techniques for effective reading, technical writing, etc.
- Learning basics of vocabulary, grammar

**2. Prerequisites:** Willingness to sharpen communication skills

**3. Course Contents:**

| Sr. No. | Course Content | No. of Sessions |
|---|---|---|
| 1 | **Introduction and Basics of Technical Communication**<br><br>Importance of Communication, Basics of Communication, Purpose, Audience, Cross- Cultural Communication, Language, Communicative Skills, Effective Communication, Modes of Communication, Objectives and Characteristics of Technical Communication, Process of Communication, Levels of Communication, Flow of Communication, Communication Networks, Visual Aids in Technical Communication | 05 |
| 2 | **Effective Speaking and Conversation**<br><br>Introduction, Paralinguistic Features, Barriers to Speaking, Types of Speaking, Persuasive Speaking, Public Speaking, Conversations, Telephonic Conversations and Etiquette, Dialogue Writing | 06 |
| 3 | **Effective Presentation Strategies, Interviews and Group Communication**<br><br>Introduction, Planning, Outlining and Structuring, Nuances of Delivery, Controlling Nervousness and Stage Fright, Visual Aids in Presentations, Objectives of Interviews, Types of Interviews, Job Interviews, Media Interviews, Press Conferences, Forms of Group Communication, Use of Body Language, Discussions, Group Discussions, Organizational GD, GD as Part of Selection Process, Meetings, Conferences, Symposia and Seminars, Negotiations | 09 |
| 4 | **Technical Writing, Words, Phrases, and Sentences**<br><br>Introduction, Audience Recognition/Analysis, Language, Elements of Style, Techniques for Good Technical Writing, Referencing and Styling, Right Words and Phrases, Sentences | 04 |
| 5 | **Letters, Memos and Email**<br><br>Introduction, Letter Writing, Business Letters, Cover Letters, Resumes, Memos, Emails | 05 |

| | | |
|---|---|---|
| 6 | **Reports**<br><br>Introduction, Characteristics of a Report, Categories of Reports, Formats, Prewriting, Structure of Reports, Types of Reports, Writing the Report | 03 |
| 7 | **Research Paper and Dissertation**<br><br>Introduction, Characteristics and Components of a Research Paper, Dissertation | 03 |
| 8 | **Introduction to Modern Communication Media**<br><br>Introduction, Technology Based Communication Tools, Positive Impact of Technology-enabled Communication, Negative Impact of Technology-enabled Communication, Selection of Appropriate Technology, Effectiveness in Technology based Communication | 03 |
| 9 | **Vocabulary**<br><br>Introduction, A Brief History of Words, Using the Dictionary and Thesaurus, Changing Words from One Form to Another, Word Formation : Prefixes and Suffixes, Synonyms and Antonyms, Idioms, Confusables, One-Word Substitutes, Homonyms, Homophones, Eponyms, Phrasal Verbs | 05 |
| 10 | **English Grammar**<br><br>Introduction, Nouns, Gerunds, Infinitives, Subject-Verb Agreement, Tenses, Active and Passive Voice, Conditional Sentences, Adjectives and Degrees of Comparison, Adverbs, Conjunctions, Prepositions, Articles | 05 |

**4. Text Book(s):**


1. Meenakshi Raman & Sangeeta Sharma, "Technical Communication – Principles and Practice", 2nd Edition, Oxford University Press, 2011.


**5. Other Reference Books:**


1. Herta A Murphy, Herbert W. Hilderbrandt, Jane P Thomas, "Effective Business Communication" 7th Edition, Tata McGraw Hill Publication
2. Hedwig Lewis, "Body Language", Response Books
3. Ashraf Rizvi, "Effective Technical Communication", TMGH Publication
4. Paul V. Anderson, "Technical Communication – A Reader Centred Approach", 6th Edition, Thomson Publication
5. Huckins Thomas, "Technical Writing and Professional Communication", McGraw Hill Publication
6. Penrose, Rasberry, Myers, "Business Communication for Managers – An Advanced Approach", 5th Edition, Thomson Publication
7. Bovee, Thill, Schatzman, "Business Communication Today" 7th Edition, Pearson Education
8. Andrea J. Rutherfoord, "Basic Communication Skills for Technology", 2nd Edition, Pearson Education
9. Sharon J. Gerson, Steven M. Gerson, "Technical Writing – Process & Product", 5th Edition, Pearson Education
10. Asha Kaul, "Effective Business Communication", Prentice-Hall India Pvt. Ltd.
11. Daniel G. Riordan, Steven E. Pauley, "Technical Report Writing Today", 8th Edition, Indian Adaptation, Biztantra Publication
12. Sunita Mishra, C. Murli Krishna, "Communication Skills for Engineers", Pearson

Education
13. Leena Sen, “Communication Skills”, 2nd Edition, PHI
14. Kenneth W. Davis, “Business Writing and Communication”, TMGH Publication
15. B. N. Basu, “Technical Writing”, Prentice-Hall India Pvt. Ltd.
16. Matthukutty M. Monippally, “Business Communication Strategies”, TMGH Publication
17. Wren & Martin, “High School English Grammar and Composition”on.

**6. Unit wise coverage from Text book(s):**

| **Unit 1** | **Topics** |
|---|---|
| **I** | Chapter 1,3 |
| **II** | Chapter 6,7 |
| **III** | Chapter 8,9,10 |
| **IV** | Chapter 13,14 |
| **V** | Chapter 17 |
| **VI** | Chapter 18 |
| **VII** | Chapter 20 |
| **VIII** | Chapter 23 |
| **IX** | Chapter 24 |
| **X** | Chapter 25 |

**Suggested Assignments for Continuous Evaluation Component:**

- Group Discussion Sessions
- Mock Interviews
- Write their own Resume
- Assignment on Report Writing, Letter Writing and Memo Writing
- Assignment on English Grammar

**7. Accomplishments of the student after completing the course:**

- Gain an insight into the types of communication
- Build good body language and communication skills while making presentations in a classroom, or boardroom.
- Would be better equipped in writing letters, technical reports etc.

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MASTERS IN COMPUTER APPLICATION

## Year – I (Semester – I) (W.E.F. JULY 2017)

**Subject Name: Software project - I**
**Subject Code: 3610006**

**Guidelines:**

**Team: group of Max 2 Person**

This is aimed to apply the learned concepts, procedures and tools to architect or build an application to develop the skill of application development using acquired knowledge. The students should be motivated to develop the model of application nearer to real life applications and present their work during the evaluation of the projects by the examiners.

A working web application on fundamental of Web and HTML 5 may be developed but before developing working application a prototypical model of Input Design and Output Design or Reports may be developed using any designing tools for understating the concept behind it.

**Option 1: Web application must explore the HTML5, CSS, JavaScript, Theme/template and Front end framework (e.g. BootStrap). Use any Database to store information.**

**Web application must be responsive and dynamic.**

**Suggested Web Applications**

1) University Web site
2) Online Book Store
3) eCommerce Web site
4) Online Library System
5) Online Shopping
6) Railway Reservation System

PS: Above list is a suggestive one. You may select any dynamic application.

**Expected Outcome:**

The objective of the Application Development is to make students aware about the industry based process and workings. As a result, working application that meet with the industry standards should be populated.

There will not be any compulsion to prepare a project report for the students but an application and supportive documents should be self-explanatory, so that evaluator may get the detail about the application developed and can evaluate the students as per the evaluation criteria are given in the last part of this annexure.

**Criteria for Evaluation of Applications Developed:**

| | | **Marks** |
|---|---|---|
| Project | **Template ( customized) / Theme** | **10** |
| | Library ( CSS) / 2D, 3D Graphics | **10** |
| | JavaScript Validation / Web storage | **10** |
| | Online Framework / Animation | **10** |
| Presentation | Communication and explanation | **10** |
| Performance | Practical Code changes to individual | **30** |
| | | |
| | | |

**Suggested Book for Game:**

1. Jacob Seidelin, HTML5 Games,creating Fun with HTML5,CSS3 and WebGL , Wiley

# GUJARAT TECHNOLOGICALUNIVERSITY

## MASTERS IN COMPUTERAPPLICATION

## Year –1(Semester–II) (W.E.F. Dec 2017)

**Subject Name: Fundamentals of Programming - II**
**Subject Code: 3620001**

**1. Objectives:**

1. To be able to understand and use pointers in C programs.
2. To be able to create user defined data types in C
3. To be able to write C application which can do input/output on files.

**2. Prerequisites: Basic knowledge of C programming**

**3. Course Contents:**

| Sr. No. | Course Content | No. of Lectures |
|---|---|---|
| 1 | **Unit 1: Pointers**<br><br>Introduction, Understanding Memory Addresses, Address Operator (&), Pointers (Declaring a Pointer, Initializing Pointers, Indirection Operator and Dereferencing, void Pointer, Null Pointer, Use of Pointers), Arrays and Pointers( One-dimensional Arrays and Pointers, Passing an Array to a Function, Differences between Array Name and Pointer), Pointer and String, Pointer Arithmetic (Assignment, Addition or Subtraction on Integer, Subtraction of Pointers, Comparing Pointers), Pointers to Pointers, Array of Pointers, Pointers to an Array, Two-dimensional Arrays and Pointers ( Passing Two-dimensional Array to a Function), Three-dimensional Arrays, Pointers to Functions (Declaration of a Pointer to a Function, Initialization of Function Pointers, Calling a Function using a Function Pointer, Passing a Function to Another Function, How to Return a Function Pointer, Arrays of Function Pointers. | **06** |
| 2 | **Unit 2: Dynamic Memory Allocation & Advanced Pointer Programming**<br><br>Dynamic Memory Allocation (Dynamic Allocation of Arrays, Freeing Memory, Reallocating Memory Blocks, Implementing Multidimensional Arrays using Pointers), Offsetting a Pointer, Memory Leak and Memory Corruption, Pointer and Const Qualifier (Pointer to Constant, Constant Pointers, Constant Parameters) | **06** |
| 3 | **Unit 3 User-defined Data Types and Variables: Structures, Unions, Enumerations, Bit-fields.**<br><br>Structures (Declaring Structures and Structure Variables, Accessing the Members of a Structure, Initialization of Structures,Copying and Comparing Structures, typedef and its Use in Structure Declarations, Nesting of Structures, Arrays of Structures, Initializing Arrays of Structures, Arrays within the Structure, Structures and Pointers, | **08** |

| | | |
|---|---|---|
| | Structures and Functions), Union (Declaring a Union and its Members, Accessing and Initializing Members of a Union, Structure Versus Union, Enumeration Types, Bitfields | |
| 4 | **Unit 4 : Files**<br><br>Files in C (Using Files in C, Declaration of a File Pointer,Opening a File, Closing and Flushing Files) Working with Text Files (Character Input and Output, End of File (EOF), Detecting the End of a File using feof() Function), Working with Binary Files, Direct File Input and Output (Sequential Versus Random File Access), Files of Records (Working with Files of Records) Random Access to Files of Records, Other File Management Functions (Deleting a File, Renaming a File) Low-Level I/O | **08** |
| 5 | **Unit 5: Linked Lists**<br><br>Singly Linked Lists (Insertion of a Node in a Singly Linked List, Deletion of a Node from a Singly Linked List, Sorting a Singly Linked List, Destroying a Singly Linked List, More Complex Operations on Singly Linked Lists),Circular Linked Lists (Appending a Node, Displaying a Circular Linked List, Inserting a Node after a Specified Node, Inserting a Node before a Particular Node, Deleting a Node, Sorting a Circular Linked List), Doubly Linked Lists (Operations on Doubly Linked Lists, Advantages/Disadvantages of Doubly Linked ) Lists, Introduction to Circular Doubly Linked Lists, Applications of Linked Lists (Dynamic Storage Management, Garbage Collection and Compaction), Disadvantages of Linked Lists, Array versus Linked List Revisited | **08** |
| 6 | **Unit 6: Bitwise Operators & Pre-Processors**<br><br>Bitwise Operator (Bitwise AND, Bitwise OR, Bitwise Exclusive-OR, Bitwise NOT, Bitwise Shift Operator), Command-line Arguments, The C Preprocessor (The C Preprocessor Directives, Predefined Identifiers), Type Qualifier (const Qualifier, volatile Qualifier, restrict Qualifier) Variable Length Argument List, Memory Models and Pointers | **04** |

**4. Text Book(s):**

1. Programming in C, 2nd Edition, Pradip Dey, Manas Ghosh , OXFORD

**5. Other Reference Books:**

1. Programming in ANSI C, by Balaguru samy, Publisher - Tata McGraw Hill.
2. Programming with ANSI and Turbo C, by Ashok N Kamthane, Publisher – Pearson Education.
3. Mastering C, by Venugopal & Prasad, Publisher – Tata McGraw Hill.
4. C: The Complete Reference, by Herbert Schildt, Publisher – Tata McGraw Hill.
5. Let us C, by Yashwant Kanitkar, Publisher – BPB Publication

**6. Accomplishment**

After completion of the course students should become reasonably good at problem solving and algorithm development. They would become capable of solving problems using computers through C programming language.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTER OF COMPUTER APPLICATION
## SEMESTER: II

Subject Name: **Fundamentals of Programming - II**
Subject Code: **3620001**

### Practical List

## A. List of Practical Related to FOP-2:

**Part A: Pointers**

1. Write a program to printing addresses of variable, 1D array, 2D array and function.
2. Write a program to swap elements using pointer with the help of user define function.
3. Write a program to perform any sorting method on string using pointer.
4. Write a program to read, print and addition of two Matrices using pointer and user define functions.
5. Write a program find substring of given string using pointer and user define function.
6. Write a program in C to find whether the given string is palindrome or not using pointer and user define function.
7. Write a program in C to concatenate two strings using pointer and user define function.
8. Write a function to calculate the roots of the quadratic equation. The function must use two pointer parameters for sending roots to its calling function and three parameters for coefficients a, b, and c.
9. Write a function using pointers to add two matrices and to return the resultant matrix to the calling function.

**Part B:**

10. Define a structure called cricket that will describe the following information:
    a. Player name
    b. Team name
    c. Batting average
11. Using cricket, declare an array player with 50 elements and wire a program to read the information about all the 50 players and print a team-wise list containing names of player with their batting average.
12. In a program declare following structure member: name, code, age, weight and height. Read all members of the structure for 100 persons and find list of persons with all related data whose weight > 50 and height > 40 and print the same with suitable format and title.
13. Write a function using pointers to multiply two matrices and to return the resultant matrix to the calling function( The size of array should be dynamic).

**Part C: Files:**

14. Write a program to Read and write in to text file (Character by character).
15. Write a program to read n integer number from keyboard and store them into a file All.txt.
    Read All.txt file, separate even and odd numbers and store them into files Even.txt and Odd.txt respectively and display contents of all the three files.
16. Read n number from keyboard, store them in file Unsort.txt.Sort all the contents of Unsort.txt and store into Sort.txt and display contents of both these files.
17. Two files Data1.txt and Data2.txt contains sorted list of integers. Generate third file that holds a single sorted, merged list of these two files using Merge Sort
18. Read a text file which name is given in command line and print the total number of character in each line and total number of lines in a file.
19. Write a program to accept the contents from the user and store it in the file one line at a time and print the contents of the file.

20. Write a program to crate copy command of DOS.
21. Write a program to find the first occurrence of a string in a file and display the line.
22. Write a program to find the number of occurrences of a string in a file and display all the lines, which contains the string.
23. Write a program to edit a given line of the file when the line number is given by the user.
24. Write a program to delete a given line of the file when the line number is given by the user.
25. Write a program to add a line into the file.
26. Write a program to insert a line before another line identified by the line number in the file.
27. Write a program to
    a) count the number of characters in the file.
    b) count the number of words in the file
    . c) count the number of lines in the file.
28. Write a program to merge two files into the third file.
29. Write a program to create a log file. Accept the username and password from the user if they are correct congratulate them otherwise display the message illegal entry.
30. To open one file called DATA stores 10 integer number in it, find maximum number from that and store in another file called MAX and display it on screen.
31. Program for deleting the spaces from the contents of file.

**Part D: Link Lists**

32. Create a linked list using structure having data and *next as members and implement following functionality. 1)AddAtLast() 2) AddAtBegning () 3) Delete() 4) Sort()
33. Create a doubly linked list using structure having data and *next as members and implement following functionality 1)AddAtLast() 2) AddAtBegning () 3) Delete() 4) Sort()
34. Write a program to create linear linked list interactively and print out the list and total number of items in the list.

**Part E: Advanced:**

35. Write a program to read data from keyboard, write it to a file named STUDENT again read the same data from STUDENT file and write it into DATA file. Same data should be displayed on the screen.
36. Write a program to read data from keyboard, write it to a file named STUDENT again read the same data from STUDENT file and write it into DATA file. Same data should be displayed on the screen.
37. Write a program to read a text file “Small.txt”, containing only small letters and copy it to another file “Capital.txt” after Converting all the small letters of “Small.txt” into capital letters. The names of both of the files are passed using command line arguments.
38. Write a program to read a text file “Salary.txt” and generate Salary Slip.
    Structure: EmpNo,Name, Basic,DA,HRA,CONV,MED
39. Write a program to read a text file “Result.txt” and generate Marksheet Slip.
    Structure: RollNo,Name, Mark1,Mark2,Mark3

# GUJARAT TECHNOLOGICALUNIVERSITY

## MASTERS IN COMPUTERAPPLICATION

**Year –1(Semester–II) (W.E.F. Dec 2017)**

## Subject Name: Data Structures (DS)
## Subject Code: 3620002

**Learning Objectives:**

- To develop proficiency in the specification, representation, and implementation of Data Types and Data Structures.
- To introduce the concepts of algorithmic paradigms and basic data structures and their applications.
- To implement and compare various searching and sorting techniques.
- To apply appropriate data structures to solve different problems.

**Prerequisites:**

- Proficiency in a programming language
- Specification and implementation of basic operations on stack, queue, tree and graph

**Outcomes:**

- Apply sorting and searching algorithms to small and large data sets.
- Ability to design and implement abstract data types such as linked list, stack, queue, graphs and trees.

**Contents:**

| Unit No. | Title | Number of Lectures |
|---|---|---|
| I | **Introduction to Data Structure and Algorithm Analysis:**<br>Data Structure Definition and classification, Storage Representation of Strings, Text Handling and KWIC Indexing. | 4 |
| II | **Linear Data Structures:**<br>Arrays, Storage Structure for Arrays,<br>Stack : List Implementation, Applications of Stacks : Function Call, Recursion, Balancing Symbols<br>Queue: List Implementation, Circular Queue, Priority Queue, double ended queue.<br>Linked List : Cursor Implementation, Multi List<br>Applications of Linked List : Addition and Multiplication of Polynomial in one and two variables | 8 |

| | | |
|---|---|---|
| **III** | **Nonlinear Data Structures:**<br>**Graphs:**<br>Introduction, Definition, Matrix Representation of Graphs, List Structures, Directed/Undirected Graphs, Weighted/Unweighted Graphs Path, Paths of different lengths, Cycle, Cylic Graphs, Acylic Graphs, Spanning Trees, Shortest Path.<br><br>**Trees:**<br>Introduction, Definition, Basic Tree Concepts, , Storage Representation of Binary Trees, Operations on Binary Trees, Tree Traversal, Conversion of General Tree to Binary Trees, Sequential & Other Representation of Trees, Application of Trees – The Manipulation of Arithmetic Expression, Multi-linked Structures - Sparse Matrices. | **14** |
| **IV** | **Sorting and Searching Techniques:**<br>Introduction, Definition, Sorting – Notation and Concepts, Selection Sort, Bubble Sort, Merge Sort, Heap Sort, Quick Sort, Radix Sort, Searching - Sequential Searching, Binary Searching, Search Trees – Height Balanced, 2-3 Trees, Weight Balanced Tree, Trie Structures, Hash Table Search Methods, Hashing Functions, Collision Resolution Techniques. | **14** |

## Text Books:


1. "An Introduction to Data Structures with Applications", Jean-Paul Tremblay, Paul G. Sorenson, Tata McGraw-Hill, 2nd Edition, (2007)
2. "Data Structures and Algorithm Analysis in C", Second Edition, Mark Allen Weiss, Pearson Education (2002)


**Chapter Wise Coverage from Text Book:**

| Unit No. | Text Books | Topics/Subtopics | No. of Lectures |
|---|---|---|---|
| I | Book-1 | 0-3.0 to 0-3.5, 2.4, 2.5.3 | 4 |
| II | Book-1<br>Book-2 | 3.2, 3.5,3.6 to 3.8,4.3.1<br>3.3.3,3.2.7,3.2.8 | 8 |
| III | Book-1 | 5.1.1 to 5.1.5, 5.2.1, 5.3.1, 5.4.1 to 5.4.6 | 14 |
| IV | Book-1 | 6.1.1, 6.1.2, 6.1.3, 6.1.4, 6.1.5, 6.1.6, 6.1.7, 6.2.1, 6.2.2, 6.2.3, 6.2.3.1, 6.2.3.2, 6.2.3.3, 6.2.3.4, 6.2.4, 6.2.4.1, 6.2.4.2, 6.2.4.3 | 14 |
| | | **Total Number of Lectures** | 40 |

## Reference Books:


1. “Introduction to Data Structures in C”, Ashok N. Kamthane, Pearson Education (2004).
2. "Introduction to Algorithm", Cormen, Leiserson, Rivest, Stein, 2nd Edition, PHI (2003).
3. “Design and Analysis of Algorithms”, Parag H Dave, Himanshu B Dave, Pearson (2014)
4. “Data Structures Using C”, Samir Kumar Bandyopadhyay, Kashi Nath Dey, Pearson Education, Year: 2004.
5. “Data Structures and Algorithms”, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Pearson Education (2002).
6. “Fundamentals of Data Structures in C”, Horowitz, Sahni, Anderson-Freed, University Press (2nd edition-2007)
7. "Data Structures and Algorithms,Concepts, Techniques and Applications", G. A.V.PAI, , TMH , 1st Edition (2008).

# GUJARAT TECHNOLOGICALUNIVERSITY

## MASTERS IN COMPUTERAPPLICATIONS

## Semester–II

**Subject Name: Data Structures (DS)**
**Subject Code: 3620002**

## Practical List

### C programming language to performs followings :

**1.** Create a Structure with following Data Members:
1. Integer Array
2. Size of the Array

Sort the Array using various Sorting algorithms such as (i) Selection Sort (ii) Bubble Sort (iii) Two-way Merge Sort (iv) Insertion Sort (v) Quick Sort (vi) Radix Sort (vii) Heap Sort and (viii) Shell Sort. And store the sorted Array in a text file.

**2.** Create a Structure with following Data Members:
1. Integer Array
2. Size of the Array

Search an element in Array using Linear (Sequential) Search and Binary Search. And Display result in file. For Sequential Search, assume (a) Unordered Array, and (b) Ordered Array and develop programs accordingly.

**3.** Create a structure with following Data members:
1. Integer Array
2. Size of the Array

Perform the following operations on stack using user-defined functions:
1. Push
2. Pop
3. Isempty
4. Isfull
5. Peep

Create a file which stores all values of Array through Stack. Has it reversed the order of the elements of the Array? Why?

**4.** Create a user-defined structure with the following data members:
1. A Data
2. A link to the next node

Perform the following operations on stack using user-defined functions:
1. Push
2. Pop
3. Isempty

4. Isfull
5. Peep
Create a file which stores all values of list.

**5.** Write a program to convert an infix arithmetic expression (parenthesize / unparenthesized) into postfix notation.

**6.** Write a program to evaluate a postfix expression.

**7.** Create a structure with following Data members:
1. Integer Array
2. Size of the Array
Search an element in a given list using Binary search by recursion. And Display result in file.

**8.** Create a structure with following Data members:
1. Integer Array
2. Size of the Array
Perform the following operations on Simple queue using user-defined functions:
1. Insert an element
2. Remove an element
3. Display
4. Isfull
5. Isempty
Create a file which stores all values of Array.

**9.** Create a user-defined structure with the following data members:
1) A Data
2) A link to the next node
Perform the following operations on Simple queue using user-defined functions:
1. Insert an element
2. Remove an element
3. Display
4. Isfull
5. Isempty
Create a file which stores all values of list.

**10.** Create a structure with following Data members:
1. Integer Array
2. Size of the Array
Perform the following operations on Circular queue using user-defined functions:
1. Insert an element
2. Remove an element
3. Display
4. Isfull
5. Isempty
Create a file which stores all values of Array.

**11.** Create a user-defined structure with the following data members:
1. A Data

2. A link to the next node
Perform the following operations on Circular queue using user-defined functions:
1. Insert an element
2. Remove an element
3. Display
4. Isfull
5. Isempty
Create a file which stores all values of list.

**12.** Create a user-defined structure with the following data members:
1. A Co-efficient
2. A Exponent
3. A link to the next node
Perform the following operations on Singly list using user-defined functions:
1. Create
2. Display
3. Addition
4. Multiplication
Create a file which stores all values of list.

**13.** Create a user-defined structure with the following data members:
1. A Data
2. A link to the next node

Perform the following operations on list using user-defined functions:
1. Create a list
2. Traverse the whole list\
3. Delete first node
4. Delete last node
5. Delete a node before specified data
6. Insert at first position
7. Insert at last position
8. Insert a node before specified data
9. Insert a node at specified position
10. Count
11. Copy
12. Merge two list
13. Reverse
14. Search
15. Sort
Create a file which stores all values of list.

**14.** Create a user-defined structure with the following data members:
1. A Data
2. A link to the next node

Perform the following operations on Circular list using user-defined functions:
1. Create a list

2. Traverse the whole list\
3. Delete first node
4. Delete last node
5. Delete a node before specified data
6. Insert at first position
7. Insert at last position
8. Insert a node before specified data
9. Insert a node at specified position
10. Count
11. Copy
12. Merge two list
13. Reverse
14. Search
15. Sort

Create a file which stores all values of list.

**15.** Create a user-defined structure with the following data members:
1. A Data
2. A link to the next node
3. A link to the previous node

Perform the following operations on the doubly-linked list using user-defined functions:
1. Create a list
2. Traverse the whole list\
3. Delete first node
4. Delete last node
5. Delete a node before specified data
6. Insert at first position
7. Insert at last position
8. Insert a node before specified data
9. Insert a node at specified position
10. Count
11. Copy
12. Merge two list
13. Reverse
14. Search
15. Sort

Create a file which stores all values of list.

**16.** Create a user-defined structure with the following data members:
1. A Data
2. A link to the next node
3. A link to the previous node

Perform the following operations on doubly-linked Circular list using user defined functions:
1. Create a list
2. Traverse the whole list\
3. Delete first node

4. Delete last node
5. Delete a node before specified data
6. Insert at first position
7. Insert at last position
8. Insert a node before specified data
9. Insert a node at specified position
10. Count
11. Copy
12. Merge two list
13. Reverse
14. Search
15. Sort

Create a file which stores all values of list.

**17.** Write a program to represent an undirected graph using the adjacency matrix to implement the graph and your program be menu driven allowing the user the following options:

1. Create graph
2. Insert an edge
3. Print Adjacency Matrix
4. List all vertices that are adjacent to a specified vertex.
5. Print out vertices using depth first search
6. Print out vertices using breadth first search
7. Exit program

**18.** Create a user-defined structure with the following data members:

1. A Data
2. A link to the Left child
3. A link to the Right child

Perform the following operations on Binary Search Tree using recursion:

1. Create
2. Traverse (Inorder, Preorder, Postorder)
3. Insert
4. Delete
5. Search

Create a file which stores all values of traversal.

*****

# GUJARAT TECHNOLOGICALUNIVERSITY

## MASTERS IN COMPUTERAPPLICATION

## Year –1(Semester–II) (W.E.F. Dec2017)

## Subject Name: Operating Systems
## Subject Code: 3620003

1. **Learning Objectives:**

   This courseis intended to give studentsbasicconcepts of modern Operating Systems. This will give conceptual insight about how OS design and implementation takes place. Also, it will provide insight about interactions between user application, hardware and OS.

2. **Pre-requisites:**

   - Basic knowledge of computer hardware and software.
   - Knowledge of programming languages like C, C++ etc.

3. **Contents:**

| Unit# | CourseContent | Numberof Lectures |
|---|---|---|
| 1 | **Operating System Overview & Processes**<br><br>**Operating system Overview:** Operating system (OS) objectives and Functions, Evolution of OS, Major Achievements of OS, Developments Leading to Modern OS, Virtual Machines, OS design considerations for multiprocessor and multi-core<br><br>**Process Description and Control:** Process, Process State, Process Description, Process Control and Execution of the OS.<br><br>**Threads:** Process and Threads, Types of threads | 08 |
| 2 | **Concurrency Control and Deadlocks**<br><br>**Concurrency: Mutual Exclusion and Synchronization:** Principles of Concurrency, Mutual Exclusion, Mutual Exclusion: Hardware Support, Semaphores, Monitors, Message Passing, Reader/Writer Problem. | 12 |

| | | |
|---|---|---|
| | **Concurrency: Deadlock and Starvation:** Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, An Integrated Deadlock Strategy, Dining Philosophers Problem | |
| **3** | **Memory**<br><br>**Memory Management:** Memory Management Requirements, Memory Partitioning, Paging, Segmentation.<br><br>**Virtual Memory:** Hardware and Control Structures, Virtual Memory Management | **8** |
| **4** | **Scheduling**<br><br>**Uni-processor Scheduling:** Types of Scheduling, Scheduling, Algorithms, Traditional UNIX Scheduling. | **8** |
| **5** | **Input / Output and File Management**<br><br>**I/O Management and Disk Scheduling:** I/O Devices, Organization of the I/O Function, OS Design Issues, I /O Buffering, Disk Scheduling, RAID, Disk cache,<br><br>File management, i-node Structure. | **6** |

**Suggested Additional Topics for Seminar / Reading:**

1) Case Study: Mobile Operating Systems (Android & iOS)
2) Trends in OS (Virtualization and Cloud)
3) VMware and Virtual Box
4) Distributed Processing, Client/Server Architecture and Clusters
5) Operating System Security

**4. Main Reference Book(s):**

1. StallingW, "OperatingSystems", 7th edition, PrenticeHallIndia.

**5. Additional Reference Book(s):**

1. Andrew S. Tanenbaum, Herbot BOS, "Modern Operating Systems", Pearson, ISBN 978-93-978-9325-7577-6
2. Silberschatz, A., Peter B.Galvin and GregGagne,"OperatingSystem Principles", Wiley-Indian Edition, 8th Ed., 2009
3. Ann McHoes,I M Flynn, "Understanding Operating Systems", 8th Edition, Cengage India Publication
4. Bach M J, "The Design of UNIX Operating System", PrenticeHallIndia, 1993.

**6. Chapter wise Coverage from Main Reference Book:**

| Unit# | Topics |
|---|---|
| **1** | **Chapter1( 1 to 6), Chapter2(1 to 5),Chapter3(1 & 2),** |
| **2** | **Chapter4(1 to 6),Chapter5(1 to 6)** |
| **3** | **Chapter6(1 to 4),Chapter7(1 & 2)** |
| **4** | **Chapter8(1 to 3)** |
| **5** | **Chapter10( 1 to 7), Chapter 11 ( 1 & 2)** |

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTER OF COMPUTER APPLICATION
## SEMESTER: II

Subject Name: **Operating Systems (OS)**
Subject Code: **3620003**

## Practical List

**A. List of Practical Related to Operating Systems:**

**Part I Shell Commands**

- date, ls, who, cal, ps, wc, cat, uname, pwd, mkdir, rmdir, cd, cp, rm, mv, diff, chmod, grep, sed, head, tail, cut, paste, sort, find,awk

**Part II: Single Line Commands**

**1** Write a command to display all lines which begins with "NOKIA" string from an prd.lst file. [ prd.lst file should exist with data ]

**2** Delete first, last and all the blank lines from the specified file.

**3** Searches for a line which does not start with the vowel letter in any specific file

**4** Display all such files from your login which has size >= 50.

**5** Display all the files with read and write permissions throughout your login and save output in file and

**6** Write a command to substitute ,,/" with ,,:" throughout the file with all occurrences of /etc/group

**7** Using awk command Count number of lines found in a file

**8** Using awk command find Sum of fields ( Marks )
File Format roll#,name,mark1,mark2,mark3

**9** Using awk command Remove duplicate lines (similar to uniq) from file.

**10** Using awk command Delete all Blank lines

**11** Using awk command Print Last Line of a file

**12** Using awk command find and replace "Hindustan" or "Bharat" to "INDIA"

**13** Using awk command Print first line of a file

**14** Using awk command search for IND, BHA and HIN

**15** Using awk command search for IND and BHA and HIN (in that order)

**16** Using awk command print only lines of less than 65 characters

**17** Using awk command print section of file between two regular expressions (inclusive)

**18** Using awk command print lines between 5 and 8 the line

## Part III: Shell Scripts

**1** Write a script to compare identically named files in two different directories and if they are same, copy one of them in a third directory.

**2** Write a script to copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories.

**3** Write a script to make following file and directory management operations menu based:

1. Display current directory
2. List directory
3. Make directory
4. Change directory
5. Copy a file
6. Rename a file
7. Delete a file

**4** Write a script which reads a text file and output the following

1. Count of character, words and lines.
2. File in reverse.
3. Frequency of particular word in the file
4. Lower case letter in place of upper case letter.

**5** Write A Script To Perform Following String Operations Using Menu:

1. COMPARE TWO STRINGS.
2. JOIN TWO STRINGS.
3. FIND THE LENGTH OF A GIVEN STRING.
4. OCCURRENCE OF CHARACTER AND WORDS
5. REVERSE THE STRING.

**6** Write a shell script to add the statement #include <stdio.h> at the beginning of every C source file in current directory containing printf and fprintf.

**7** Write a script that behaves both in interactive and non-interactive mode. When no arguments are supplied, it picks up each C program from current directory and lists the first 10 lines. It then prompts for deletion of the file. If the user supplies arguments with the script, then it works on those files only.

**8** Write a script that deletes all leading and trailing spaces in all lines in a file. Also remove blank lines from a file. Locate lines containing only printf but not fprintf.

**9** Write a script to display the files
1) Having size more than user entered from given directory.
2) Files not accessed in last Month
3) List zero size files
4) Delete all output files (files with extension lst and out).

## Part IV: Practicals for OS Conceptual clarity (desirable)

| | |
|---|---|
| **1** | Write a C program for implementing concurrency using producer, consumer problem using process or thread under LINUX. |
| **2** | Write a C program for implementing dining philosopher problem under LINUX. |
| **3** | Write a C program for implementing UNIX Processor Scheduling algorithms<br>1) FCFS<br>2) RR<br>3) SPN<br>4) SRT |

# GUJARAT TECHNOLOGICALUNIVERSITY

## MASTERS IN COMPUTERAPPLICATION

## Year –1(Semester–II) (W.E.F. Dec 2017)

## Subject Name: Object-Oriented Unified Modelling
## Subject Code: 3620004

**Learning Objectives:**

- UML is rapidly accepted throughout the software industry for modelling of software requirement and design.
- To understand what the Unified Modeling Language (UML) is, and why it is relevant to the development of software-intensive systems.
- To learn how to apply the UML.
- To learn design patterns and solve problems with the design patterns.
- Recognize and define design and enterprise integration patterns in current common use.

**Prerequisites:**

- There are no formal prerequisites for this course. An exposure to Object-Oriented Programming Language would be helpful, but it is not mandatory.

**Outcomes:**

- Student will be able to do requirements elicitation, requirements analysis, system design and document those in Unified Modeling Language (UML).

**Contents:**

| Unit No. | Title | Number of Lectures |
|---|---|---|
| I | **Basics of UML**<br>Why We Model, Introduction to UML, Classes, Relationships, Common Mechanisms, Diagrams and Class Diagrams. | 10 |
| II | **Advanced Structural Modeling**<br>Advanced Classes, Advanced Relationships, Instances, Object Diagrams. | 8 |
| III | **Basic Behavioral Modeling**<br>Interactions, Use Cases, Use Case Diagrams, Interaction Diagrams, Activity Diagrams. | 10 |
| IV | **Advanced Behavioral Modeling**<br>Events and Signals, State Machines, Statechart Diagrams. | 6 |
| V | **Architectural Modeling**<br>Components, Deployment, Component Diagrams, Deployment Diagrams. | 8 |

## Text Books:

1. "The Unified Modeling Language User Guide", Grady Booch, James Rumbaugh, Ivar Jacobson, ISBN: 9788177583724, Pearson Education

   Also Book Available online:
   https://books.google.co.in/books?id=a5J49FoFKq8C&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

## Reference Books:


1. The Unified Modeling Language User Guide, Booch, Rumbaugh, Jacobson, Addison Wesley, 1999.
2. Object Oriented Modeling and Design, James Rumbaugh, et al, Prentice Hall, 1991.
3. Appying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process, Craig Larman, Prentice-Hall, 2000.
4. The Unified Modeling Language Reference Manual, Second Edition, Rumbaugh, Jacobson and Booch, Addison-Wesley, 2004.
5. UML Distilled: A Brief Guide to the Standard Object Modeling Language , Third Edition, Addison-Wesley Object Technology Series by Martin Fowler .
6. Learning UML 2.0, Russ Miles, Kim Hamilton, O'Reilly Media
7. Visual Modeling with Rational Rose and UML; Terry Quatrani, Addison Wesley, 1998
8. Internet material (e.g., http://www.ambysoft.com/books/agileModeling.html - Agile Modeling Effective Practices for Extreme Programming and the Unified Process)


## Chapter Wise Coverage from Text Book:

| Unit No. | Text Books | Topics/Subtopics | No. of Lectures |
|---|---|---|---|
| I | Book-1 | Chapter 1, Chapter 2, Chapter 3, Chapter 4, Chapter 5, Chapter 6, Chapter 7 and Chapter 8. | 10 |
| II | Book-1 | Chapter 9, Chapter 10, Chapter 13 and Chapter 14 | 8 |
| III | Book-1 | Chapter 15, Chapter 16, Chapter 17, Chapter 18 and Chapter 19. | 10 |
| IV | Book-1 | Chapter 20, Chapter 21, Chapter 24 | 6 |
| V | Book-1 | Chapter 25, Chapter 26, Chapter 29 and Chapter 30 | 8 |
| | | **Total Number of Lectures** | 42 |

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTER OF COMPUTER APPLICATION
## SEMESTER: II

Subject Name: **Object Oriented - Unified Modeling**
Subject Code: **3620004**

## Practical List

**A. List of Practical Related to OOUM:**

1. Consider a Book Shop, Consider a book store in a shopping mall. The customer selects the books from racks to purchase. Prepare a sequence diagram for bookstore checkout system. The customer brings selected books to cashier. The cashier scans each item with checkout system to prepare an order. **Prepare a Use-case and Use-case documentation.**

2. Consider an Online shopping web site, where customer selects the items and adds into cart. At the customer will proceed for payments. Prepare a Use case, Activity and sequence diagram for online shoppingcheckout system. The customer brings selected books to cashier. The cashier scans each item with checkout system to prepare an order. **Prepare a Use-case and Use-case documentation.**

3. The case study 'Online Mobile Recharge' gives us the information about all the mobile service providers. This application provides us the complete information regarding any mobile service provider in terms of their plans, options, benefits, etc. Suppose, any Airtel customer wants to have the information of all the schemes and services provided by the company, he/she can have the information and according to his convenience he can recharge the mobile from the same application. The major advantage of this proposed system is to have the recharging facility of any service provider under same roof. **Prepare a Use-case and Use-case documentation.**

4. In tour management system, System will check whether the customer is existing or new. New user will enter his personal and tour details for reservation. This login information could be used for further transactions. When customer is satisfied with tour package he/she would request for reservation of tour. Personal details of new customer is stored in cust_info while the details regarding the tour selected by particular customer is stored in tour_info and the details regarding it would be restructured in Tour Information System. Existing customer can update his/her personal details in cust_info and cancel reservation for tour from tour_info and changes regarding it are also reflected in Tour Information System. After confirming the tour package the customer will make payment either online or through staff by personally going at the office. Customer can make payment by cash, credit card or by cheque. System checks for the validity of staff. Once the payment is done by customer, valid staff will make Ticket Reservation System. Reserved customer will be able to view details about reservation by providing login information from cust_info and tour_info system. Administrator can add, delete or modify tour schemes from Tour Information System. **Prepare a Use-case and Use-case documentation.**

5. **Prepare a Use-case and Use-case documentation for Railway Reservation System.** Its detail is following:
   "Railway Reservation System is a system used for booking tickets over internet.Any Customer Can book tickets for different trains.Customer can book a ticket only if the tickets are available. Customer searches for the availability of tickets then if the tickets are available he books the tickets by initially filling details in a form. Tickets can be booked in two ways by i-ticket or by e-ticket booking. In case of i-ticket booking customer can book the tickets online and the tickets are couriered to Particular customer at their address. But in case of e-ticket booking and cancelling

tickets are booked and cancelled online sitting at the home and customer himself has to take print of the ticket but in both the cases amount for tickets are deducted from customers account. For cancellation of ticket the customer has to go at reservation office than fill cancellation form and ask the clerk to cancel the ticket than the refund is transferred to customer account. After booking ticket the customer has to checkout by paying fare amount to clerk".

6. **Prepare a Use-case and Use-case documentation for a restaurant bill.** There should be a charge for each delivered item. The total amount should be subject to tax and service charge of 18% for groups of six or more. Any coupons charge submitted by the customer is subtracted from bill. 6. The University runs various diploma, graduation and post-graduation courses such as DE, Dpharm, BE, MBA. MCA, ME, Mpharm etc. The courses follows semester patterns and under each course various subjects are taught. Students seek admissions to these courses and if found elligible, the student is enrolled for the requested course. There are several faculty members in the university who teach the various subjects of these courses. The subject teacher conducts semester examinations for the concerned subject at the end of the semester and the student's performance is recorded. Even if a student is unable to pass a subject, he is promoted to the next semester but has to reappear for the subject examination again and clear his backlog.

7. The University runs various diploma, graduation and post-graduation courses such as DE, Dpharm, BE, MBA. MCA, ME, Mpharm etc. The courses follows semester patterns and under each course various subjects are taught. Students seek admissions to these courses and if found elligible, the student is enrolled for the requested course. There are several faculty members in the university who teach the various subjects of these courses. The subject teacher conducts semester examinations for the concerned subject at the end of the semester and the student's performance is recorded. Even if a student is unable to pass a subject, he is promoted to the next semester but has to reappear for the subject examination again and clear his backlog.
   - Draw Use Case diagram for it
   - Draw Class Diagram for University
   - Draw Activity Diagram for Admission
   - Draw Sequence Diagram for Admission

8. **Prepare a Use-case and Use-case documentation for:**
   a) Facebook Application.
   b) WhatsApp Application.
   c) Identify Actors and Use Cases of Library Management System and draw its Use Case Diagram, Sequence Diagram, Activity Diagram and Class Diagram.
   d) Identify Actors and Use Cases of Online Quiz System and draw its Use Case Diagram, Sequence Diagram, Activity Diagram and Class Diagram.
   e) Identify Actors and Use Cases of ATM System and draw its Use Case Diagram, Sequence Diagram, Activity Diagram and Class Diagram.

**Tools: All the programs mentioned in this list are to be performed in VISIO, Open Source Tools like, Dia etc.**

# GUJARAT TECHNOLOGICALUNIVERSITY

## MASTERS IN COMPUTERAPPLICATION

### Year –1(Semester–II)

Subject Name: **Software Projects - II**
Subject Code: **3620005**

## Preamble

The main purpose of a software project is to enable students to apply their learning to develop software for different applications. This, in turn, will equip them to develop enough knowledge, skills and confidence to solve real-life problems, and, thereby, enhance their chances of getting good employment.
But it will be realized only when faculty members along with students put their sincere efforts. These guidelines are aimed at assisting faculty members and students to achieve the above-stated objectives.

**Project Selection (Statement)**

- Project statement for most of the students will be derived from the concepts either already learnt or being learnt in the current semester. However, there may be some students willing to take up projects in areas, which are out of course curriculum.

- Irrespective of the areas, from where project statements are derived, it should be ensured that the project definition is challenging. However, project scope should be time bound, measurable, and achievable.

- It is recommended that the project list is prepared by the faculty (students may choose a project outside the list based on his interest). It is also recommended that help from Industry may be sought in finding interesting industry problems.

- Project definition can be taken from either course subject, or real-life projects, or research-type of projects, or from any exciting area (e.g. Gaming, etc.)

- **It should be ensured that project definitions are not repeated.**

- **Project should not be copied from earlier batch or from downloaded from Internet. It should be own Developed.**

- It is strongly recommended that faculty guides should take ownership of the student projects being guided by them. This will help the next batch of students (or even same students) to extend and build on the previously done projects.

- The project should be able to test the student's ability to use the technology and the features of the language he/she studied during the semester even if the problem does not belong to the curriculum.

**Project Execution**

- Students should be better be trained and oriented to go through all stages of the software development life cycle (SDLC).
- During initial semesters, students need not follow the SDLC life cycle steps very formally. However, they should be encouraged to follow these steps even in an informal and in a preliminary manner.
- Project statement and scope should be written very clearly along with potential benefits (and the beneficiaries) right from the initial semesters.
- Students should be oriented to follow the concept of algorithmic approach right from the initial semesters.
- Coding standards should be followed meticulously. Clear justification of data structures used and approach taken is appreciated.
- Students should prepare test plan, test data and should go through testing of their software.

**Project Monitoring**

- Projects have a tendency to go through time overrun and cost overrun. Therefore, the project should be monitored frequently, regularly, and closely.
- Monitoring would be possible if the project plan is first prepared with clearly stated mile-stone events.
- Project activities should include SDLC life cycle stages, and the estimated time to complete a milestone activity will be prepared for each SDLC stage. Even if SDLC is not strictly followed, it is advised to record milestone events which represent critical points of the project development process.
- Monitoring would be done with respect to the project plan. Strategy and action-plan should be regularly prepared to cover any shortfall that is observed during monitoring of the project
- It is recommended that the project plan and monitoring is properly documented and the record is maintained.

**Team Size: group of max 3 Person**

**Database: File Based i.e. Data must be stored in File.**

**Expected Outcome:**

- The objective of the Application Development is to make students aware about the industry based process and workings. As a result, working application that meet with the industry standards should be populated.
- There will not be any compulsion to prepare a project report for the students but an application and supportive documents should be self- explanatory,

so that evaluator may get the detail about the application developed and can evaluate the students as per the evaluation criteria are given in the last part of this annexure.

**Project Evaluation**

- Project evaluation will be formally done as per the examination scheme started by GTU.
- Based on periodic monitoring, internal marks will be given as per teaching scheme, based on the quality and quantity of work done at each stage of the SDLC by students.
- Each student team will be required to present the project for 10-15 minutes to external examiners. Examiners may ask for an explanation of the code and other aspects of the project. The recommended team site is 3.
- Additionally, projects can be voluntarily got evaluated by industry professionals to get a good feedback about the suitability of project definitions and utility value to industry.
- Such industry evaluations will improve the status of the students and their institutes in addition to enhancing employability of the students
- GTU External Evaluation Break-up

| | |
|---|---|
| Best Practices :<br>Coding Standard Followed<br>Indentation and Comment<br>Naming Conventions<br>Modular Coding | 25% |
| Code changes / Performance | 60% |
| VIVA | 15% |

.

**Suggested Applications**

1) Library Management System
2) Banking System
3) Railway Reservation System
4) Book Store
5) Admission Registration System (ACPC)
6) Computer Based MCQ Exam System

PS: Above list is a suggestive one. You may select any dynamic application.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTER OF COMPUTER APPLICATION
## SEMESTER: II

Subject Name: **Software Projects - II**
Subject Code: **3620005**

## Practical List

**Preamble**

The main purpose of a software project is to enable students to apply their learning to develop software for different applications. This, in turn, will equip them to develop enough knowledge, skills and confidence to solve real-life problems, and, thereby, enhance their chances of getting good employment.
But it will be realized only when faculty members along with students put their sincere efforts. These guidelines are aimed at assisting faculty members and students to achieve the above-stated objectives.

**Project Selection (Statement)**

- Project statement for most of the students will be derived from the concepts either already learnt or being learnt in the current semester. However, there may be some students willing to take up projects in areas, which are out of course curriculum.

- Irrespective of the areas, from where project statements are derived, it should be ensured that the project definition is challenging. However, project scope should be time bound, measurable, and achievable.

- It is recommended that the project list is prepared by the faculty (students may choose a project outside the list based on his interest). It is also recommended that help from Industry may be sought in finding interesting industry problems.

- Project definition can be taken from either course subject, or real-life projects, or research-type of projects, or from any exciting area (e.g. Gaming, etc.)

- **It should be ensured that project definitions are not repeated.**

- **Project should not be copied from earlier batch or from downloaded from Internet. it should be own Developed.**

- It is strongly recommended that faculty guides should take ownership of the student projects being guided by them. This will help the next batch of students (or even same students) to extend and build on the previously done projects.

- The project should be able to test the student's ability to use the technology and the features of the language he/she studied during the semester even if the problem does not belong to the curriculum.

**Project Execution**

- Students should be better be trained and oriented to go through all stages of the software development life cycle (SDLC).

- During initial semesters, students need not follow the SDLC life cycle steps very formally. However, they should be encouraged to follow these steps even in an informal and in a preliminary manner.

- Project statement and scope should be written very clearly along with potential benefits (and the beneficiaries) right from the initial semesters.

- Students should be oriented to follow the concept of algorithmic approach right from the initial semesters.

- Coding standards should be followed meticulously. Clear justification of data structures used and approach taken is appreciated.

- Students should prepare test plan, test data and should go through testing of their software.

**Project Monitoring**

- Projects have a tendency to go through time overrun and cost overrun. Therefore, the project should be monitored frequently, regularly, and closely.

- Monitoring would be possible if the project plan is first prepared with clearly stated mile-stone events.

- Project activities should include SDLC life cycle stages, and the estimated time to complete a milestone activity will be prepared for each SDLC stage. Even if SDLC is not strictly followed, it is advised to record milestone events which represent critical points of the project development process.

- Monitoring would be done with respect to the project plan. Strategy and action-plan should be regularly prepared to cover any shortfall that is observed during monitoring of the project

- It is recommended that the project plan and monitoring is properly documented and the record is maintained.

**Team Size: group of max 3 Person**

**Database: File Based i.e. Data must be stored in File.**

**Expected Outcome:**

- The objective of the Application Development is to make students aware about the industry based process and workings. As a result, working application that meet with the industry standards should be populated.

- There will not be any compulsion to prepare a project report for the students but an application and supportive documents should be self- explanatory, so that evaluator may get the detail about the application developed and can evaluate the students as per the evaluation criteria are given in the last part of this annexure.

**Project Evaluation**

- Project evaluation will be formally done as per the examination scheme started by GTU.

- Based on periodic monitoring, internal marks will be given as per teaching scheme, based on the quality and quantity of work done at each stage of the SDLC by students.

- Each student team will be required to present the project for 10-15 minutes to external examiners. Examiners may ask for an explanation of the code and other aspects of the project. The recommended team site is 3.

- Additionally, projects can be voluntarily got evaluated by industry professionals to get a good feedback about the suitability of project definitions and utility value to industry.

- Such industry evaluations will improve the status of the students and their institutes in addition to enhancing employability of the students

- GTU External Evaluation Break-up

| | |
|---|---|
| Best Practices :<br>Coding Standard Followed<br>Indentation and Comment<br>Naming Conventions<br>Modular Coding | 25% |
| Code changes / Performance | 60% |
| VIVA | 15% |

**Suggested Applications**

1) Library Management System
2) Banking System
3) Railway Reservation System
4) Book Store
5) Admission Registration System (ACPC)
6) Computer Based MCQ Exam System

PS: Above list is a suggestive one. You may select any dynamic application.

# GUJARAT TECHNOLOGICAL UNIVERSITY

## MCA Sem-2/ B.E. Sem-2 & 7 / B.Pharm. Sem-2 & 7/ Diploma Engineering Sem-2 & 5 (01-07-2013)

Subject Name: **Contributor Personality Development**
Subject Code: **1990001**

**Table-I -For MCA/B.E. / D.E**

| Table-I | | | | | | |
|---|---|---|---|---|---|---|
| Teaching Scheme | | | | *Evaluation Scheme* | | |
| **Theory (Hrs.)** | **Tutorial (Hrs.)** | **Practical (Hrs.)** | **Credit** | ***University Exam (E)*** | ***Mid Sem Exam (Theory) (M)*** | ***Practical (Internal)*** |
| 4 | 0 | 0 | 4 | 70 | 30 | 50 |

***For the Evaluation Scheme of Diploma Engineering Sem – 2, please refer the link**
**http://www.gtu.ac.in/Syllabus/New_Diploma/sem-2/Pdf/3990001.pdf**

**Table-I –For B.Pharm only**

| Table-II | | | | | | |
|---|---|---|---|---|---|---|
| Teaching Scheme | | | | *Evaluation Scheme* | | |
| **Theory (Hrs.)** | **Tutorial (Hrs.)** | **Practical (Hrs.)** | **Credit** | ***University Exam (E)*** | ***Mid Sem Exam (Theory) (M)*** | ***Practical (Internal)*** |
| 4 | 0 | 0 | 4 | 80 | 0 | 20 |

**Note:**

1. **This subject is compulsory.**
2. **4 Credits will be over and above the existing credit structure.**
3. **This subject will be taught by faculty of English. For B. Pharm., the institute will have to nominate one faculty member for the subject.**
4. **In Institutes, where as the load is not managed by the lecturers of English only, please nominate the other faculty for teaching the course of Contributor Personality Development.**

## (A) Background

The Contributor Personality Program has been designed keeping in mind the following:

1.0 Technology students should not only be excellently trained in the technological field, they should acquire soft skills if they are to be successful. Every student must also learn about the techniques of effective participation in a group discussion. He/she must learn to prepare his/her resume and he/she should also be groomed for presenting himself/herself at an interview.

2..0 There is a great need to equip students with not only the right skill-sets but also the right mindsets.


3.0 The 'mindsets' needed in today's environment must support both (i) effective action and (ii) values and service oriented behavior.

Effective action without human values can lead to personal benefits for individuals but a long-term cost to both nation and society. Human values without effective action can lead to an inability on the part of the individual to perform and flourish in today's environment.

This combination of effectiveness with human values is crystallized in the concept of "contributor ship".

4.0 Students who adopt and develop the right mindsets early in their professional career are able to bring about a positive and sustainable change in their overall personality.

They are able to grow the right approaches to their peers, seniors, industry, and their own future. They become more responsible and capable of shaping their own lives.

Therefore, the program may be rightly called a "Contributor Personality Development Program".

5.0 Any program of this sort must, in order to be effective, be inspired and guided by a high ideal and principles/ practices flowing from that ideal.

The Contributor Personality Program is guided by the ideals and ideas of Swami Vivekananda – who represented in his leonine personality the highest ideals of human values combined with effective action.


## (B) Course Outline

Topics 1-6 relate to the basic axioms or "mental models" that students carry about themselves, about success, careers, contribution, etc. The right mental models are a necessary prerequisite for developing into a Contributor.

Topics 7-12 are 6 core practices that will help a student manifest the ideal of contributor ship in one's life.

Topics 13-15 relate to the students capability to connect into the job-market.

| Topic | Course Title |
|---|---|
| 1 | **Who is a Contributor**<br>Student develops an appreciation of who the Contributors are and how they fundamentally differ from Non-contributors in their overall approach to work, to other human beings, to society as a whole. |

| | |
|---|---|
| 2 | **The Contributor's identity**<br>Student develops his/ her own answer to the question "who am I?" The student becomes aware of the fact that Non-contributors usually define themselves in terms of what they have acquired in life (e.g. qualifications, position, years of experience, etc.) while Contributors define themselves in terms of what they will become or accomplish (e.g. capacity to deliver, commitment and ownership of the organization's purpose, etc.). |
| 3 | **The Contributor's vision of success**<br>The student explores the meaning of success in his life. Through this exploration, the student is expected to recognize that Contributors have a wider definition of success than Non-contributors. While Non-contributors define success in terms of material success, achievement, external impact, etc., Contributors are able to widen this definition of success to include personal fulfillment, development of self-esteem, ongoing development of personal capabilities etc. |
| 4 | **The Contributor's vision of career**<br>The student learns to distinguish between an "acquisitive career" and a "contributive career". An acquisitive career is one in which the career-seeker is focused on acquiring higher position, higher salary, more benefits etc. This preoccupation with selfish interests often damages the individual's career, as well as, damages the organization and society. A contributive career is one where the career-seeker is focused on contributing, with rewards being a by-product of the contributions made. |
| 5 | **The scope of contribution**<br>The student learns to perceive that in all type of work, every type of role, there is a possibility of contributing at multiple levels – contributing to self, contributing to organization, and contributing to society.<br><br>The student also appreciates the difference between "acquisition for self" and "contribution to self" – the former being material acquisition and the latter being conscious development of oneself through the medium of one's career. |
| 6 | **Embarking on the journey to contributor ship**<br>The student recognizes the fundamental "building blocks" for becoming a Contributor – the first building block being a shift from a "victim" to being a "creator of one's destiny"; the second building block being acceptance of the ideal of contributor ship; the third building block being the willingness to take full responsibility for one's own development; the fourth building block being the capacity to reflect on one's development and make appropriate modifications. |

| | |
|---|---|
| 7 | **Design Solutions**<br>When faced with a challenge, the Contributor's first response is: "Can we find a solution?" This is unlike a Non-contributor who may respond to the challenge by trying a little and giving up, blaming others, or finding excuses to cover up the issue.<br><br>Whereas, the Contributor finds a solution. In other words, the Contributor develops the capacity to find solutions through continuous practice and learning from other Contributors.<br><br>In this topic, students learn the importance of willingness and ability to find solutions. |
| 8 | **Focus on value**<br>What does creating value mean? It means making a positive difference, a tangible impact, a specific contribution to any situation. This positive difference or impact can be in the form of achieving a specific goal, creating a product, creating 'human touch' in a particular interaction, or enhancing one's own capacity, or the capacity of one's colleagues and team- mates.<br><br>Contributors are therefore extremely result-focused, but the result is measured in terms of value created.<br><br>In this topic, students learn to clarify the meaning of the word "value" and how value is created in various situations. |
| 9 | **Engage deeply**<br>Contributors are instantly distinguished by the way they approach work. They get involved. They are enthusiastic. They go deep into the subject. In short, Contributors love what they do.<br><br>This is in direct contrast to Non-contributors who want to do only what they love - an approach that seems reasonable until you realize that life and workplaces have so much variety that you may very often be called upon to do tasks that seem unpleasant or boring until you get involved.<br><br>In this topic, students learn the importance of engaging deeply with whatever work they do – at work, in study, in personal life. |
| 10 | **Think in Enlightened Self-interest**<br>Contributors think in Enlightened Self-Interest. In every situation they get into, they find a way to create something good for self and for all at the same time – including team mates, bosses, customers and their organization.<br><br>Contrasting to this is the mindset of a Non-Contributor. Such a person is only concerned with his/ her own self-interest in a situation. He/she is not concerned about the impact (positive or negative) on the other person. This leads to unpleasant |

| | |
|---|---|
| | situations, broken relationships, unhappy team-mates, subordinates, and bosses, and lower trust in any situation.<br><br>Students are expected to learn to appreciate the importance of thinking win-win for all stakeholders and also in various situations. |
| 11 | **Practice Imaginative Sympathy**<br>One of the unique qualities of Contributors is their ability to appreciate and understand others' life situation, others' mental condition, and others' point of view. How do they do this?<br><br>They have consciously developed a 'way of thinking' called 'Imaginative Sympathy'. In this way of thinking, they are able to give due importance to the human aspects of a situation, and not just the technical or commercial aspects.<br><br>But this is not all. Imaginative Sympathy goes beyond looking at the human aspects of the situation. It also means that Contributors are able to anticipate possible interactions or reactions, they are able to take a multi-dimensional view of a situation and they are able to bring about changes or results while taking everybody along with them.<br><br>Imaginative Sympathy translates itself into active concern for others. Students will learn the importance and consequences of Imaginative Sympathy in a workplace situation. |
| 12 | **Demonstrate Trust Behavior**<br>Contributors recognize that they are able to achieve results and make contributions with the help of other human beings. They receive this help if and only if they are trusted and, in turn, trust. Contributors practice trust behavior from very early in their career, thereby building a huge trust balance (like a bank balance) over their career and relationships.<br><br>The term Trust Behavior may be described as character-in-action. This includes keeping one's word and commitments, staying with a task, acting with integrity in every situation, making sure that there is complete transparency in one's actions and interactions, etc.<br><br>Students are expected to learn to develop a deep appreciation of trust behavior and how it is practiced. |
| 13 | **Resume Building**<br>In this topic, students learn to develop a resume for the job-market. Students will learn to develop both a generic resume and resumes specific to some types of jobs. Students learn about best practices and common errors in developing their resume. |

| | |
|---|---|
| | Most important, students learn to analyze the jobs offered and present themselves in terms of their potential / willingness to contribute to the job. |
| 14 | **Group Discussions (GDs)**<br>In this topic, students learn (i) how to participate in a group discussion from the contributor's view-point (i.e. how to speak) (ii) how to contribute to the development of the topic (i.e. what to speak) and (iii) to develop the Contributor's view-point on various GD topics (i.e. how to interpret a topic of discussion from the point of view of a contributor) |
| 15 | **Interview Skills**<br>In this topic, students learn about (i) common interview questions and how to develop answers (ii) typical challenges faced in interviews beyond the questions (such as body language, grooming, presentation) (iii) most important, the student learns the importance of trust building and creating confidence in the interview. |

## (C) Course Plan

The course duration is 48 hours. It can be conducted in sessions of 1 hour each or some of the sessions can be combined as 2 hours each. The course plan is as follows –

| | |
|---|---|
| **Topic 1: Who is a Contributor –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol I)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 2: The Contributor's identity –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol I)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 13: Resume Building**<br>– 2 hours for Concepts, Tools, and Techniques<br>– 2 hours for Projects | **4 hours** |
| **Topic 3: The Contributor's vision of success –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol I)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 4: The Contributor's vision of career –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol I)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 5: The scope of contribution –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol I)<br>– 1 hour Presentations and Projects | **3 hours** |

| | |
|---|---|
| **Topic 6: Embarking on the journey to contributorship –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol I)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 14: Group Discussions (GDs)**<br>– 2 hours for Concepts, Tools, and Techniques<br>– 2 hours for Projects and Practice | **4 hours** |
| **Topic 7: Design Solutions –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol II)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 8: Focus on value –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol II)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 9: Engage deeply –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol II)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 10: Think in Enlightened Self-interest –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol II)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 11: Practise Imaginative Sympathy –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol II)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 12: Demonstrate Trust Behavior –**<br>– 2 hours Theory and practice exercises based on Contributor Personality Program Workbook (Vol II)<br>– 1 hour Presentations and Projects | **3 hours** |
| **Topic 15: Interview Skills**<br>– 2 hours for Concepts, Tools, and Techniques<br>– 2 hours for Projects and Practice | **4 hours** |
| **TOTAL** | **48 hours** |

## (D) Examination Approach

Total marks: 150. Break-up of marks -

(i) Final exam : 70 marks (equal weightage for topics 1-15)

(ii) Presentations and projects for topics 1-12: 30 marks

(iii) Projects for topics 13-15: 50 marks

## (E) Instructional Strategy

1.0 The entire course will use a three-level instructional strategy

Level I: Classroom Explorations

Level II: Projects and Presentations

Level III: Self-study by students

2.0 Level I: Classroom Explorations

1. The Classroom Explorations will be organized around the 'Contributor Personality Program – Study Book'.

   The Study Book may be downloaded by the student from the resource site produced by GTU.

2. The Classroom Explorations involves two kinds of explorations:

   (i) Exploration of key concepts / frameworks such as "contributors vision of success" etc.

   (ii) Exploration of the examples provided in the CPP Study Book.

3. The Classroom Explorations will be supported by Session Guide Sheets available online in the CPP ActivGuide.

3.0 Level II: Projects & Presentations

1. The entire Classroom Exploration process will be supplemented by projects and presentations.
2. Session Guides will provide sample topics for projects and presentations. Individual instructors will be free to develop their own projects/ presentation topics also.
3. This will not only enhance conceptual clarity but also build presentation, public-speaking, report writing, and group discussion skills of the students.

4.0 Level III: Self Study by students

1. Students will be given extensive learning support (upto 400 learning units) in the ActivGuide website. This will include videos, presentations, tests, etc.
2. Students can refer to ActivGuide on their own time through internet.

## (F) Reference Material

## Basic Study Material

| SN | Author/s | Name of Reference | Publisher | Edition |
|---|---|---|---|---|
| 1 | Illumine Knowledge Resources Pvt. Ltd. (Downloadable from the internet) | Contributor Personality Program Workbook (Vols I & II) | Illumine Knowledge Resources Pvt. Ltd | Latest |
| 2 | Illumine Knowledge Resources Pvt. Ltd. (will be made available to all students on the Internet) | Contributor Personality Program ActivGuide | Illumine Knowledge Resources Pvt. Ltd | Latest |

## Reference Books

| Topic | Course Title | Reference |
|---|---|---|
| 1 | **Who is a Contributor** | 1. On Contributors, Srinivas V.; Illumine Ideas, 2011<br>2. Enlightened Citizenship and Democracy; Swami Ranganathananda, Bharatiya Vidya Bhavan, 1989<br>3. Personality Development, Swami Vivekananda; Advaita Ashrama |
| 2 | **The Contributor's identity** | 1. To have or to be, Erich Fromm; Continuum International Publishing Group, 2005<br>2. The art of being, Erich Fromm; Continuum International Publishing Group, 1992<br>3. Raja Yoga, Swami Vivekananda; Advaita Ashrama |
| 3 | **The Contributor's vision of success** | 1. Eternal Values for a Changing Society – Vol IV (Ch 25, 35), Swami Ranganathananda; Bharatiya Vidya Bhavan, 1993<br>2. Karma Yoga, Swami Vivekananda; Advaita Ashrama |
| 4 | **The Contributor's vision of career** | 1. Six Pillars of Self Esteem , Nathaniel Branden; Bantam, 1995<br>2. Mindset: The New Psychology of Success, Carol S. Dweck; Random House Publishing Group, 2007 |
| 5 | **The scope of contribution** | 1. Awakening India, Swami Vivekananda; Ramakrishna Mission, New Delhi, 2011<br>2. Eternal Values for a Changing Society – Vol IV |

| | | |
|---|---|---|
| | | (Ch 35), Swami Ranganathananda; Bharatiya Vidya Bhavan, 1993<br>3. Lasting Contribution: How to Think, Plan, and Act to Accomplish Meaningful Work, Tad Waddington; Agate Publishing, 2007 |
| 6 | **Embarking on the journey to contributor ship** | 1. Vivekananda: His Call to the Nation, Swami Vivekananda; Advaita Ashrama<br>2. Eternal Values for a Changing Society – Vol IV (Ch 33), Vol III (Ch 19, 21, 30) Swami Ranganathananda; Bharatiya Vidya Bhavan, 1993<br>3. Lectures from Colombo to Almora, Swami Vivekananda; Advaita Ashrama |
| 7 | **Design Solutions** | 1. Why not?: how to use everyday ingenuity to solve problems big and small, Barry Nalebuff, Ian Ayres; Harvard Business School Press, 2003<br>2. How to Have a Beautiful Mind, Edward De Bono; Vermilion, 2004 |
| 8 | **Focus on value** | 1. The value mindset: returning to the first principles of capitalist enterprise (Ch 8 & 9); Erik Stern, Mike Hutchinson; John Wiley and Sons, 2004<br>2. Managing for Results, Peter F. Drucker; HarperCollins, 2009 |
| 9 | **Engage deeply** | 1. The Power of Full Engagement: Managing Energy, Not Time, is the Key to High Performance and Personal Renewal, Jim Loehr, Tony Schwartz; Simon and Schuster, 2003 |
| 10 | **Think in Enlightened Self-interest** | 1. The 7 Habits of Highly Effective People, Stephen R. Covey; Simon and Schuster, 2004<br>2. Creating Shared Value, Michael E. Porter and Mark R. Kramer; Harvard Business Review; Jan/Feb2011, Vol. 89 Issue 1/2 |
| 11 | **Practice Imaginative Sympathy** | 1. Eternal Values for a Changing Society – Vol IV (Ch 8, 10, 23, 35, 37), Swami Ranganathananda; Bharatiya Vidya Bhavan, 1993<br>2. Eternal Values for a Changing Society – Vol III (Ch 18), Swami Ranganathananda; Bharatiya Vidya Bhavan, 1993 |
| 12 | **Demonstrate Trust Behavior** | 1. The Speed of Trust: The One Thing That Changes Everything, Stephen M. R. Covey, Rebecca R. Merrill, Stephen R. Covey; Free Press, 2008<br>2. Integrity: The Courage to Meet the Demands of |

| | | |
|---|---|---|
| | | Reality, Henry Cloud; HarperCollins, 2009<br>3. Responsibility at work: how leading professionals act (or don't act) responsibly, Howard Gardner; John Wiley & Sons, 2007 |
| 13 | **Resume Building** | 1. What Color Is Your Parachute? 2012: A Practical Manual for Job-Hunters and Career-Changers, Richard Nelson Bolles; Ten Speed Press, 2011<br>2. The what color is your parachute workbook: how to create a picture of your ideal job or next career, Richard Nelson Bolles; Ten Speed Press, 2011 |
| 14 | **Group Discussions (GDs)** | 1. Effective Group Discussion: Theory and Practice, Gloria J. Galanes, Katherine Adams; McGraw-Hill, 2004 |
| 15 | **Interview Skills** | 1. What Color Is Your Parachute? 2012: A Practical Manual for Job-Hunters and Career-Changers, Richard Nelson Bolles; Ten Speed Press, 2011<br>2. The what color is your parachute workbook: how to create a picture of your ideal job or next career, Richard Nelson Bolles; Ten Speed Press, 2011 |

## General References:-

| SN | Author/s | Name of Book | Publisher | Edition |
|---|---|---|---|---|
| 1 | Swami Ranganathananda | Universal Message of the Bhagavad Gita (Vol 1-3) | Advaita Ashrama, Kolkata | Latest |
| 2 | Swami Ranganathananda | Eternal Values for a Changing Society (Vol 1-4) | Bharatiya Vidya Bhavan | Latest |
| 3 | Asim Chaudhuri | Vivekananda: A Born Leader | Advaita Ashrama, Kolkata | Latest |
| 4 | Swami Vivekananda | Complete Works of Swami Vivekananda (Vol 1-9) | Advaita Ashrama, Kolkata | Latest |
| 5 | Swami Vivekananda | Letters of Swami Vivekananda | Advaita Ashrama, Kolkata | Latest |

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
### Year – 2 (Semester – III) (W.E.F. JULY 2018)

**Subject Name: Basic Mathematics**
**Subject Code: 4639301**

**Objective**

The objective of this course is to present the foundations of many basic mathematical topics used in Computer Science including RDBMS, Data Structures, Analysis of Algorithms, Theory of Computation, Cryptography, Artificial Intelligence, Statistics and others. This course will enhance the student's ability to think logically and mathematically.

**Prerequisites**: Binary number system, C Language

**Contents**:

| Sr. No. | Topics | Weightage Percentage |
|---|---|---|
| 1 | **Set Theory, Propositional & Predicate Logic:**<br>Set Theory: Basic Concepts of Set Theory: Definition, Two Methods to Describe (Represent) Sets; Examples, (Im)proper Subsets, Superset, Equality of Sets; Empty (Null) Set, Universal Set, Finite and Infinite Sets, Power Set; Operations on Sets: Union, Intersection, Complement, Venn Diagrams; Disjoint Sets, Various Laws: Identity, Idempotent, Commutative, Associative, Distributive, Absroption, DeMorgan; Difference (Relative Complement), Symmetric Difference of Two Sets; Cartesian Product; Power Set of a Set; Computer Representation of Sets; Examples; Theorems and Exercises (without Proof)<br><br>**Propositional Logic:** Definition, Statement (Proposition) & Notation, Truth Values, Connectives: Negation, Conjunction, Disjunction, Implication (condition), Bi-implication (Bi-conditional),<br>Truth Tables for all Connectives, Statement Formulas (Well-formed Formulas), Truth Tables, Tautologies, Contradiction, Logical Equivalence: Commutative Laws, Associative Laws, Distributive Laws, Absroption Laws, Idempotent Laws, Double Negation Law, DeMorgan's laws, Examples; Validity of Arguments, Some Valid Argument Forms: Modus Ponens, Modus Tollens, Disjunctive Syllogism, Dilemma, Equivalence of Formulas: Conjunctive Simplification, Disjunctive Addition, Conjunctive Addition, Examples and Exercises; Theorems (without Proof)<br><br>**Predicate Logic:** Definition of Predicates; Variables, Quantifiers: Universal Quantifiers, Existential Quantifiers; Free & Bound Variables; Negation of Predicates; Generalized DeMorgan's Laws; Valid Formulas and Equivalences; Additional Rules of Inference; Examples and Exercises; Theorems (without Proof) | **26%** |

| | | |
|---|---|---|
| **2** | **Proof Techniques, Matrices**<br><br>**Proof Techniques**: Direct Proof, Indirect Proof, Proof by Contradiction; Proving Bi-implications; Proving Equivalence Statements; Fallacies (Errors) in Proofs; Examples and Exercises<br><br>**Matrices**: Introduction; Representation of a Matrix; Equality of Matrices; Special Matrices: Rectangular / Square Matrices, Null (Zero) Matrix, Unit Matrix, Diagonal Matrices, Triangular Matrices; Sum and Difference of 2 Matrices; Multiplication of 2 matrices; Transpose of a Matrix, Symmetric Matrices; Boolean (Zero-One) Matrices, Boolean Join, Boolean Meet; Theorems and Exercises (without Proof) | **13%** |
| **3** | **Integers, Mathematical Induction**<br><br>**Integers**: Introduction, Basic Properties of Integers: Closure, Commutative Laws, Associative Laws, Identity Elements, Additive Inverse, Distributive Laws, Cancellation Laws; Well-ordering Principle; Division Algorithm: Quotient, Remainder, ***div*** and ***mod*** operators, Divisibility; Greatest Common Divisor (GCD); Euclidean Algorithm for Finding the GCD; Relative Prime; Least Common Multiples (LCM); Representation of Integers in Computer; Decimal, Binary, Octal, and Hexadecimal Representation; Operations on Binary Numbers: Addition, Subtraction; Theorems and Exercises (without Proof)<br><br>**Mathematical Induction**: Introduction; First Principle of Mathematical Induction; 3 Steps: Basis Step, Inductive Hypothesis, Inductive Step; Second Principle of mathematical Induction; Application: Loop Invariant (Program Correctness) | **13%** |
| **4** | **Relations and Functions**<br><br>**Relations**: Introduction, Binary Relation, Definition; Representation: Set of Ordered Pairs, Arrow Diagram, Matrix, Graph; Domain & Range of Relation; Universal Relation, Void Relation; Properties of a Relation: Reflexive, Symmetric, Transitive, Anti-symmetric, Irreflexive; Equivalence Relations, Partition, Block of Partition, Equivalence Classes and Partitions; Inverse of a Relation; Composition of Relations; Closures: Reflexive, Symmetric, Transitive; Theorems and Exercises (without Proof)<br><br>**Functions**: Introduction & Definition; Arrow Diagram of a Function; Domain, Co-domain (Target), Range of a Function; Special Function: Identity Function, Constant Function; One-to-One (Injective), Onto (Surjective), and One-to-One & Onto (Bijective) Functions; Composition of Functions; Inverse of a Function, Left-invertible & Right-invertible Functions; Floor & Ceiling Functions; Cardinality of a Finite Set; Theorems and Exercises (without Proof) | **23%** |
| **5** | **Graphs and Trees**<br><br>**Graphs**: Introduction, Definition; Initial & Terminal Nodes, Adjacent Nodes; Directed Edge, Undirected Edge, Directed Graph (Digraph), Undirected Graph, Mixed Graph; Loop (Sling); Distinct Edges, Parallel Edges; Multi-graph, Simple Graph; Weighted Graph; Isolated Nodes, Null Graph; Isomorphic | **25%** |

| | | |
|---|---|---|
| | Graphs; In-degree, Out-degree, Total-degree; Subgraphs; Reflexive, Symmetric, Transitive Digraphs; Paths, Length of Path of a Graph; Simple Path (Edge Simple), Elementary Path (Node Simple), Cycle (Circuit), Simple Cycle, Elementary Cycle; Path of Minimum Length (Geodesic), Distance between Two Nodes, Triangle Inequality; Reachability, Reachable Set of a Node, Reachable Set of a Set of Nodes, Node Base; Connected Graphs: Strongly, Unilaterally, Weakly Connected Graphs & Components; Matrix Representation of Graphs (Adjacency Matrix). In-degree, Out-degree of a Graph from Adjacency Matrix; An (A is Adjacency Matrix) to give Number of Paths of Length n; Path Matrix (Reachability Matrix) of a Graph; Warshall's Algorithm to Produce Path Matrix; Algorithm to Give Lengths of Minimum Paths; Theorems and Exercises (without Proof)<br><br>**Trees**: Introduction, Definition, Root, Branch Nodes, Leaf (Terminal Node); Different Representations of Trees; Forests, Subtrees; M-ary Tree, Full or Complete M-ary Tree; Binary Tree, Full (Complete) Binary Tree; Conversion of M-ary Tree to Binary Tree; Linked Allocation Technique to Represent Binary Tree in Computer. Traversal of Binary Tree: Pre-order, In-order, and Post-order Traversal; Theorems and Exercises (without Proof) | |

**Text Book:**


1. D. S. Malik & M. K. Sen, "Discrete Mathematics", Cengage Learning (2004)
2. J. P. Tremblay and R.Manohar, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw-Hill (2010) – only for Unit-5 (Graphs & Trees).


**Reference Books:**


1. K. H. Rosen, "Discrete Mathematics and its applications", Tata McGraw-Hill, 6th edition,
2. Bernard Kolmann & others, "Discrete Mathematical Structure", Pearson Education, Sixth Edition
3. Edgar G. Goodaire, Michael M. Parmenter. "Discrete Mathematics with Graph Theory", PHI
4. Ralph P Grimaldi & B V Ramana, "Discreet and Combinatorial mathematics: An Applied Introduction", Pearson Education, 5th Edition (2018)
5. J. P. Tremblay and W. K. Grassman. "Logic and Discrete Mathematics", Pearson Education


**Chapter-wise Coverage from the Text Book:**

Unit-1: Chapter-1
Unit-2: Chapters-1 & 4
Unit-3: Chapter-2
Unit-4: Chapters-3 & 5
Unit-5: Text Book-2: 5-1: 5-1.1 to 5-1.4; 5-2: 5-2.1.

**Accomplishment of Students after Completing the Course:**

Students will be able to understand various algorithms and implement them in C language. More specifically, they will be able to understand and apply the concepts of sets, logic, cross product of sets and relation, functions, matrices, and basic algorithms related with binary tree and graphs.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
## Year – 2 (Semester – III) (W.E.F. JULY 2018)

## Subject Name: Basic Mathematics
## Subject Code: 4639301_Practical List

**List of Computer Lab Exercises (To be implemented in C Language)**

**Objectives**: To get an insight of data structures used in implementation of various basic mathematical concepts and algorithms based on these concepts.

**Prerequisites**: C Programming Language

**Advice (Note) to Teachers**:

- The list of exercises given below is an indicative list. It is expected that students will be able to visualize how various mathematical concepts can be used to solve real-life problems through programming.
- Some exercises have been labeled as "Mandatory" while other exercises have been marked as "Desirable". It is expected that all the students will do Mandatory exercises while bright students will additionally do Desirable exercises as well.

**List of Computer Lab Exercises**

**1. Mandatory Exercises: Set Theory and Logic**

(a) Start with a NULL set and add elements one-by-one: Use different ways of implementing sets and understand the pros and cons of each of these methods
(b) Given an element value, check whether it is a member of the set or not
(c) Find out the number of elements of a given set.
(d) Complement of a set; Union, Intersection
(e) Test whether a given set X is a subset of the set A or not.
(f) Test whether two given sets are equal or not
(g) Difference and Symmetric Difference of two sets
(h) Write functions for evaluating Logical-And, Logical-Or, Logical-Not, and Logical-XOR of given two Boolean values.

**Desirable Exercises: Set Theory & Boolean Logic**

(a) Find Cartesian Product of two given sets
(b) Find the Power Set of a given set
(c) Truth table of a Logical expression

**2. Mandatory Exercises: Matrices**

(a) Write a function to Create a Matrix of size m x n, and another function to Print a Matrix of size m x n.
(b) Print the Transpose of a given matrix A.
(c) Write a program to generate Null matrix of order (m x n) and Unit matrix of order n.
(d) Take as input two matrices, A & B and print A + B and A – B. First check whether it is possible to compute (A + B) and (A – B) or not.
(e) For a given square matrix A, print the Diagonal, Upper Triangular and the Lower Triangular

Matrices of A.
(f) Evaluate Scalar Product of a Matrix A: For example, k A, where k is a constant (number)
(g) Take as input two matrices, A & B and print (A * B) and (B * A). First check which ones out of (A * B) and (A * B) are possible to compute.
(h) Given two matrices, determine whether one matrix is the inverse of the other matrix.

**Desirable Exercises – Matrices**
(a) For a given matrix A, find its inverse.
(b) Convert a given matrix with many zeros (0's) to Sparse matrix structure and vice versa.
(c) Evaluate the Value of the Determinant of the given Matrix.
(d) Find the Determinant of a square matrix and compute its value. Can it be done recursively?

**3. Mandatory Exercises: Integers**
(a) Write a program to generate all Prime Numbers between 1 and a given integer N.
(b) Given an integer N, test whether it is a Prime Number or a Composite Number.
(c) Write a program to find Factorial of a given number using (i) an iterative algorithm, and (ii) a recursive algorithm.
(d) Write a program to generate a set of Fibonacci Series up to (less than or equal to) a given number N using (i) an iterative algorithm, and (ii) a recursive algorithm.
(e) Evaluate Prime Factors of a given positive integer. (Assume that an array of prime numbers is available)
(f) Write a program to evaluate GCD (Greatest Common Divisor) of two positive integers using Euclidean algorithm
(g) Write Floor and Ceiling functions.
(h) Evaluate LCM (Least Common Multiple) of two positive integers.
(i) Convert a given Integer in decimal umber system into Binary, Octal, and Hexadecimal number systems.

**Desirable Exercises – Integers**
(a) Write a program using recursive algorithm to implement Tower of Hanoi problem.
(b) Convert a given Real Number with 3 decimal digits into Binary, Octal, and Hexadecimal.

**4. Mandatory Exercises: Binary Relation**
(a) Given the Boolean Matrix of a Binary Relation, determine whether the Relation is Reflexive and / or Symmetric.

**Desirable Exercises: Binary Relation**
(a) For a given Relation, generate a Boolean Matrix. Using Boolean Matrix, determine whether the Relation is Transitive
(b) Find the Partitions of the set using the given equivalence relation defined on the set.

**5. Mandatory Exercises: Graphs & Trees**
(a) Take inputs of a graph and generate Adjacency Matrix of the graph. Print In-degree, Out-degree, and Total degree of each node.
(b) Represent a given binary tree using linked list structures. Print (i) Number of Leaf nodes, (ii) Intermediate nodes, and (iii) Total number of nodes.
(c) Given a binary tree represented as linked structure, traverse that binary tree using (i) Pre-order (or Depth-first) Traversal, (ii) In-order Traversal, (iii) Post-order Traversal using Recursive Algorithm
(d) Search whether an element is present in a given binary tree or not using Depth-first Search (DFS).

**Desirable Exercises: Graphs & Trees**

(a) Represent the given Adjacency Matrix of a graph as Sparse Matrix. Print In-degree, Out-degree, and Total degree of each node.

(b) Given a binary tree represented as linked structure, traverse that binary tree using (i) Pre-order Traversal, (ii) In-order Traversal, (iii) Post-order Traversal using Iterative Algorithm

(c) Use Linked List structure to represent threaded binary tree.

(d) Add a new node in a given AVL binary tree.

(e) Delete a specified node from a given AVL binary tree.

(f) Search whether an element is present in a given binary tree or not using Breadth-first Search (BFS).

**Reference Books:**

1. Kernighan and Ritchie, "C Programming", Pearson Education
2. J. P. Tremblay and W. K. Grassman. "Logic and Discrete Mathematics", Pearson Education,
3. ISRD Group, "Data Structures using C", Tata McGraw Hill, 2006

**Reference Websites:**

1. www.uva.onlinejudge.org
2. www.cse.iitd.ernet.in/~bagchi/courses/discrete-book/fullbook.pdf

**Accomplishment of the student after completing the course:**

The student will be able to implement many of the concepts in C language. More specifically, the concept of Sets, Cross Product of Sets, Prime Numbers, Matrices, and basic algorithms related with Binary Tree and Graphs.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
### Year – 2 (Semester – III) (W.E.F. JULY 2018)

**Subject Name: Programming in JAVA**
**Subject Code: 4639302**

## 1. Learning Objectives:

- To interpret the concepts of object oriented programming language and easily use these concepts in Java program.
- To develop proficiency in creating robust applications using the Java Programming Language.
- To understand concepts of Interface, Lambda Expressions, Inner classes, Generic Programming and to implement them.
- To implement application including different file operations.
- To understand database connectivity and work with the JDBC applications.

### 2. Desirable:

Knowledge of the C, C++ programming language and Database.

### 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to Java**<br>Programming Platform, Java Buzzwords, overview of applet and internet, History of Java, common misconception about Java, The Java Programming Environment: installing JDK, using the command line tools, using IDE, Programming Structures in Java: data types, operators, working with Strings, working with Scanner class for input and output, control flow, big number and Arrays. | **10%** |
| **Unit II** | **Objects and Classes, Inheritance, Interface**<br>Objects and Classes: classes, objects, objects and object variables, LocalDate Class, Mutator and Accessor methods, defining your own classes, static fields and methods, method parameters, object construction, packages and the class path.<br>Inheritance: classes, superclasses, and subclasses, overriding methods, inheritance hierarchies, polymorphism, final class and methods, casting, abstract classes and, protected access, Object: Cosmic superclass, Object Wrappers and Autobxing and Enumeration classes. Interface. | **20%** |
| **Unit III** | **Lambda Expressions, Inner classes.**<br>Lambda Expressions, Inner classes: Interfaces, examples of interfaces, Why Lambdas? Syntex of lambda expression, functional interfaces, method reference, constructor reference, variable scope, processing lambda expression and inner classes. | **20%** |
| **Unit IV** | **Exception Handling, Generic Programming**<br>Exception Handling: dealing with errors, catching exceptions, tips for using exceptions:<br>Generic Programming: A Simple Generic class, generic methods, | **20%** |

| | | |
|---|---|---|
| | bounds for type variables, generic code and the VM, restrictions and limitations and inheritance rules for generic types. | |
| **Unit V** | **Input and Output**<br>Input/Output Streams: reading writing bytes, combining IO steam filers, Text Input and Output: write text output, read text outpout, saving object in text format, character encoding, Reading and Writing, Working with Files: paths, reading and writing files, creating files and directories, copying, moving and deleting files and getting file info. | **20%** |
| **Unit VI** | **Database Programming**<br>The Design of JDBC, JDBC Driver Types, SQL, JDBC Configuration: URL, driver jar files, starting the database, registering the driver class, connecting to the database, Working with JDBC Statements: executing SQL statement, managing connections, statements, resultsets, SQL exceptions, Query Execution: prepared statement. | **10%** |

**4. Text Book(s):**

1. Cay S. Horstmann "Core Java , Volume I – Fundamentals", 10th Edition, Pearson Education , 2017
2. Cay S. Horstmann "Core Java , Volume II – Advanced Features", 10th Edition, Pearson Education , 2017

**5 Suggested Additional Reading:**

1) The complete reference Java, Herbert Schildt, Mc Grow Hill
2) H.M.Deitel, P.J.Deitel, "Java How to Program", Sixth Edition, Pearson Education (2007)
3) Ivor Horton's "Beginning Java 2" JDK 5 Edition, Wiley Computer Publishing, (2007).
4) Pravin Jain, "The class of Java" Pearson Education, (2010).
5) Ken Arnold, James Gosling, David Holmes, "The Java Programming Language", Addison-Wesley Pearson Education (4th Edition – 2005).
6) Raj Kumar Buyya, S. Thamarai Selvi, & Xing Chen Chu, "Object-Oriented Programming with Java: Essentials & Applications", Tata McGraw Hill
7) Sharan Zakhour, Scott Hommel, Jacob Royal, Isaac Rabinovitch, Tom Risser, Mark Hoeber "The Java Tutorial", Addison-Wesley Pearson Education(4th Edition),
8) E.Balagurusamy, "Programming with Java", 3e, TMH (2007)

**Web Reference:**

1. https://docs.oracle.com/en/java
2. http://www.learnjavaonline.org/
3. http://java.sun.com/docs/books/tutorial/index.html
4. http://www.javaworld.com

**6. Chapter wise Coverage from the Text Book:**

| Unit # | Book# | Chapter |
|---|---|---|
| I | 1 | 1.1 to 1.5,2.1 to 2.3, 3.1 to 3.10 |
| II | 1 | 4.1 to 4.8, 5.1, 5.2, 5.4, 5.6, 6.1, 6.2 |
| III | 1 | 6.3 , 6.4 |

| | | |
|---|---|---|
| IV | 1 | 7.1-7.3, 8.1-8.7 |
| V | 2 | 2.1 to 2.3, 2.5.1 to 2.5.5 |
| VI | 2 | 5.1 to 5.5.1 |

## 7. Accomplishments

- Students will understand object oriented concepts, lambda expressions, generic classes and to implement them using the Java programming language
- Ability to create a robust application using Java programming and to work with Database connectivity.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
## Year – 2 (Semester – III) (W.E.F. JULY 2018)

## Subject Name: Programming in JAVA
## Subject Code: 4639302_Practical List

### A. List of Practical Related to JAVA:

**General Instructions for Faculty Members/Lab Instructors:**

- Use of integrated development environment (IDE) software application is restricted. All the programs mentioned in this list are to be performed using "Notepad or another text editor", JRE environment: Linux or Windows.
- Java is one of the most widely used programming languages in the field of Enterprise applications, Web Applications, Mobile Applications, Embedded systems, web server and applications server , networking applications , security functionality and environment and scientific applications because of its simplicity, modular programming and nice design etc.
- Students should be exposed to best programming practices for the given development environment.

---

| | |
|---|---|
| 1 | Install the JDK (Download the JDK and install it.)<br>• Set path of the jdk/bin directory.<br>• Create the java program<br>• Compile and run the java program<br>Write a simple "Hello World" java program, compilation, debugging, executing using java compiler and interpreter. |
| 2 | Write a program to pass Starting and Ending limit and print all prime numbers and Fibonacci numbers between this range. |
| 3 | Write a java program to check palindrome number.<br>Input: 329 Output: not palindrome number<br>Input: 12321 Output: palindrome numbe |
| 4 | Write a java program to print factorial of a number.<br>Input: 5 Output: 120<br>Input: 6 Output: 720 |
| 5 | Write a java program to check Armstrong number.<br>Input: 153 Output: Armstrong number<br>Input: 22 Output: not Armstrong number |
| 6 | Write a program in Java to find maximum of three numbers using conditional operator |
| 7 | Write a java program which should display maximum and minimum number of given 3 numbers. |
| 8 | Write a program in Java to multiply two matrix |
| 9 | Write a java program to create a class "Matrix" that would contain integer values having varied numbers of columns for each row. Print row-wise sum of the integer values for each row. |
| 10 | Write a Java application which takes several command line arguments, which are supposed to be names of students and prints output as given below:<br>(Suppose we enter 3 names then output should be as follows)..<br>Number of arguments = 3<br>1.: First Student Name is = Tom<br>2.: Second Student Name is = Dick |

| | |
|---|---|
| | 3.: Third Student Name is = Harry<br>Hint: An array may be used for converting from numeric values from 1 to 20 into String |
| 11 | Write a Java application to count and display frequency of letters and digits from the String given by user as command-line argument. |
| 12 | Create a class "Student" that would contain enrollmentNo, name, and gender as data members. Create appropriate getter and setter methods for the "Student" class and constructors to initialize the data members. Also demonstrate constructor chaining. |
| 13 | Write a program in Java to demonstrate use of **this** keyword. Check whether **this** can access the private members of the class or not. [Refer class student in Q12 to perform the task] |
| 14 | Create a class "Rectangle" that would contain length and width as data members.<br>Define constructors **[constructor overloading (default, parameterized and copy)]** to initialize the data members. Define the member functions to find area and to display the number of objects created.<br>[Note: define initializer block, static initializer block and the static data member and member function. Also demonstrate the sequence of execution of initializer block and static initializer block] |
| 15 | Write a java program static block which will be executed before main ( ) method in a class. |
| 16 | Write programs in Java to use Wrapper class of each primitive data types |
| 17 | Write a class "circle" with radius as data member and count the number of instances created using default constructor only. [**Constructor Chaining]** |
| 18 | Create a class Vehicle with data member vehicle_type. Inherit the class in a class called car with data member model_type, company name etc. display the information of the vehicle by defining the display function in both super and sub class **[ Method Overriding]** |
| 19 | Create a class "Account" containing accountNo, and balance as data members.<br>Derive the Account class into two classes named "Savings" and "Current". The "Savings" class should contain a data member named interestRate, and the "Current" class should contain a data member called overdraftLimit. Create appropriate member functions for all the classes to enable functionalities to check balance, deposit, and withdraw amount in Savings and Current account.<br>[Ensure that the Account class cannot be instantiated.] |
| 20 | Write a program in Java in which a subclass constructor invokes the constructor of the super class and instantiate the values. [ refer class Account and sub classes savingAccount and CurrentAccount in Q 19 for this task] |
| 21 | Write a program in Java to demonstrate the use of **'final'** keyword in the field declaration. How it is accessed using the objects. |
| 22 | Write a java program to illustrates how to access a hidden variable. Class **A** declares a static variable **x**. The class **B** extends **A** and declares an instance variable **x**. **display ( )** method in **B** displays both of these variables. |
| 23 | Describe **abstract** class called **Shape** which has three subclasses say **Triangle**,**Rectangle**, **and Circle**. Define one method **area ()** in the abstract class and override this **area ()** in these three subclasses to calculate for specific object i.e. area () of Triangle subclass should calculate area of triangle etc. Same for **Rectangle** and **Circle** |
| 24 | Write a java program to implement an interface called Exam with a method Pass (int mark) that returns a boolean. Write another interface called Classify with a method Division (int average) which returns a String. Write a class called Result which implements both Exam and Classify. The Pass method should return true if the mark is greater than or equal to 50 else false. The Division method must return "First" when the parameter average is 60 or more, "Second" when average is 50 or more but below 60, "No division" when average is less than 50. |
| 25 | Assume that there are two packages, student and exam. A student package contains Student class |

| | |
|---|---|
| | and the exam package contains Result class. Write a program that generates mark sheet for students. |
| 26 | Define a class A in package apack. In class A, three variables are defined of access modifiers protected, private and public. Define class B in package bpack which extends A and write display method which accesses variables of class A.<br>Define class C in package cpack which has one method display() in that create one object of class A and display its variables. Define class ProtectedDemo in package dpack in which write main () method. Create objects of class B and C and class display method for both these objects. |
| 27 | Write a java program to implement lambda expression with functional interface in java |
| 28 | Write a java program to accept string check whether it is in Upper or Lower case. As per case change it in according vise versa. |
| 29 | Write a java program to use important methods of String class. |
| 30 | Write a program in Java to demonstrate use of final class. |
| 31 | Write a program in Java to develop user defined exception for 'Divide by Zero' error |
| 32 | Write a program in Java to demonstrate throw, throws, finally, multiple try block and multiple catch exception. |
| 33 | Write a small application in Java to develop Banking Application in which user deposits the amount Rs 1000.00 and then start withdrawing of Rs 400.00, Rs 300.00 and it throws exception "Not Sufficient Fund" when user withdraws Rs. 500 thereafter. |
| 34 | Write a program to write at least 10 objects of the Circle class in a File using ObjectOutputStream and perform basic operations: adding, retrieving, updating, removing elements**.**<br>[ Use Generic Data types and Collections for the this task] |
| 35 | Write a program for Java Generics and Collections Sorting operations:<br>1. Sorting a list according to natural ordering of elements<br>2. Reversing sort order<br>3. Sorting a list whose elements of a custom type<br>4. Sorting a list using a Comparator |
| 36 | Write a program in Java to create, write, modify, read operations on a Text file. |
| 37 | Write a java program to illustrates use of standard input stream to read the user input. |
| 38 | Write a java program to checks the existence of a specified file. |
| 39 | Write a java program to create a file to the specified location. |
| 40 | Write a java program to demonstrate the way contents are read from a file. |
| 41 | Write a java program to first checks the existence of the specified file. If the file exists, the data is written to the file through the object of the FileOutputStream class. |
| 42 | Write a java program to count the availability of text lines in the particular file. A file is read before counting lines of a particular file. |
| 43 | Write a generic method to count the number of elements in a collection that have a specific property (for example, odd integers, prime numbers, palindromes). |
| 44 | Write a generic method to exchange the positions of two different elements in an array. |
| 45 | Write a generic method to find the maximal element in the range [begin, end) of a list. |
| 46 | Write a program to implement JDBC/ODBC connectivity to data base using java program. |
| 47 | Write a java program to connect any of database e.g. MYSQL / Oracle /MS Access/ etc. |
| 48 | Write a java program to create Employee table(Empno., Name, Designation, Salary) and insert a record in it. |
| 49 | Write a JAVA program to accept the details of student (Rno , SName , Per) from the user and insert it into the table. (use PreparedStatement Class). |
| 50 | Write a Menu driven program in Java for the following. |

| | |
|---|---|
| | 1. Create a Library Table (BookID, ISSNNo., Author, BookTitle, Price, Publisher, Year)<br>2. Insert Record into the Library Table.<br>3. Update The Existing Record.<br>4. Display all the Records from the Table.<br>5. Display names of books stating with “J” character<br>6. Delete the record<br>7. Exit from the program. |
| | |
| | **Desirable : Thread Programming** |
| 1 | Write a program to implement the concept of threading by extending Thread Class |
| 2 | Write a program to implement the concept of threading by implementing Runnable Interface |
| 3. | Write a program that executes two threads. One thread displays “Thread1” every 2,000 milliseconds, and the other displays “Thread2” every 4,000 milliseconds. |
| 4. | Write a program that executes two threads. One thread will print the even numbers and another thread will print odd numbers from 1 to 50. |
| 5. | Write java program that create and runs following threads:<br>i) print "A" 20 times<br>ii) print "B" 30 times<br>iii) print "C" 15 times |
| 6. | Write a program in Java to demonstrate use of synchronization of threads when multiple threads are trying to update common variable |

**Reference: To learn Java language (Online tutorials):**

1. https://docs.oracle.com/en/java
2. http://www.learnjavaonline.org/
3. http://java.sun.com/docs/books/tutorial/index.html
4. http://www.javaworld.com

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
## Year – 2 (Semester – III) (W.E.F. JULY 2018)

## Subject Name: Database Management Systems (DBMS)
## Subject Code: 4639303

1. **Learning Objectives:**

   - To understand the fundamental concepts of Database Management Systems.
   - To understand the concepts necessary for designing, using and implementing database systems and applications

2. **Prerequisites:** Basic knowledge of working with computers.

3. **Contents:**

| Unit No. | Chapter Details | Weightage Percentage |
|---|---|---|
| 1 | **Introduction to Database System**<br><br>**Database and Users:** Introduction( Basic Concepts : Data, Database, Database systems, Database Management Systems), Characteristics of Database Approach, Actors on Scene, Workers behind the Scene, Advantages of using the DBMS approach<br><br>**Database System Concepts and Architecture:** Data Models, Schemas, Instances, the three schema architecture and data independence, Database Languages and interfaces, Database System environment, Centralized and client / Server Architecture for DBMSs, Classifications of Database Management Systems | **10%** |
| 2 | **Entity Relationship Diagram**<br><br>Using high level conceptual data models for database design (Design Phases of database design), Entity types, Entity Sets, Attributes and keys, Relationship Types, Relationship sets, Roles and structural constraints, Weak entity Types, Refining teh ER diagram for company Database, Entity Relationship Diagram Naming conventions and Design issues, Example of other Notation: UML class diagram, Relationship types of degree higher than 2<br><br>Subclasses, Super Classes, Inheritance, Specialization and Generalization<br>Relational Database design by ER and EER to Relational Mapping | **20%** |

| | | |
|---|---|---|
| **3** | **Database Design**<br><br>Informal Design Guidelines for Relational Schema, Functional Dependencies, Normal Forms based on Primary keys, General definitions of 1NF,2NF and 3NF, Boyce-Codd Normal Forms(BCNF), Multi-valued Dependency and Fourth Normal Form | **20%** |
| **4** | **Relational Model**<br><br>**Relational Model concepts:** Relational Model concepts, Relational Model constraints and Relational Database Schemas<br><br>**Relational Algebra:** Unary Relational Operations (Select and Project), Relational Algebra operations from Set Theory, Binary Relational Operations (JOIN and Division) and Additional Relational Operations (Generalized projection, aggregate functions and grouping, Recursive Closure Operations, Outer Join Operations, the outer union operation) | **20%** |
| **5** | **Introduction to Transaction Processing Concepts**<br><br>Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, characterizing Schedule Based on Recoverability and Serializability | **10%** |
| **6** | **Concurrency Control Techniques**<br><br>Two-Phase Locking Techniques for Concurrency Control, Concurrency Control Based on Timestamp Ordering<br><br>Overview of Multi-version Concurrency Control Techniques, Validation (Optimistic) Techniques and Snapshot Isolation Concurrency Control<br><br>**Database Recovery Techniques**<br><br>Recovery Concepts, NO-UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update<br><br>Overview of Shadow Paging | **20%** |

**4. Reference Book(s):**

Ramez Elmsari,Shamkant B Navathe, “Fundamentals of Database Systems”, 7th Edition, Pearson Education

## 5. Suggested Additional Reading:


1. Silberschatz, Korth, Sudarshan, "Database System Concepts", 5th Edition, McGraw Hill Publication.
2. S K Singh, "Database Systems : Concepts, Design and Applications", Pearson Education
3. Peter Rob, Carlos Coronel, "Database Systems : Design, Implementation and Management", Cengage Learning
4. C J Date, A Kannan, S Swaminathan, "An Introduction to Database Systems", 8th Edition, Pearson Education


## 6. Chapter wise Coverage from Main Reference Book(s):

| Unit No. | Text Books | Topics/Subtopics |
|---|---|---|
| 1 | Book-I | Chapter 1( 1.1 to 1.6), Chapter 2 |
| 2 | Book-I | Chapter 3, Chapter4 ( 4.1 & 4.2), Chapter 9 |
| 3 | Book-I | Chapter 14 ( 14.1 to 14.6) |
| 4 | Book-I | Chapter 5 ( 5.1 & 5.2), Chapter 8 |
| 5 | Book-I | Chapter 20 (20.1 to 20.5) |
| 6 | Book-I | Chapter 21 (21.1, 21.2), Overview of 21.3 and 21.4<br>Chapter 22 ( 22.1 to 22.4) |

## 7. Accomplishments of the student after completing the course :

- A student would be able to effectively squeeze the "real world" data into the relational data model of the database system and data retrieval
- Clear understanding for the need of a database.
- Understand the uses the database schema
- Understand the need for normalization
- Use different types of physical implementation of database
- Use database for concurrent use.
- Backup data from database.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
### Year – 2 (Semester – III) (W.E.F. JULY 2018)

## Subject Name: Database Management Systems (DBMS)
## Subject Code: 4639303_Practical List

**A. List of Practical Related to DBMS:**

**Part I: RDBMS (Module Weightage: 100%)**

**Tools:** MySQL
**Topics:**

| | |
|---|---|
| 1 | Introduction to Database: MySQL, Installation, Data Types, |
| **2** | Manage Database: Create Database, Drop Database, Select Database (MySQL> prompt) |
| **3** | Create Table:<br>The Create Table Command,<br>Creating a table from a table (with data, without data, with all columns, with selected columns)<br>Drop Table<br>Alter Table |
| **4** | Study DML Commands ( Select, insert, update, delete ) |
| **5** | Sorting Data, Handling Null values ( IS NULL) |
| **6** | Join |
| **7** | Like Clause , REGEXP |
| **8** | Transaction Control statements: Commit, Rollback |
| **9** | Advanced Concepts: View, Index, Sequences |
| **10** | Database Export / Import |
| **11** | Study single row functions: String functions, Numeric Functions, Date Functions |
| **12** | Study aggregate / group functions |
| **13** | Study subquery concepts |
| **14** | Constraints: Primary Key, Foreign Key, Check, Default, Not Null, Unique |
| **15** | Set Operators |
| **16** | Compound Statement Handling:<br><br>Syntax, Variables, flow of control, condition handling, Cursor Management,<br>Create procedure and Function |
| **17** | Create Triggers |
| **18** | Data dictionary |
| **19** | Security / privileges (Desirable) |

**References:**
1) Steve Suehring, Tim Converse, Joyce Park, PHP 6 and MySQL Bible, Wiley
2) Andrea Tarr, PHP and MySQL 24-Hour Trainer, Wiley

**Web references**:
1) https://dev.mysql.com

**Part II: NoSQL Database (Desirable)**

Tools: MongoDB

| | |
|---|---|
| 1 | Introduction, Installation |
| **2** | Create Database, Drop Database |
| **3** | Create Collection, show collection |
| **4** | Insert document, Query Document, Update document, delete document |
| **5** | Projection |
| **6** | Limiting rows |
| **7** | Export and Import |

https://docs.mongodb.com/manual/mongo/

**Part III: Practical's to perform: (SQL and PL/SQL)**

**SET 1**

DEPARTMENT (dept_no, dept_name, location)

1. Create the Simple DEPARTMENT Table.
2. Display structure of department table.
3. Insert below records into Department Table

| Dept_no | Dept_name | Location |
|---|---|---|
| 10 | Account | NY |
| 20 | HR | NY |
| 30 | Production | DL |
| 40 | Sales | NY |
| 50 | EDP | MU |
| 60 | TRG | |
| 110 | RND | AH |

4. Display all records of Department table
5. Display all department belonging to location 'NY'
6. Display details of Department 10
7. List all department names starting with 'A'
8. List all departments whose number is between 1 and 100
9. Delete 'TRG' department
10. Change department name 'EDP' to 'IT

**SET 2**

EMPLOYEE (emp_id, emp_name, birth_date, gender, dept_no, address, designation, salary, experience, email)
DEPARTMENT (dept_no, dept_name, location)

**Do as directed:**

11. Create the EMP Table with all necessary constraints such as
    In EMP TABLE: Employee id should be primary key, Department no should be

Foreign key, employee age (birth_date) should be greater than 18 years, salary should be greater than zero, email should have (@ and dot) sign in address, designation of employee can be "manager", "clerk", "leader", "analyst", "designer", "coder", "tester".

1. Create DEPT table with neccessary constraint such as
   Department no should be primary key, department name should be unique.
12. After creation of above tables, modify Employee table by adding the constraints as 'Male' or 'Female' in gender field and display the structure.
13. Insert proper data (at least 5 appropriate records) in all the tables.
14. Describe the structure of table created
15. List all records of each table in ascending order.
16. Delete the department whose loction is Ahmedabad.
17. Display female employee list
18. Display Departname wise employee Names
19. Find the names of the employee who has salary less than 5000 and greater than 2000.
20. Display the names and the designation of all female employee in descending order.
21. Display the names of all the employees who names starts with 'A' ends with 'A'.
22. Find the name of employee and salary for those who had obtain minimum salary.
23. Add 10% raise in salary of all employees whose department is 'IT'.
24. Count total number of employees of 'IT' department.
25. List all employees who born in the current month.
26. Print the record of employee and dept table as "Employee works in department 'MBA'.
27. List names of employees who are fresher's (less than 1 year of experience).
**28.** List department wise names of employees who has more than 5 years of experience.
**29.** Crete Sequence to generate department ID
**30.** List department having no employees

**SET 3**

**STUDENT (rollno, name, class, birthdate)**
**COURSE (courseno, coursename, max_marks, pass_marks)**
**SC (rollno, courseno, marks)**

1. Create the above three tables along with key constraints.
2. Write an Insert script for insertion of rows with substitution variables and insertappropriate data.
3. Add a constraint that the marks entered should strictly be between 0 and 100.
4. While creating SC table, composite key constraint was forgotten. Add the composite keynow.
5. Display details of student who takes 'Database Management System' course.
6. Display the names of students who have scored more than 70% in Computer Networksand have not failed in any subject.
7. Display the average marks obtained by each student.
8. Select all courses where passing marks are more than 30% of average maximum mark.
9. Display details of students who are born in 1980 or 1982.
10. Create a view that displays student courseno and its corresponding marks.

**SET 4**

Create the database COMPANY and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

EMPLOYEE (emp_id, emp_name, birth_date, gender, dept_no, address, designation, salary, experience, email)
DEPART (dept_no, dept_name, total_employees, location)
PROJECT (proj_id, type_of_project, status, start_date, emp_id)

Insert proper data (at least 5 appropriate records) in all the tables.

**Do as directed:**

1. Delete the department whose total number of employees less than 1.
2. Display the names and the designation of all female employee in descending order.
3. Display the names of all the employees who names starts with 'A' ends with 'A'.
4. Find the name of employee and salary for those who had obtain minimum salary.
5. Add 10% raise in salary of all employees whose department is 'CIVIL'.
6. Count total number of employees of 'MCA' department.
7. List all employees who born in the current month.
8. Print the record of employee and dept table as "Employee works in department 'CE'.
9. List names of employees who are fresher's(less than 1 year of experience).
10. List department wise names of employees who has more than 5 years of experience.
11. Write a function which will display total number of projects based on status (pass status as parameter).
12. Write a procedure that will display list of projects which is going to start today.
13. Write a trigger which do not allow insertion/updation/deletion into Project table if status type is 'pending'

**SET 5**

Create the database STUD and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

HOSTEL (HNO, HNAME, HADDR, TOTAL_CAPACITY, WARDEN)

ROOM (HNO, RNO, RTYPE, LOCATION, NO_OF_STUDENTS, STATUS)

CHARGES (HNO, RTYPE, CHARGES)

STUDENT (SID, SNAME, MOBILE-NO, GENDER, FACULTY, DEPT, CLASS, HNO, RNO)

FEES (SID, FDATE, FAMOUNT)

The STATUS field tells us whether the room is occupied or vacant. The charges represent the term fees to be paid half yearly. A student can pay either the annual fees at one time or the half yearly fees twice a year.

Insert proper data (at least 5 appropriate records) in all the tables.

**Do as directed:**

1. Display the total number of rooms that are presently vacant.
2. Display number of students of each faculty and department wise staying in each hostel.
3. Display hostels, which have at least one single-seated room.
4. Display the warden name and hostel address of students of Computer Science department.
5. Display those hostel details where single seated or double-seated rooms are vacant.
6. Display details of hostels occupied by medical students.
7. Display hostels, which are totally occupied to its fullest capacity.
8. List details about students who are staying in the double-seated rooms of Chanakya Hostel.

9. Display the total number of students staying in each room type of each hostel.
10. Display details about students who have paid fees in the month of Nov. 2017.
11. For those hostels where total capacity is more than 300, display details of students studying in Science faculty.
12. Display hostel details where there are at least 10 vacant rooms.
13. Display details of students who have still not paid fees.
14. Display those hostels where single-seated room is the costliest.
15. Write a trigger which do not allow to insert or update student record if mobile_no length is less than 10 digits.
16. Write a PL/SQL block which will count total number of student's gender wise.
    Male Students: 999 students
    Female Students: 999 students

**SET 6**

Create the database HOSPITAL and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

DOCTOR (DNO, DNAME, SPECIALIZATION, CLINIC_ADDR)
MEDICINE (MNO, MNAME, TYPE, CONTENT, MANUFACTURER)
DISEASE (DISEASE_NAME, SYMPTOM1, SYMPTOM2, SYMPTOM3)
TREATMENT (TNO, DNO, DISEASE_NAME, MNO, DOSAGE, AVG_CURE_TIME)

Insert proper data (at least 5 appropriate records) in all the tables.

**Do as directed:**

1. Display records of each table in ascending order.
2. Count total number of doctors which has not given any treatment.
3. Display all Chennai doctors who treat cancer.
4. Remove disease "polio" from disease table as well as treatment table.
5. Delete all those treatment related to liver of Dr.Shah.
6. Create index on dno, Disease name in the treatment table.
7. Display details of doctors who treat migraines.
8. What is the maximum dosage of "penicillin" prescribe by the doctor for the treatment of any disease?
9. Display total number of disease treated by every doctor.
10. Which doctor have no treatment for "depression"?
11. Create a view which contains the treatment and doctors details. Make sure that no body is allowed to modify any detail in the view.
12. Write a PL/SQL block to print the following report ( Symptoms wise print total number of medicine given )

| Medicine | Symptom1 | Symptom2 | Symptom3 | Total |
|---|---|---|---|---|
| M1 | 999 | 999 | 999 | 9999 |
| M2 | 999 | 999 | 999 | 9999 |
| M3 | 999 | 999 | 999 | 9999 |

13. Write a trigger which does not allow to insert or update treatment table if AVG_CURE_TIME is less than 1.

**SET 7**

Create the database SHOPPING and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

CUSTOMER (cno, cust_name, cust_phone, location,gender)
ITEM (itemno, itemname, color, weight, expire_date, price, shop_name)
CUST_ITEM (cno, itemno, quantity_purchased, date_purchase)

Insert proper data (at least 5 appropriate records) in all the tables.

**Do as directed:**

1. Delete the items whose price is more than 50000. .
2. Find the names of the customer who is located in same location as that of other customer.
3. Display the names of items which is black, white & brown in color.
4. Display the names of all the items whose names lies between 'p' and's'.
5. Find the item which is having less weight.
6. Add one month more to those items whose item no =40.
7. Count total number of items which is going to expire in next month
8. List all customers whose phone number starts with '99'.
9. Display total value (qty*price) for all items.
10. List customer details who has purchased maximum number of items
11. Display total price item wise.
12. List name of items, customer details and qty purchased.
13. Write a PL/SQL procedure which will display records in the following format

Today's Date: _____ Shop name: _____

| Item number | Item name | Expire date | Quantity | Price | Total Rs. |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |

Grand Total Rs. _____

14. Write a trigger which do not allow insertion / updation / deletion of Item details on Sunday.

**SET 8**

Create the database THEATRE and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**SCREEN (SCREEN_ID, LOCATION, SEATING_CAP)**
**MOVIE (MOVIE_ID, MOVIE_NAME, DATE_OF_RELEASE)**
**CURRENT (SCREEN_ID, MOVIE_ID, DATE_OF_ARRIVAL, DATE_OF_CLOSURE)**

**Check Constraints:**
Value of screen_id must start with letters 'S'.
Attribute location can be any one of 'FF', 'SF', or 'TF'.

**Do as directed:**

1. Get the name of movie which has run the longest in the multiplex so far.
2. Get the average duration of a movie on screen number 'S4'.
3. Get the details of movie that closed on date 24-november-2004.
4. Movie 'star wars III 'was released in the 7th week of 2005. Find out the date of its release considering that a movie releases only on Friday.
5. Get the full outer join of the relations screen and current.
6. Write a PL/SQL function which will count total number of day's horror movie last longer.
7. Write a PL/SQL procedure that will display movie which is going to release today.
8. Write a trigger which will not allow to insert/update in current table if Date_of_arrival is less than date_of_closure.

---

**SET 9**

Create the database EXAM and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**APPLICANT (AID, ANAME, ADDR, ABIRTH_DT)**

**ENTRANCE_TEST (ETID, ETNAME, MAX_SCORE, CUT_SCORE)**

**ETEST_CENTRE (ETCID, LOCATION, INCHARGE, CAPACITY)**

**ETEST_DETAILS (AID, ETID, ETCID, ETEST_DT, SCORE)**

(This database is for a common entrance test which is being conducted at a number of centers and can be taken by an applicant on any day except holidays)

**Do as directed:**

1. Modify the APPLICANT table so that every applicant id has an 'A' before its value. E.g. if value is '1123', it should become 'A1123'.
2. Display test center details where no tests were conducted.

3. Display details about applicants who have the same score as that of Ajaykumar in 'ORACLE FUNDAMENTALS'.
4. Display details of applicants who appeared for all tests.
5. Display those tests where no applicant has failed.
6. Display details of entrance test centers which had full attendance between 1st Oct 15 and 15th Oct 16.
7. Display details of the applicants who scored more than the cut score in the tests they appeared in.
8. Display average and maximum score test wise of tests conducted at Mumbai.
9. Display the number of applicants who have appeared for each test, test center wise.
10. Display details about test centers where no tests have been conducted.
11. For tests, which have been conducted between 2-3-17 and 23-4-17, show details of the tests as well as the test centre.
12. How many applicants appeared in the 'ORACLE FUNDAMENTALS' test at Chennai in the month of February?
13. Display details about applicants who appeared for tests in the same month as the month in which they were born.
14. Display the details about APPLICANTS who have scored the highest in each test, test centre wise.
15. Design a read only view, which has details about applicants and the tests that he has appeared for.
16. Write a procedure which will print maximum score centre wise.
17. Write a procedure which will print details of entrance test.

---

Centre name: _____ candidate id: ____ date: ___ score: _____

---

15. Write a trigger which do not allow insertion / updation / deletion of Enterance test details on Sunday.

**SET 10**

Create the database BUS TRANSPORT and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**CATEGORY (CAT_CODE, CATDESC)**
**ROUTEMASTER (ROUTENO, ORIGIN, DESTINATION, FARE, DISTANCE, CAPACITY, DAY, CAT_CODE)**
**TICKETHEADER (TICKETNO, DATEOFISSUE, DATEOFTRAVEL, BOARDPLACE, ROUTENO)**

**TICKET DETAILS (TICKETNO, NAME, SEX, AGE, FARE)**

ADD THE FOLLOWING CONSTRAINTS:

1. DELUXE, SUPERDELUXE, SUPERFAST AND NORMAL ARE THE CATDESC
2. ORIGIN AND DESTINATION CANNOT BE SAME,
3. CAPACITY SHOULD BE>0 AND <=60

Do as directed:

1. Display the total number of people traveled on each ticket group by ticket no 23.
2. Give the total collection of fare for each route.
3. Give the number of months between issue date and travel date of each ticket issued.
4. Count number of person boarding from the same place and same route.
5. Display count of person who has traveled in each category.
6. Write a trigger which allow to insert or update the bus capacity only greater than zero and less than 60.
7. Write a Procedure which will print tour details, a driver is going to take it. ( pass route_no as parameter)

   Route No: _____ Vehicle number: _____

   Transport Details

| Source | Destination | Start date | Total days | Capacity |
|---|---|---|---|---|
| Xxx | Xxx | Xxx | 999 | 999 |

**SET 11**

Create the database BUS TRANSPORT and create given tables with all necessary constraints such as primary key, foreign key, unique key, not null and check constraints.

**TRAIN _MASTER:**

| FIELD NAME | DATA TYPE | CONSTRAINTS |
|---|---|---|
| TRAIN NUMBER | VARCHAR2(6) | PRIMARY KEY AND LAST TWO CHARS SHOULD BE 'DN' OR 'UP' |
| TRAIN NAME | VARCHAR2(25) | NOT NULL |
| ARRIVAL TIME | DATE | NOT NULL |
| DEPARTURE TIME | DATE | NOT NULL |
| NO OF HOURS | NUMBER(5,2) | NOT NULL |
| SOURCE STATION | VARCHAR2(25) | NOT NULL |
| END STATION | VHARCHAR2(25) | NOT NULL |

**PASSENGER_DETAILS:**

| FIELD NAME | DATA TYPE | CONSTRAINTS |
|---|---|---|
| TICKET NUMBER | NUMBER(5) | |
| TRAIN NUMBER | VARCHAR2(6) | FOREIGN KEY REFERENCE TO TRAIN_MASTER |

| | | |
|---|---|---|
| | | THIS RELATED RECORD SHOULD BE DELETED IF MASTER RECORD IS DELETED. |
| SEAT NUMBER | NUMBER(2) | NOT NULL |
| PASSENGER NAME | VARCHAR2(35) | NOT NULL |
| AGE | NUMBER(2) | NOT NULL |
| GENDER | CHAR(1) | SHOULD BE 'M' FOR MALE OR 'F' FOR FEMALE |
| TRAVEL DATE | DATE | |
| CLASS | VARCHAR2(4) | SHOULD BE IN (IA, IIA, IIIA, IC, II) |

**TRAIN_SEAT_MASTER:**

| FIELD NAME | DATA TYPE | CONSTRAINTS |
|---|---|---|
| TRAIN NUMBER | VARCHAR2(6) | FOREIGN KEY REFERENCE TO TRAIN_MASTER THIS RELATED RECORD SHOULD BE DELETED IF MASTER RECORD IS DELETED. |
| CLASS | VARCHAR2(4) | SHOULD BE IN (IA, IIA, IIIA, IC, II) |
| TOTAL SEATS | NUMBER(2) | SHOULD BE >= 25 AND <= 90 |

**TRAIN_DAY_MASTER:**

| FIELD NAME | DATA TYPE | CONSTRAINTS |
|---|---|---|
| TRAIN NUMBER | VARCHAR2(6) | FOREIGN KEY REFERENCE TO TRAIN_MASTER THIS RELATED RECORD SHOULD BE DELETED IF MASTER RECORD IS DELETED. |
| DAY | VARCHAR2(3) | VALUE SHOULD BE IN 'MON' … TO … 'SUN' |

Do as directed:

1. Give all the train nanes starting from “Bombay” and going to “Ahmedabad” on Tuesday and Wednesday.
2. List all trains which is available on Sunday.
3. Give classwise seat availability on 10-June-2018 for train 9012DN.
4. List total seats classwise for train running on thrusday.
5. List train names which have no sleeper class.
6. List train number which run on Monday during 8:00: am to 1:00pm.
7. Write a procedure which will print all train details going from Baroda to Banglore.
8. Write a function which will print arrival time and departure time for a given train. ( pass train no as a parameter)
9. Write a trigger which do not allow to insert or update passenger record if age is greater than 100.

**SET 12**

CUSTOMER(cid, fname, lname, city, country, phone)

ORDER (oid, oDate, oNumber, cid, oTotalAmount)

1. List the number of customers in each country. Only include countries with more than 100 customers.
2. List the number of customers in each country, except China, sorted high to low. Only include countries with 5 or more customers.
3. List all customers with average orders between Rs.5000 and Rs.6500.
4. Create a trigger that executes whenever country is updated in CUSTOMER table.
5. Create a function to return customer with maximum orders.
6. Create a procedure to display month names of dates of ORDER table. The month names should be unique.

**SET 13**

**DISTRIBUTOR (dno, dname, daddress, dphone)**
**ITEM (itemno, itemname, colour, weight)**
**DIST_ITEM (dno, itemno, qty)**

1. Add a column CONTACT_PERSON to the DISTRIBUTOR table with the not null constraint.
2. Create a view LONDON_DIST on DIST_ITEM which contains only those records where distributors are from London. Make sure that this condition is checked for every DML against this view.
3. Display the details of all those items that have never been supplied.
4. Delete all those items that have been supplied only once.
5. List the names of distributors who have an 'A' and also a 'B' somewhere in their names.
6. Count the number of items having the same colour but not having weight between 20 and 100.
7. Display all those distributors who have supplied more than 1000 parts of the same type.
8. Display the average weight of items of the same colour provided at least three items have thatcolour.
9. Display the position where a distributor name has an 'OH' in its spelling somewhere after the fourth character.
10. Count the number of distributors who have a phone connection and are supplying item number 'I100'.
11. Create a view on the tables in such a way that the view contains the distributor name, item name and the quantity supplied.
12. List the name, address and phone number of distributors who have the same three digits in their number as 'Mr. Talkative'.
13. List all distributor names who supply either item I1 or I7 or the quantity supplied is more than 100.
14. Display the data of the top three heaviest ITEMS.

**SET 14**

a) Write a PLSQL block which will print Employee list (Empno and Name)
EMP (empno, empnm, empadd, salary, date_birth, joindt, deptno)

b) Write a function that returns total number of incomplete jobs, using table
JOB (jobid, type_of_job, status)

c) Write a function which displays the number of items whose weight fall between a given ranges for a particular color using table ITEM (itemno, name, color, weight)

d) Write a procedure to display top five highest paid workers who are specialized in "PAINTING" using table WORKER (workerid, name, wage_per_hour, specialized_in, manager_id)

**SET 15**

EMP (empno, empnm, empadd, salary, date_birth, joindt, deptno)
DEPT (deptno, deptnm)

Write a PL/SQL block (table above EMP-DEPT table) which takes as input Department name and displays all the employees of this department who has been working since last five years

**SET 16**

EMPMAST (empno, name, pfno, empbasic, deptno, designation)
DEPT (DNO, DNAME)

Rules: HRA = 15% of basic
DA = 50% of basic
Medical = 100
PF = 8.33%of basic

Print Salary slip. Design your own format

**SET 17**

**Consider the Bank schema as**
**ACCOUNT (AC_NO, NAME, AC_TYPE, BALANCE_AMT, BALANCE_DATE)**
**TRANSACTION (AC_NO, DATE, TR_TYPE, AMOUNT, PREV_BALANCE, REMARK)**
Note: 1. AC_type may be S for saving or C for current, 2. TR_type may be D for deposit or W for withdrawal.
a. Write a procedure to print the Bank Transaction details by passing from and to dates.

**SET 18**

**BRANCH** (branch_no, area, city)
**MEMBERS** (mno, name branch_no, salary, manager_no)
Note: Manager can be from one of the members.
1. Write a procedure which list the name of members who earns more than that of his managers.
2. Write a procedure which gives details of employee having maximum salary branch wise.

**SET 19**

Employee (eid, fname, lname, salary)

1. Use a Cursor for Loop inside a function to calculate and return total paid salary to all employees by the company.

2. Modify the function created above to become a procedure and display the total paid salary from the procedure itself. Instead of calculating for all employees, calculate only for those employees whose name starts from a character passed as parameter to the procedure and hence to the cursor.

**SET 20**

Consider the DUAL and data dictionary tables/views to solve the following Queries.

1. Find out the names of all the tables, views and constraints associated with current tables in the system.
2. Write a query to add 15 days to the current date.
3. Write a query to Add and subtract 5 months from the current month.
4. Find out the ASCII equivalent of character 'M'.
5. Find out the character equivalent of ASCII 67, 65 and 84.
6. Write a query to find the last day of the month.
7. Find out how many days are left in the current month.
8. Write a query to calculate the Date difference between current date and 20/05/2015.
9. Write a query to Calculate the number of months between current date and 03/03/2016.
10. Find out the second occurrence of 'or' from third position in the string 'corporate floor '.
11. Find out log to the base 3 of 81.
12. Convert the string 'gujarat technological university' so that first character of each work is in capital.
13. Convert the string 'jack and jue' Into 'black and blue'.
14. Round off the date 27-July-2016 to the current year.
15. Find out the user name and user id off currently logged on user.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
## Year – 2 (Semester – III) (W.E.F. JULY 2018)

## Subject Name: Programming in Python
## Subject Code: 4639304

1. **Learning Objectives:**
   - To develop proficiency in creating based applications using the Python Programming Language.
   - To be able to understand the various data structures available in Python programming language and apply them in solving computational problems.
   - To be able to draw various kinds of plots using PyLab and Pandas
   - To be able to understand the creation DB API in Python
   - To be able to understand the applications of advanced concepts like networking, multithreading and data science in python

2. **Prerequisites**: Knowledge of some programming Concepts

3. **Contents:**

| Unit No. | Course Content | Weightage Percentage |
|---|---|---|
| 1 | **Introduction to Python:**<br><br>The basic elements of Python, Objects, expressions and numerical Types, Variables and assignments, IDLE, Branching programs, Strings and Input, Iteration<br><br>**Structured Types, Mutability and Higher-order Functions:**<br><br>Tuples, Lists and Mutability, Functions as Objects, Strings, Tuples and Lists, Dictionaries | 10% |
| 2 | **Functions, Exception, Modules and Files**<br><br>**Functions:** Difference between a Function and a Method, Defining a Function, Calling a Function, Returning Results from a Function, Returning Multiple Values from a Function, Functions are First Class Objects, Pass by Object Reference, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Local and Global Variables, The Global Keyword, Passing a Group of Elements to a Function, Recursive Functions, Anonymous Functions or Lambdas (Using Lambdas with filter() Function, Using Lambdas with map() Function, Using Lambdas with reduce() Function), Function Decorators, Generators, Structured Programming, Creating our Own Modules in Python, The Special Variable __name__<br><br>**Exceptions:** Errors in a Python Program (Compile-Time Errors, Runtime Errors, Logical Errors),Exceptions, Exception Handling, Types of Exceptions, The Except Block, The assert Statement, User-Defined Exceptions, Logging the Exceptions | 20% |

| | | |
|---|---|---|
| | **Files**: Files, Types of Files in Python, Opening a File, Closing a File, Working with Text Files Containing Strings, Knowing Whether a File Exists or Not, Working with Binary Files, The with Statement, Pickle in Python, The seek() and tell() Methods, Random Accessing of Binary Files, Random Accessing of Binary Files using mmap, Zipping and Unzipping Files, Working with Directories, Running Other Programs from Python Program | |
| 3 | **Classes and Object-oriented Programming:**<br><br>**Classes:** Creating a Class, The Self Variable, Constructor, Types of Variables, Namespaces, Types of Methods (Instance Methods, Class Methods, Static Methods), Passing Members of One Class to Another Class, Inner Classes<br><br>**Inheritance and Polymorphism:** Constructors in Inheritance, Overriding Super Class Constructors and Methods, The super() Method, Types of Inheritance, Single Inheritance, Multiple Inheritance, Method Resolution Order (MRO), Polymorphism, Duck Typing Philosophy of Python, Operator Overloading, Method Overloading, Method Overriding<br><br>**Abstract Classes and Interfaces**: Abstract Method and Abstract Class, Interfaces in Python, Abstract Classes vs. Interfaces, | 15% |
| 4 | **Advanced Topics I: Plotting and Data Science**<br><br>Plotting using PyLab, Plotting mortgages and extended examples<br><br>**Data Science Using Python:** Data Frame (Creating Data Frame from an Excel Spreadsheet, Creating Data Frame from .csv Files, Creating Data Frame from a Python Dictionary, Creating Data from Python List of Tuples, Operations on Data Frames),<br><br>**Data Visualization** : Bar Graph, Histogram, Creating a Pie Chart, Creating Line Graph | 15% |
| 5 | **Advanced Topics II: Regular Expressions**<br><br>**REs and Python:** Regular Expressions, Sequence Characters in Regular Expressions, Quantifiers in Regular Expressions, Special Characters in Regular Expressions, Using Regular Expressions on Files, Retrieving Information from a HTML File | 10% |
| | **Threading :** Concurrent Programming and GIL, Uses of Threads, Creating Threads in Python, Thread Class Methods, Single Tasking using a Thread, Multitasking using Multiple Threads, Thread Synchronization Deadlock of Threads, Avoiding Deadlocks in a Program, Communication between Threads, Thread Communication using notify() and wait() Methods, Thread Communication using a Queue, Daemon Threads | 10% |
| | **Networking:** Protocol, Sockets, Knowing IP Address, URL, Reading the Source Code of a Web Page, Downloading a Web Page from Internet, Downloading an Image from Internet, A TCP/IP Server, A TCP/IP Client, A UDP Server, A UDP Client, File Server, File Client, Two-Way Communication between Server and Client, Sending a Simple Mail, | 5% |

| | | |
|---|---|---|
| 6 | **Python's Database Connectivity**<br><br>Verifying the MySQLdb Interface Installation, Working with MySQL Database, Using MySQL from Python, Retrieving All Rows from a Table, Inserting Rows into a Table, Deleting Rows from a Table, Updating Rows in a Table, Creating Database Tables through Python | 15% |

**4. Text Book:**

1) John V Guttag. "Introduction to Computation and Programming Using Python", Prentice Hall of India
2) R Nageswara Rao, Core Python Programming, 2nd Edition, Dreamtech Press

**5. Reference Books:**

1) Wesley J Chun, Core Python Applications Programming, 3rd Edition.Pearson
2) Luke Sneeringer, Professional Python, WROX
3) Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python, Pearson
4) Doug Hellmann, The python 3 standard Library by example, Pearson Education
5) Alex Martelli, Python Cookbook, O'REILLY
6) Laura Cassell, Python Projects, WROX
7) Daniel Y Chen, Pandas for Everyone: Python Data Analysis , 1st Edition, Pearson Eduction

**Web References:**

1) Charles Severance, Python for informatics: www.pythonlearn.com
2) Swaroop C H. "A Byte of Python", http://www.swaroopch.com/notes/python
3) "Python Programming", http://en.wikibooks.org/wiki/Python_Programming
4) "The Python Tutorial", http://docs.python.org/release/3.0.1/tutorial/
5) "Learn Python the Hard way", http://learnpythonthehardway.org/
6) Dive Into Python 3: http://www.diveintopython.net/

**6. Chapter wise Coverage from Text Book:**

| Unit # | Book# | Chapter |
|---|---|---|
| 1 | 1 | 2,5 |
| 2 | 2 | 9,16,17 |
| 3 | 2 | 13,14,15 |
| 4 | 1 | 11 |
| | 2 | 25 |
| 5 | 2 | Chapter: 18,<br>Chapter: 21(Page 542 to 563),<br>Chapter: 23 |
| 6 | 2 | Chapter 24 ( Page 663 to 681 ) |

**7. Accomplishments of the student after completing the course:**

- Ability to create roboust applications using the Python programming language
- Ability to test and debug applications written using the Python programming language
- Ability to create applications for solving computational problems using the Python Programming Language.

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
### Year – 2 (Semester – III) (W.E.F. JULY 2018)

**Subject Name: Programming in Python**
**Subject Code: 4639304_Practical List**

**A. List of Practical Related to Python:**

**Part I:**

| | |
|---|---|
| 1 | Write a Python Program to Convert Celsius to Fahrenheit and vice –a-versa. |
| **2** | Write a program in python to swap two variables without using temporary variable. |
| **3** | Write a Python Program to Convert Decimal to Binary, Octal and Hexadecimal |
| **4** | Write a program to make a simple calculator (using functions). |
| **5** | Write a program in python to find out maximum and minimum number out of three user entered number. |
| **6** | Write a program which will allow user to enter 10 numbers and display largest odd number from them. It will display appropriate message in case if no odd number is found. |
| **7** | Write a Python program to check if the number provided by the user is an Armstrong number. |
| **8** | Write a Python program to check if the number provided by the user is a palindrome or not. |
| **9** | Write a Python program to perform following operation on given string input:<br>a) Count Number of Vowel in given string<br>b) Count Length of string (donot use len() )<br>c) Reverse string<br>d) Find and replace operation<br>e) check whether string entered is a palindrome or not |
| **10** | Define a procedure histogram() that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:<br>****<br>*********<br>******* |
| **11** | Write a program in python to implement Fibonacci series up to user entered number. (Use recursive Function) |
| **12** | Write a program in python to implement Factorial series up to user entered number. (Use recursive Function) |
| | Write a program in python to implement simple interest and compound interest values on chart using PyLab. Show the difference between both. (Note: Use of object oriented paradigm is compulsory.) |
| **13** | Write a program in Python to implement readline, readlines, write line and writelines file handling mechanisms. |
| **14** | Write a program in python to implement Salary printing file read operation. (File format: EmployeeNo, name, deptno, basic, DA, HRA, Conveyance) should perform below operations.<br>a) Print Salary Slip for given Employee Number<br>b) Print Employee List for Given Department Number |
| **15** | Write a program in python to implement Railway Reservation System using file handling technique. System should perform below operations.<br>a. Reserve a ticket for a passenger.<br>b. List information all reservations done for today's trains. |

| | |
|---|---|
| | (Note: Use of object oriented paradigm is compulsory.) |
| **16** | Write a program in python to implement Library Management System using file handling technique. System should perform below operations.<br>a. Issue a book for student.<br>b. List information today's issued books.<br>(Note: Use of object oriented paradigm is compulsory.) |
| **17** | Write a program in python to implement Bank System using Class and Methods and perform below operations. (Note: Use of object oriented paradigm is compulsory.)<br>a) Add Bank account<br>b) Deposit of money<br>c) withdrawal operation<br>d) Money transfer<br>e) Show Balance |
| **18** | A Python program to display employee id numbers on X-axis and their salaries on Y-axis in the form a bar graph. |
| **19** | A program to display a histogram showing the number of employees in specific age groups. |
| **20** | A program to display a pie chart showing the percentage of employees in each department of a company. |
| **21** | A program to create a line graph to show the profits of a company in various years. |
| **22** | A program to create a line graph to show the profits of a company in various years. |

**Part II: Advanced Topic: Regular Expression**

| | |
|---|---|
| 1 | Create Regular Expressions that |
| | a) Recongnize following strings bit, but, bat, hit, hat or hut |
| | b) Match any pair of words separated by a single space, that is, first and last names. |
| | c) Match any word and single letter separated by a comma and single space, as in last name, first initial. |
| | d) Match simple Web domain names that begin with www and end with a ".com" suffix; for example, www.yahoo.com. Extra Credit: If your regex also supports other high-level domain names, such as .edu, .net, etc. (for example: www.foothill.edu). |
| | e) Match a street address according to your local format (keep your regex general enough to match any number of street words, including the type designation). For example, American street addresses use the format: 1180 Bordeaux Drive. Make your regex flexible enough to support multi-word street names such as: 3120 De la Cruz Boulevard. |
| | |
| **2** | Create utility script to process telephone numbers such that |
| | a. Area codes (the first set of three-digits and the accompanying hyphen) are optional, that is, your regex should match both 800-555-1212 as well as just 555-1212. |
| | b. Either parenthesized or hyphenated area codes are supported, not to mention optional; make your regex match 800-555-1212, 555-1212, and also (800) 555-1212. |
| **3** | Chapter End Practical List of Main Text Book |

**Part III: Database**

| | |
|---|---|
| **1** | Create Web Database Application "Address Book" with options to<br><br>a) add/ insert a record ,<br>b) modify a record ,<br>c) display a record<br>d) delete a record |

| | |
|---|---|
| **2** | A Python program to retrieve all rows from employee table and display thecolumn values in tabular form. |
| **3** | A program to read CSV file and upload data into table |
| **4** | A program to retrieve all rows from employee table and dump into CSV file. |

**Part IV: Thread and Networking (Desirable)**

| | |
|---|---|
| **1** | A Python program to find the currently running thread in a Python program. |
| **2** | A Python program to create a thread and use it to run a function. |
| **3** | A program where two threads are acting on the same method to allot a berth for the passenger. |
| **4** | A Python program to find the IP Address of a website. |
| **5** | A Python program that reads the source code of a Web page. |
| **6** | A Python program to download a Web page from Internet and save it into our computer. |
| **7** | A Python program to create a basic chat server program in Python. |
| **8** | Creating a basic chat client program in Python. |
| **9** | A Python program to create a TCP/IP server program that sends messages to a client. |
| **10** | A Python program to create TCP/IP client program that receives messages from the server. |
| **11** | A Python program to create a UDP server that sends messages to the client. |
| **12** | A Python program to create a UDP client that receives messages from the server. |

# GUJARAT TECHNOLOGICAL UNIVERSITY
## MASTERS IN COMPUTER APPLICATION
### Year – 2 (Semester – III) (W.E.F. JULY 2018)

**Subject Name: Project (LAMP)**
**Subject Code: 4639305**

**1. Learning Objectives:**

- To be able to develop web application using open source technologies
- To learn PHP scripting language and deploying application on Apache Web Server
- To learn Apache Web Server configuration
- To learn MySQL database deployment for web applications

**2. Knowledge of HTML, JavaScript, CSS and Database concepts is desirable**

**3. General Guidelines for Web Project using Open source Technologies**

- The project should be free from plagiarism of any kind.
- It is mandatory that the project should be developed using Apache, MySQL and PHP, on Linux or Windows Platform. Linux is the preferred platform
- Group size : 2-3 Persons per groupDocumentation is not compulsory
- This may not be a live project
- Use of a database is mandatory
- Though use of AJAX and XML is not mandatory, their proper use will be appreciated
- Use of any popular PHP based frameworks like JOOMLA, DRUPAL, etc. is not prohibited, in which case the framework internals should also be known.

**4. Knowledge about the following is expected to be demonstrated.**

- Proper knowledge about the purpose of the application
- Apache web server configuration, use of configuration file httpd.conf
- Mysql Database Server configuration using my.cnf file
- Use of php.ini
- PHP flow-control, functions, arrays and Objects
- Use of String formatting and Date Time functions in PHP
- Use of Forms, Cookies and User sessions in PHP
- Using text and image files in PHP
- Using database connection from PHP

5. **Minimum Expectations:** Application must include CRUD, Master/Detail form, Transactions, Application Reports, Search (with Pagination).

6. **Recommended Book(s):**

   1) Julie C Meloni, "Sams Teach Yourself PHP, MySQL and Apache All in One" 4th edition, Pearson Education
   2) James Lee and Brent Ware, "Open source web development with LAMP" , Pearson Education
   3) Jason Gerner, Morgan Owens, Elizabeth Naramore, Matt Warden, "Professional LAMP: Linux, Apache, MySQL and PHP5 Web Development" WROX publication


7. **Evaluation**

| Sr. No. | Particulars | Weightage |
|---|---|---|
| 1 | Project | 40% |
| 2 | Code Changes | 40% |
| 3 | VIVA | 20% |

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Computer Networking (CN)**
**Subject Code: 4649301**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To equip the students with basics concept of Computer Networks.
- To familiarize the students with the standard models for the layered approach OSI and TCP/IP to communication between machines in a network and the protocols of the various layers.
- To gain basic insight of programming for network solutions.

**2. Prerequisites:** Basic Concept of Data Structures, Operating systems, Programming Language and Knowledge of Linux OS.

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction:** Personal Area Network, Local Area Networks, Metropolitan Area Networks, Wide Area Networks, Internetworks, Network software, protocol hierarchies, Design issues for the layers, connection oriented vs. Connectionless service, service primitives, relationship of services in protocols, Reference Models, Open System Interconnection (OSI), TCP/IP Reference models, Example Networks: ARPANET, NSFNET, Architecture of Internet. | **15%** |
| **Unit II** | **Physical Layer:** Guided Transmission Media, Magnetic Media, Twisted Pairs, Coaxial Cable, Power Lines, Fiber Optics, Wireless Transmission, Electromagnetic Spectrum, Radio Transmission, Microwave Transmission, Infrared Transmission, Light Transmission, Digital Modulation and Multiplexing, Public Switched Telephone Networks. | **15%** |
| **Unit III** | **Data Link Layer:** Design issues, Error detection and correction. Elementary data link protocols: Utopian simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel. Sliding Window protocols, Example data link protocols. | **15%** |
| **Unit IV** | **Medium Access Control Sublayer:** The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, IEEE 802.11 Architecture and Protocol Stack, Physical layer, sublayer, Frame Structure, Services, Data link layer switching. | **15%** |
| **Unit V** | **Network Layer:** Design issues, Routing algorithms: Optimality principle, shortest path routing, Flooding, distance vector routing,Link State routing, Congestion Control Algorithms, The Network layer in the Internet. | **15%** |
| **Unit VI** | **Transport Layer:** Transport Services, Elements of Transport protocols, Connection establishment, connection release, Error control, flow control, congestion control, UDP and TCP protocols. | **15%** |
| **Unit VII** | **Application Layer:** Domain name system, Electronic Mail; the World Wide Web, HTTP | **10%** |

**4. Text Book:**

**1)** Andrew S Tanenbaum, David. J. Wetherall, "Computer Networks", Pearson Education, 5th Edition,

**5. Reference Books:**

**1)** Bhushan H Trivedi ,"Computer Networks", Oxford University Press
**2)** Behrouz A. Forouzan, "Data Communications and Networking", Tata McGraw-Hill, Fourth Edition
**3)** Kurose and Ross,Computer Networking- A Top-Down approach, Pearson, 5th edition
**4)** Larry L. Peterson, Bruce S. Davie, "Computer Networks: A Systems Approach", Morgan Kaufmann Publishers, Fifth Edition, 2011.
**5)** Fred Halsall,Computer Networking and the Internet, Addison Wesley, (5th edition)
**6)** Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, "Computer Networks: An Open Source Approach", Mc Graw Hill Publisher, 2011

**6. Chapter wise Coverage from the Text Book:**

| Unit # | Chapter |
|---|---|
| I | Chapter 1: 2,3,4.1 to 4.4,5.1 |
| II | Chapter 2: 2,3,5,6 |
| III | Chapter 3: 1,2,3,4,5 |
| IV | Chapter 4: 1,2,3.1 to 3.7,4,5,6 |
| V | Chapter 5: 1,2.1 to 2.5,3,6.1 to 6.4 |
| VI | Chapter 6: 1.1,1.2, 2.1 to 2.4, 3.1 to 3.2, 4.1, 4.2,5.1 to 5.9 |
| VII | Chapter 7: 1,2,3.1, 3.4 |

Students are not required to reproduce the entire algorithms/protocol code in the theory exam for any protocols and routing algorithms. Concepts based on these algorithms/ protocols should be asked in theory exam.

- **7. Accomplishment of the student after completing the course**:

At the end of the course, the student should be able to:
**1)** Understand concepts of networking and gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
**2)** Identify the components required to build different types of networks
**3)** Obtain the skills of sub netting and routing mechanisms.
**4)** Have a working knowledge of datagram
**5)** Trace the flow of information from one node to another node in the network

## Practical List

1. Download Wireshark from its official webpage, It is a network packet analyzer https://www.wireshark.org/
2. Install Wireshark under Windows/Linux/MAC/Solaris platform, Windows installer names contain the platform and version. Install WinPcap

**Lab1:**
**Objective:** To make the students aware about the IT/Network infrastructure of their parent institute.

Introduction to the lab infrastructure, the cabling/cable type and specifications/switch/topology/router/network infrastructure, Internet Connectivity, Wi-fi Connectivity of your department (labs and lecture halls) and how it is connected with the overall institute level IT infrastructure, How the institute obtains Internet Connectivity from its ISP, The Bandwidth and other specifications, Visit to the Computer Center of the Institute, Introduction to various Enterprise Servers/Servers of Licensed Software, Server hardware configuration, Server Management Tools, Enterprise Level Firewall, etc. So, lab 1 is intended as an in-campus IT infrastructure industrial visit for students. Students are supposed to prepare a report of this visit. Students are also supposed to note down the model/make of various network interface devices (NIC card, switch, router) used in their lab/department/institute, download their technical specifications from their respective web-site and attach those specifications in the report. The significance and interpretation of these specifications may be discussed by the faculty member during the corresponding lab/lecture hours pertaining to that particular device.

**Lab 2:**
**Objective:** To make the students aware about and learn the detailed use of the following OS level TCP/IP diagnostic and troubleshooting commands: ping, ns lookup

**Lab 3:**
**Objective:** To make the students aware about and learn the detailed use of the following OS level TCP/IP diagnostic and troubleshooting commands: ipconfig, arp, netstat, tracert, telnet

**Lab 4a:**
**Objective:** Implement a simple TCP socket based client server program in Python in which the client connects to the server. The server displays the ip address and port number of client and sends an acknowledgement message back to client. The client displays the received acknowledgement message on screen.

**Lab 4b:**
**Objective:** Implement a simple UDP socket based client server program in Python in which the client connects to the server. The server displays the ip address and port number of client and sends an acknowledgement message back to client. The client displays the received acknowledgement message on screen.

**Lab 5:**
**Objective:** Wireshark Installation, understanding the Wireshark Environment / Menu System, to start and stop live capture of traffic from given wired ethernet network interface, capturing

options, store captured data in different supported file formats, to open already stored captured data file.

**Lab 6:**
**Objective:** Learn and use view level filters and capture level filters in Wireshark for different traffic types like Ethernet, ARP, IP, TCP, UDP, DNS, HTTP, etc. For each captured category, observe how different headers are encapsulated within each other. Eg. TCP encapsulated within IP, HTTP encapsulated within TCP, etc.

**Lab 7:**
**Objective:** Capture the following traffic types and Interpret/ Analyze the corresponding header and payload: Ethernet and ARP

**Lab 8:**
**Objective:** Capture the following traffic types and Interpret/ Analyze the corresponding header and payload: IP and ICMP

**Lab 9:**
**Objective:** Capture the following traffic types and Interpret/ Analyze the corresponding header and payload: TCP and UDP

**Lab 10:**
**Objective:** Capture the following traffic types and Interpret/ Analyze the corresponding header and payload: HTTP and DNS

**Lab 11:**
**Objective:** Capture the following traffic types and Interpret/ Analyze the corresponding header and payload: FTP, SMTP and Telnet

**Lab 12:**
**Objective:** Capture the following traffic types and Interpret/ Analyze the corresponding header and payload: Ethernet and ARP

**Lab 13:**
**Objective:** Capture Wi-FI and Bluetooth Traffic and Interpret/ Analyze the corresponding header and payload using Wireless Traffic Sniffing tools like WireShark-USB/AirCrack-ng/Kismet, etc.

**Lab 14:**
**Objective:** Analyze Email Traffic: Normal POP Communications, POP Problems, Dissect the POP Packet Structure, Filter on POP Traffic, Normal SMTP Communications, SMTP Problems, Dissect the SMTP Packet Structure, Filter on SMTP Traffic

**Lab 15:**
**Objective:** Analyze IEEE 802.11 (WLAN): Wireless LANs (WLANs) Traffic, Signal Strength and Interference, Capture WLAN Traffic, 802.11 Traffic Basics like Data Frame, Normal 802.11 Communications

The Practical Examination Exercises would be based on Lab Exercises based on above list EXCEPT Lab # 1, 4a, 4b, 13 and 15. For Lab # 1, 4a, 4b, 13 and 15, documentation/report/file should be maintained.

**References:**

**1)** https://technet.microsoft.com/en-us/library/bb727023.aspx
**2)** https://docs.python.org/2/howto/sockets.html
**3)** https://www.aircrack-ng.org/
**4)** https://www.kismetwireless.net/

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Statistical Methods (SM)**
**Subject Code: 4649302**

**With effective from academic year 2018-19**

1. **Learning Objectives:**
To understand and apply various concepts, techniques and methods used in Descriptive Statistics and Inferential Statistics. The knowledge and skills gained will equip students in carrying out preliminary Data Analytics tasks, and to prepare foundation to understand and apply the statistical techniques in various fields such as Total Quality Management, Simulation, Game Theory, Operations Research, etc in addition to Computer Science topics such as Machine Learning, Cryptography, Artificial Intelligence, Operating Systems, Data Structures and Algorithms, etc.

2. **Prerequisites**: Preliminary mathematical concepts

3. **Contents**:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to Statistics and Descriptive Statistics**<br>Introduction, Broad areas (classification) of Statistics;<br><br>**Describing Data Visually:** Frequency Distributions and Histograms; Pie Charts; Bar Charts: Pareto Chart, Scatter Plots (Degree of Association); Line Charts<br><br>**Descriptive Statistics:** Central Tendency; Mean and its Characteristics,<br>Median and its Characteristics, Quartiles and Percentiles, Mode;<br><br>**Dispersion:** Range, Mean Absolute Deviation, Interquartile Range (IQR);<br>Variance, Standard Deviation and its Characteristics, Coefficient of Variation;<br>**Standardized Data:** Chebyshev's Theorem, Outliers;<br><br>**Box Plots:** Fences and Unusual Data Values<br><br>**Grouped Data:** Nature, Mean and Standard Deviation, Accuracy Issues<br><br>**Skewness:** Coefficient of Skewness;<br><br>**Kurtosis**: Leptokurtic, Platykurtic, Mesokurtic;<br><br>**Measures of Association:** Covariance, Correlation, Coefficient of Correlation; Correlation and Causation | **18%** |
| **Unit II** | **Probability and Probability Distributions**<br><br>**Introduction:** Common Framework: Experiment, Event, Elementary Events, Sample Space; Definition of Probability; Marginal Probability;<br>Probability of Union of Events (Addition Laws), Probability Matrix;<br>Probability of Complement of a Union; Probability of Joint Events | **24%** |

| | | |
|---|---|---|
| | (General Laws of Multiplication); Conditional Probability; Mutually Exclusive Events, Independent Events; Revision of Probabilities: Bayes' Rule<br><br>**Discrete Probability Distributions:** Introduction, Binomial Distribution, Poisson Distribution, Applications; Overview of other Discrete Probability Distributions<br><br>**Continuous Probability Distributions:** Introduction, Normal Distribution, Exponential Distribution, Applications; Overview of other Continuous Probability Distributions | |
| **Unit III** | **Sampling, Sampling Distributions and Estimation**<br><br>**Types of Sampling:** Random, Nonrandom; Sampling Distribution of x-bar; Central Limit Theorem; z Formula for Sample Mean; Standard Error of Mean; Sampling from a Finite Population; Sampling Distribution of a Proportion, Standard Error of Proportion<br><br>**Estimation for Single Population:** Estimating the Population Mean using z Statistic (σ Known); Estimating the Population Mean using the z Statistic when the Sample Size is Small; Estimating the Population Mean using t Statistic ( σ Unknown); Estimating the Population Proportion; Estimating the Population Variance; Estimating Sample Size | **24%** |
| **Unit IV** | **One Sample Hypothesis Tests**<br>Introduction; Null Hypothesis, Alternate Hypothesis; Type I & Type II Errors, Testing Hypotheses about a Population Mean using z Statistic ( σ Known); Using Critical Value Method to test Hypotheses, Examples; Population Mean Testing Hypotheses about a Population Mean using t Statistic ( σ Unknown); Testing Hypotheses about a Proportion; Testing Hypotheses about a Variance<br><br>**Overview:** Statistical Inferences about Two Populations; Analysis of Variance | **18%** |
| **Unit V** | **Regression**<br>Introduction, Simple Regression Analysis, Least Square Analysis to Determine the Equation of Regression Line; Residual Analysis, Using Residual to Test the Assumptions of the Regression Model; Standard<br>Error of the Estimate; Coefficient of Determination; Hypothesis Testing for the Slope of the Regression Model; Testing the Overall Model; Using Regression to Develop a Forecasting Trend Line<br><br>**Overview:** Multiple Regression Model; Mathematical Transformation of Nonlinear Models | **16%** |

**4. Text Book:**

**1)** Ken Black, "Business Statistics for Contemporary Decision Making", Wiley Student Edition, 2010

## 5. Reference Books:

**1)** David P. Doane, Lori E. Seward, "Applied Statistics in Business and Economics" Tata McGraw-Hill, 2010
**2)** Anderson, Sweeney, Williams, "Statistics for business and economics", 9th edition,
**3)** Thompson Publication
**4)** Bharat Jhunjhunwala, "Business Statistics", first edition, S Chand, 2008
**5)** Richard Levin, David Rubin, "Statistics for Management", 7th edition, PHI
**6)** Nabendu Pal, Sahadeb Sarkar, "Statistics-Concepts and Applications", 2nd edition, PHI
**7)** J. Susan Milton & Jesse Arnold, "Introduction to Probability & Statistics: Principles & Applications for Engineering & Computing Sciences", McGraw-Hill Education
**8)** S P Gupta, "Statistical Methods", 30th edition, S Chand

## 6. Chapter wise coverage from the Text Books:

| Unit# | Chapter # |
|---|---|
| I | Chapter 1,2,3 |
| II | Chapter 4,5,6 |
| III | Chapter 7,8 |
| IV | Chapter 9,10,11 |
| V | Chapter 14,15,16 |

## 7. Accomplishment of the student after completing the course:

Students will be able to apply various concepts, techniques and methods used in Descriptive Statistics and Inferential Statistics in carrying out preliminary Data Analytics tasks. They will also be able to apply the statistical techniques in various fields such as Total Quality Management, Simulation, Game Theory, Operations Research, etc in addition to Computer Science topics such as Machine Learning, Cryptography, Artificial Intelligence, Operating Systems, Data Structures and Algorithms, etc.

# Practical List

**Objectives**: To implement statistical concepts using a standard tool, such as R. Such implementation is aimed at improved visualization of theoretical concepts. It is also aimed at laying a foundation for Data Analytics and Data Science

**Prerequisites**: Logical Thinking and Basic Statistical Concepts

**Advice (Note) to Teachers**:
The list of exercises given below is an indicative list.

Note: R has many datasets. Get the available datasets through command data(). Use R commands related to Statistics for several datasets for a good practice.

Some exercises have been labeled as "**Mandatory**" while other exercises have been marked as "**Desirable**". It is expected that all the students will do **Mandatory** exercises while bright students will additionally do **Desirable** exercises as well.

**List of Computer Lab Exercises**

**1. Introduction and a quick tour to R and R Studio (to be done in Lab) [09 Hours]**

**(a)** Basic data structures and constructs
**(b)** Available R Datasets, such as mtcars, faithful, etc
**(c)** Null, NA, Missing Values
**(d)** Basic Packages related to Statistics: e. g. stats, stats4, graphics, grDevices, modeest, agricolae, etc.

**2. Descriptive Statistics [09 Hours]**

**(a)** Compute Mean, Median, Quartiles, Percentile (use quantile() function), Variance, Standard Deviation, IQR, Minimum & Maximum Values, Summary Statistics & interpretation (**Mandatory**)
**(b)** Histogram, Scatter Plot, Box Plot, Density Plot of R data sets and interpretation (**Mandatory**)
**(c)** Generate Frequency Distribution of data as a data frame (**Mandatory**)
**(d)** Compute Correlation Coefficient and Covariance (**Mandatory**)

**3. Probability and Probability Distributions [09 Hours]**

**(a)** Use pnorm(), pbinom(), ppois(), pexp() functions to compute probabilities (**Mandatory**)
**(b)** Use dnorm(), dbinom(), dpois(), dexp() functions to compute probability density functions (**Mandatory**)
**(c)** Use qnorm(), qbinom(), qpois, qexp() functions to get x value corresponding to given probability value (**Mandatory**)
**(d)** Use different parameter values in 3 (a), and 3 (b) to observe the impact of different parameter values and prepare a note on that. (**Mandatory**)
**(e)** Plot above results and interpret (**Desirable**)
**(f)** Statistical test for normality using shapiro.test() function (**Desirable**)

**4. Sampling, Sampling Distribution, Hypothesis Testing [12 Hours]**

**(a)** Random sampling with or without replacement using sample() function (**Mandatory**)

**(b)** Generate n random samples (take n = 10, 50, 100, 200, 500, 1000 as an example), create a vector of Sample Means. Draw the Density Plot of Sample Means to visualize Central Limit Theorem (**Mandatory**)

**(c)** Take a sample and carry out Hypothesis Testing for the following cases: (**Mandatory**)

1. Std. Deviation known, Large Sample Size, Sample from Non-Normal Population
2. Std. Deviation known, Small Sample Size, Sample from Normal Population
3. Std. Deviation known, Small Sample Size, Sample from non-Normal Population
4. Std. Deviation not known
5. Hypothesis Test for Variance (Chi-square Test)

**5. Regression and Linear Modeling [06 Hours]**

**(a)** Linear regression: One Independent Variable using lm() function; Interpret the output of Model Analysis, Compute Correlation Coefficient, Interpret results (**Mandatory**)

**(b)** Linear regression: Multiple Independent Variables using lm() function; Interpret the output of Model Analysis (**Mandatory**)

**Reference Books:**

1. Pierre-Andre Cornillon, Arnaud Guyader, Francois Husson, Nicolas Jegou, Julie Josse, Maela Kloareg, Eric Matzner-Lober, Laurent Rouvière, "R for Statistics", CRC Press, Rs. 525/-.
2. Dr. Mark Gardener, "Beginning R: The Statistical Programming Language", Wiley, Rs. 450/-
3. Paul Teetor, "R Cookbook: Proven Recipes for Data Analysis, Statistics, and Graphics", O'Reilly Cookbooks, Rs. 700/-

**Reference Websites:**

1. https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf
2. https://cran.r-project.org/web/packages/IPSUR/vignettes/IPSUR.pdf
3. https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/readings/reading-questions-r-intro/
4. https://www.datacamp.com/introduction-to-statistics
5. http://tut-dl.com/item/lynda-r-statistics-essential-training
6. https://www.analyticsvidhya.com › Machine Learning
7. https://www.coursera.org/learn/r-programming
8. https://www.analyticsvidhya.com/blog/2016/02/free-read-books-statistics-mathematics data-science/

**Accomplishment of the student after completing the course:**

1. Students will be able to carry out preliminary data analysis with results displayed graphically, and study the characteristics of standard probability distributions with their plots.
2. Students will also be able to demonstrate the inductive proof of Central Limit Theorem and go through linear regression (model) with fitness test of model.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Mobile Computing (MC)**
**Subject Code: 4649303**

**With effective from academic year 2018-19**

1. **Learning Objectives:**
   - To be able to understand the process of developing software for the mobile
   - To be able to create mobile applications on the Android Platform
   - To be able to create mobile applications involving data storage in SQLite database.

2. **Prerequisites**: Knowledge of the Core Java Programming, database concepts

3. **Contents:**

| Unit No. | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to ANDROID :**<br>ANDROID SDK Features, Introduction to Development Features<br><br>**Basics of ANDROID**<br>Developing for ANDROID, developing for mobile and embedded devices, ANDROID development tools<br><br>**Creating Applications using ANDROID**<br>Basics of an ANDROID application, introduction to manifest, externalizing resources, application life cycle, ANDROID activities | **10%** |
| **Unit II** | **Building user interfaces**<br>Introduction to layouts, introduction to fragments, creating new views, introduction to adapters<br><br>**Intents and broadcast receivers**<br>Introduction to intents, creating intents and broadcast receivers<br><br>**Using Internet resources**<br>Downloading and parsing Internet resources, using the download manager, using Internet services | **25%** |
| **Unit III** | **Files, saving state and preferences**<br>Creating, saving and retrieving shares preferences, including static files as resources, working with the file system<br><br>**Database and content providers**<br>Introducing ANDROID databases, content values and cursors, working with SQLite databases, creating content providers, using content providers, native ANDROID content providers | **25%** |
| **Unit IV** | **Working in background**<br>Introducing services, using background threads, using alarms<br>**Enhancing user experience**<br>Introduction and addition of action bar, menus and dialogs, drawables and gradients, custom animations | |

| | | |
|---|---|---|
| | **Maps and location based services (**Using location based services, selecting a location provider, finding your current location, creating map based activities) | **20%** |
| **Unit V** | **Audio, video and using the camera**<br>Playing audio and video, manipulating raw audio, using camera to take pictures, recording video, adding media to media store<br>**Telephony and SMS**<br>Hardware support for telephony, using telephony, introducing SMS and MMS<br>**Monetizing, promoting and distributing the applications**<br>Signing and publishing applications, distributing applications, introduction to monetizing applications | **20%** |

**Desirable Topics:**

**1)** Navigation drawer. recycler view and material design: https://developer.android.com
**2)** Android 9 Overview (Kotlin)

**4. Text Book:**

**1)** Reto Meier Professional ANDROID 4 Application Development, WROX Latest Edition

**5. Reference Books:**

**1)** Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed. (2011)
**2)** Mark L Murphy, "Beginning Android", Wiley India Pvt Ltd (2009)
**3)** Sayed Y Hashimi and Satya Komatineni, "Pro Android", Wiley India Pvt Ltd (2009)

**6. Chapter wise Coverage from Text Book:**

| Unit No | Chapters |
|---|---|
| I | Chapter 1,2,3 |
| II | Chapter 4,5,6 |
| III | Chapter 7,8,9 |
| IV | Chapter 10,11,13 |
| V | Chapter 15, 17,19 |

**Tools/Technologies to be used:**

**1)** ANDROID Studio [Latest Version]
**2)** ANDROID Version [Jelly Bean and later]

**7. Accomplishments of the student after completing the course:**

Student will visualize the real world mobile application scenario and enables them for development and implementation of mobile applications

# Practical List

**Tools/Technologies to be used:**
ANDROID Studio [Latest Version]
ANDROID Version [Jelly Bean and later]

**Part 1: Mandatory**

**1. Installing "Android Studio IDE" and "Android SDK"**
Step 0: Pre-Installation Check List
Step 1: Install "Android Studio IDE" ((For Windows), (For Mac OSX) (For Linux))
Step 2: Installing Android SDK

**2. Write your First Android App**

**2.1 Hello-World**

Step 1: Create a New Android Project
Step 2: Setup Emulator (aka Android Virtual Device (AVD))
Step 3: Run the Android App on Emulator
Step 4: Run the Android App on Real Devices

**2.2 Hello-world "by Coding"**

Create User Interface (UI) on Android: (1) Write Java codes; (2) Layout via XML descriptions and let the system generates the Java Code for you.

- ***MainActivity.java***
- Dissecting the "***MainActivity.java***" - Application, Activity & View
- Android Application Descriptor File - "AndroidManifest.xml"
- Common Errors Modify

**2.3 Hello-World using "XML Layout"**

Step 1: Create a New Android Application
Step 2: Define the Layout in XML ***res\layout\activity_main.xml***"
Step 3: Defining String References and Values r***es\values\string.xml***"
Step 4: The Activity - "***MainActivity.java***"
Step 5: Run the App

**3.** Create "Hello World" application. That will display "Hello World" in the middle of the screen in the red color with white background.

**4.** Create an application that designs a layout with a text box and button named Submit. The user should enter the text in the text box. When the submit button is clicked then the text in the text box should be displayed in the toast.

**5.** Create an application to demonstrate Android Activity Life Cycle.

**6.** To understand Activity,
Intent Create sample application with login module.(Check username and password) On successful login, go to next screen. And on failing login, alert user using Toast. Also pass username to next screen.

**7.** Create an application that designs a layout having two text boxes user name and password. The user and password will be taken from the user. The username and password will be verified from the username and password stored in the resources folder. If the verification is successful then a new layout will appear which will open any URL in browser.

**8.** Create an application to call specific entered number by user in the Edit Text

**9.** Create an application that will show List of Countries in One fragment , and on selecting one country, second fragment should be displayed with name selected country and its flag.

**10.** Understanding of UI :

a. Create an UI such that, one fragment of screen have list of all the types of cars.

b. On selecting of any car name, second fragment of screen should show Car details Like: name, launched date, company name, images (using gallery) if available, show different colors in which it is available.

**11.** Android Program to Perform all Operations using Calculators

**12.** Understand resource folder & Adapter :

a. Create spinner with strings taken from resource folder (res >> value folder).

b. On changing spinner value, change image.

**13.** Create an application that parses given XML File.

**14.** Create an application that parses given JSON File.

**15.** Create an application that fetches foreground and background colors from User. Save it as preference. On the next screen change background color and font color based on values passed by user. If no preferences are set then use Black and White as default background and foreground color respectively.

**16.** Create an application that designs a layout having two text boxes user name and password. The user and password will be taken from the user. The username and password will be verified from database. If the verification is successful then a new layout will appear which will contain text "Welcome ,<UserName>" otherwise displays error message.

**17.** Create an application that will create database with country table( Country short Name ( IND), Name). Create an application to make Insert, update, Delete and retrieve operation on the database.

**18.** Understanding content providers and permissions: Read phonebook contacts using content providers and display in list.

**19.** Android Program to Demonstrate Layouts in an Activity and Nesting of Layouts and Demonstrate List View Activity

**20.** Create application that works like an alarm.

**21.** Understand Menu option and Context Menu : Create an application that will change color of the screen, based on selected options from the menu.

**22.** Create an application that will have spinner with list of animation names. On selecting animation name, that animation should affect on the images displayed below.

**23.** Android Program to Demonstrate an Adapter and Advanced Adapter.

**24.** Android Program to Demonstrate Broadcast Receiver, Broadcast Receiver to Intercept Custom Intent, Pending Intent, OrderedBroadcast.

**25.** Android Program to Demonstrate Intent Filter, Local Broadcast Manager, to

Monitoring Device State Changes Using Broadcast Intents.

**26.** Android Program to Open an Internet Data Stream.

**27.** Android Program to Parse Xml Using Xml Pull Parser, Parse Xml Using Dom Parser.

**28.** Android to demonstrate Download Manager in Android, Android Program to Demonstrate Connection to an Internet Resource.

**29.** Android Program to Demonstrate Creating and Saving Shared Preferences and Demonstrate Preference Screen and Sub-screen in a Preference Screen and Intent in Preference Screen.

**30.** Android Program to Demonstrate Preference Fragment, Preference Headers Preference Activity.

**31.** Android Program to Demonstrate Reading a File on SD Card, Reading and Writing to a File in Android and Instance Save State.

**32.** Create an application that will play a media file from the memory card.

**33.** Create an application to take picture using native application.

**34.** Create an application to pick up any image from the native application gallery and display it on the screen.

**35.** Read messages from the mobile and display it on the screen.

**36.** Create an application to send message between two emulators.

**37.** Android Program to Demonstrate Action Bar in Android, Disable the Action Bar in an Activity, Hide Title Label of Action Bar in Android, Add Actions in Action Bar, Respond to Added Actions in Action Bar in Android, Change the Displayed Text alongside the Application Icon at Runtime, Change the Background of Action Bar in Android.

**38.** Android Program to Demonstrate Shape Drawables, Gradient Drawables, Radial Gradient, Sweep Gradient in Android.

**Part 3: Advanced (Mandatory)**

**39.** Create an application that designs a layout having a rating bar. Whenever user gives the rating then a dialog box should appear with a message and button OK. The message in the rating bar should be displayed (e.g.: 1.5 rating selected) 'You have given 1.5 Rating'. When button OK is pressed than the dialog box should disappear.

**40.** Create an application that designs a layout to take contact label (name) and phone no from the user and store in the contacts using content provider. Also design a layout to read phone contacts stored using content provider and display contact label (name) in list view. When the user selects any contact label (name) from the list view dialer application should be launched and call should be made using dialer application.

**41.** Create an application that designs a layout with spinner and an image view. The spinner should contain various shapes like circle, rectangle and rounded rectangle. When a shape is selected from the spinner that shape should be drawn in the image view. (Hint: Use shape Drawable or xml files for shapes on image view)

**42.** Create an application that designs a layout containing a list view having options:

a. Create file

By clicking on first option Create file a new layout should be displayed having two text boxes file name and file data and one button save. Both file name and data of the file should be entered by the user and on clicking save the file should be saved and proper message should be displayed on saving a file.

b. Delete file

By clicking on second option delete file a new layout should be displayed having one text box file name and one button delete. The file name should be entered by the user and on clicking delete the file should be deleted and proper message should be displayed on deleting a file.

c. Display File.

By clicking on third option display file a new layout should be displayed having one text box for file name and one text view for file data and one button display. The file name should be entered by the user and on clicking the display button the contents of the file should be displayed in text view.

**43.** Create an application that designs a layout with 3 text boxes and an options menu. The options menu should contain options like Simple Interest and compound Interest. The text boxes should be used for the input of information like the principle amount, rate of interest and number of years. Then clicking on the option from menu appropriate operation should be performed and correct result should be displayed.

**44.** Create an application that designs a layout having text boxes and button submit. The details of doctor like doctor's first name, last name, mobile number, address, city and specialization should be entered by the user in the textboxes and on clicking on the button submit the data should be saved into the database. Create another layout that contains a text box, a button search and a text view that gives searching facility. User can search doctor's full information by providing doctor's name in the text box. On clicking on button search the information of the doctor should be displayed in text view.

**Doc_detail (doc_id, firstname, lastname, mob, add, city, specialization)**

**45.** Create an application that designs a layout with a list view and fetches the audio files stored in the SD Card. The names of the Audio files should be displayed in the list view. When the user clicks on the specific file name the title and duration of the audio should be displayed in the next layout.

**46.** Create an application that designs a layout having gallery of images and an image view. The gallery contains number of images. Whenever an image is selected from the gallery it should be displayed in the image view. The images should be fetched from the resources folder. (You can take any readymade image available or you can create your own)

**47.** Create an application that designs a layout having options menu and 2 text boxes for currency converter which allows user to select a particular conversion from following options.

a. Rupees to dollars
b. dollars to Rupees
c. Rupees to pound
d. Pound to Rupees

The data for conversion should be entered by the user in textbox. Then clicking on the option from menu appropriate operation should be performed and correct result should be displayed.

**48.** Create an application that designs a layout which provides field for enrollment number (text box), student's name (text box), course (radio buttons), semester (radio buttons)) and marks for sub1 (text box), sub2 (text box), sub3 (text box) and a button name submit. When user clicks on submit button data should be added to database. Design a layout which displays current record and previously added record in a tabular manner. Total should be calculated automatically and then stored into the database.

**Stud_marks(en_no, stud_nm, course, sem, sub1, sub2, sub3, total)**

**49.** Create an application that designs a layout having a list view having two options:

a. Add contacts

On selection of the first option a new layout should be displayed containing two text boxes name and phone no and a button save. The name and phone no should be entered by the user and on clicking the save button the contact should be saved in the contacts directory and appropriate message should be displayed.

b. Display contacts.

On selection of second option display contacts a new layout should be displayed with list view and all the names from the contacts should be fetched and displayed in the list view. (Hint: Use Contacts Contract Content Provider).

**50.** Create an application that designs a layout of a student registration form. The layout should contain fields like first name, last name, phone no, date of birth (use date control) gender (use radio buttons), hobbies (use check boxes) and a button register. On clicking register an alert dialog box should appear with a message "do you want to register" and with two buttons yes or no. If yes is clicked then a new layout should appear where all the entered details should be displayed and also calculate the age (in years only) of the student and display it. If no is clicked than the alert dialog box should disappear.

**51.** Create an application that designs a layout for making an Admission Form. The layout should contain details like student enrollment no (textbox), Sem (spinner), Course (spinner), Date (date control), DOB (Date picker), SSC marks (textbox), HSC Marks (spinner), CMAT Score (Textbox) and a button named Submit. When the button submit is clicked than a new layout should be opened that displays all the student details as a summary and also display Merit Score. Merit Score should be calculated when the submit button is clicked.

Merit score = Average of (SSC marks, HSC Marks, CMAT score)

**52.** Create an application that designs a layout to store Result Details for MCA. The layout must contain details like name (textbox), Sem 1 Marks(spinner), Sem 2 Marks(spinner), Sem 3 Marks(spinner) , Sem 4 Marks(spinner), Sem 5 Marks(spinner) , Sem 6 Marks(spinner) and a Button named Submit.

Note: Marks must be from (AA, AB, BB, BC, CC and FF)

When the submit button is selected all the details should be saved in the database. Also provide a layout to perform the following operations:

a. Update the Marks /grade.
b. Search the Students on basis of his marks / grade.

**53.** Create an application that designs a layout to store Student Details. The layout should contain information like: Roll number (textbox), Name (textbox), Birth date (Date Picker), Marks1 (textbox), Marks2 (textbox), Marks3 (textbox) and button submit. When button submit is clicked than all the details must be saved in the database. Also percentage and total should be calculated automatically and saved into the database. Also design a layout to perform following operations:

a. Update Any student's marks

b. Search the student details by roll no
c. Delete any student by roll no

**54.** Create an application to implement shopping cart.
**55.** Create an application to provide feedback using text and rating( do not use text box).

**Part 4: Desirable**

**56.** Create an application that creates an XML file with <Customer> as main tag. The structure of XML file should be as follows:
<Game>
< Game -id> </ Game -id>
< Game -name> </ Game -name>
< Game -price> </ Game -price>
</ Game>

Design a layout containing a list view which fetches all the Game names in the list view. When the user selects any particular Game name than a new layout should be displayed with a text view and all the information regarding that Game should be fetched from the XML file and displayed in the text view

**57.** Android Program to Demonstrate Material View Design.

**58.** Android Program to Demonstrate Recycler view.

**59.** Android Program to Demonstrate Navigation Drawer, Navigation Using Fragments, adding actions to Navigation Bar in Android, Navigation Drawer with Different Fragments in Android, Tab Navigation in Android, Drop down Navigation in Android.

**Note: It is expected that most of the practical's are performed from above list.**

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Software Project -2 (Data Science)**
**Subject Code: 4649304**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To be able to develop Data Science Project using open source technologies
- To learn Data Processing, Visualization and Analytical techniques on data set

## 2. Prerequisites: Knowledge of Programming Language, Python and Database concepts

## 3. General Guidelines for Data Science Project using Open source Technologies

**1)** Group: 2-3 Person.
**2)** The project should be free from plagiarism of any kind.
**3)** It is mandatory that the project should be developed using Python on Linux or Windows Platform. Linux is the preferred platform
**4)** Project must have proper documentation
**5)** This may not be a live project
**6)** Use of a data set is mandatory
**7)** Use of Data Visualization and Analytical methods is mandatory
**8)** Use of any popular Libraries/Framework based on Python is not prohibited, in which case the framework internals should also be known.

## 4. Knowledge about the following is expected to be demonstrated.

**1)** Proper knowledge about the purpose of the application
**2)** Use of Large Data set
**3)** Use of Data pre-processing (Cleaning, Dimension reduction etc.)
**4)** Use of Data Visualization
**5)** Use of Data Analytical algorithm

## 5. Expected Outcome

**1)** The objective of the Data Science Project Development is to make students aware about the industry based process and workings. As a result, Project must meet with the industry standards.
**2)** There will not be any compulsion to prepare a project report for the students but an application and supportive documents should be self-explanatory, so that evaluator may get the detail about the Project developed and can evaluate the students as per the evaluation criteria.
- Group size: 2-3 Persons.

**3)** Power Point Presentation Content (30 Slides Max.):

| | |
|---|---|
| **Business Objective** | Introduction, Problem Statement |
| **Plagiarism Report** | Similarity should be less more than 30% |
| **Understand Data** | About Data Source<br>Understand Data: Basic Questions<br>Understand Data: Data Wrangling<br>Understand Data: Exploratory Analysis |

| | |
|---|---|
| **Methodology** | Extract Features & Model, Methodology |
| **Data Visualization** | Implementation of Data visualization Techniques |
| **Present Results** | Present Results |
| **Conclusion** | Conclusion |
| **References** | References |
| **YouTube Link (*)** | |

**Note (*): It is preferable to have project Presentation uploaded on YouTube.**

## 6. Suggested

PS: Below list (a & b) are suggestive one. You may select any other relevant topics/Data Sets.

**a) Project Definitions**

**1)** A Study on Employee Attrition Prediction and Analysis
**2)** A Study on Student Dropout Prediction and Analysis
**3)** A Study on Student Result Prediction and Analysis
**4)** A Study on Heights and Weights Data
**5)** A Study on Loan Prediction and Analysis
**6)** A Study on Housing Data
**7)** A Study on Weather Data
**8)** A Study on Movie Lens ( https://movielens.org)
**9)** A Study on Trip Data
**10)** A Study on Census and Income Data
**11)** A Study on Songs Data
**12)** A Study on Sales Data
**13)** A Study on Online Shopping Data
**14)** A Study on Cyber Crime Data
**15)** A Study on Airline Safety
**16)** A Study on Spam emails /Get rid of Spam emails ()
**17)** A Study on Pictures / Working with Pictures
**18)** Working with Handwritten Information
**19)** Analyzing Reviews ( e.g. amazon.com)

**b) Suggested Data Sets**

**1)** Kaggle Data Sets (https://www.kaggle.com/datasets )
**2)** Data.gov Data Sets (https://data.gov.in/https://www.data.gov/ )
**3)** kdnuggets Data Sets (https://www.kdnuggets.com/datasets/index.html )
**4)** Titanic Data Set. ( https://www.rdocumentation.org/packages/titanic/versions/0.1.0 )
**5)** Boston Housing Data Set. (https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html )
**6)** Walmart Sales Forecasting Data Set. (https://data.world/just4jcgeorge/walmart-sales-datahttps://relational.fit.cvut.cz/dataset/Walmart )
**7)** Hubway Data Visualization Challenge. (**https://anitagraser.com/projects/hubway-data-visualization-challenge/ )**

**8)** Text Mining Data Set. ( https://archive.ics.uci.edu/ml/datasets.htmlhttp://www.rdatamining.com/resources/datahttps://searchbusinessanalytics.techtarget.com/feature/Simple-data-mining-examples-and-datasets )
**9)** Census Income Data Set. ( https://www2.1010data.com/documentationcenter/prod/Tutorials/MachineLearningExamples/CensusIncomeDataSet.html )
**10)** Movie Lens Data Set. ( http://files.grouplens.org/datasets/movielens/ml-latest-small-README.htmlhttp://files.grouplens.org/datasets/movielens/ml-latest-README.html )
**11)** Yelp Data Set. ( https://www.yelp.com/datasethttps://catalog.data.gov/dataset/yelp-datahttps://github.com/vc1492a/Yelp-Challenge-Dataset )
**12)** AWS Public Data Sets (https://aws.amazon.com/opendata/https://registry.opendata.aws/ )
**13)** Google Public Data Sets ( https://cloud.google.com/public-datasets/https://www.google.com/publicdata/directoryhttps://ai.google/tools/datasets/ )
**14)** Wikipedia Data Sets ( https://en.wikipedia.org/wiki/Wikipedia:Database_downloadhttps://meta.wikimedia.org/wiki/Datasetshttps://wiki.dbpedia.org/data-set-34https://piktochart.com/blog/8-useful-databases-to-dig-for-data/https://perso.telecom-paristech.fr/eagan/class/igr204/datasets )
**15)** UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/index.php )
**16)** World Bank Data Sets ( https://data.worldbank.org/indicatorhttp://wdi.worldbank.org/tableshttps://openknowledge.worldbank.org/http://www.worldbank.org/en/news/feature/2012/03/29/world-bank-more-open-accessible-and-searchable-data-support-vietnams-developmenthttps://twitter.com/worldbankdata?ref_src=twsrc%5Egoogle%7Ctwcamp%5Eserp%7Ctwgr%5Eauthor
**17)** Academic Torrents Data Sets (http://academictorrents.com/browse.php?cat=6http://academictorrents.com/collection/nasa-datasetshttps://www.techleer.com/articles/545-academic-torrents-a-distributed-system-for-sharing-enormous-datasets/ )
**18)** Twitter Data Set (https://twitter.github.io/typeahead.js/examples/https://github.com/twitter/typeahead.js/issues/1004https://github.com/guyz/twitter-sentiment-datasethttps://data.world/datasets/twitterhttps://old.datahub.io/dataset/twitter-sentiment-analysis
**19)** https://vincentarelbundock.github.io/Rdatasets/datasets.html
**20)** https://www.dataquest.io/blog/free-datasets-for-projects/

## 7. Evaluation

| Sr. No | Particulars | Weightage |
|---|---|---|
| 1 | Topic & Selection of Algorithm | 10% |
| 2 | Data Pre-processing ( Cleaning, Reducing Dimensionality ) | 20% |
| 3 | Data Visualization | 20% |
| 4 | Data Analysis / Algorithm | 20% |
| 5 | Result | 30% |

**Recommended Book(s):**

1) Field Cady, 'The Data Science Handbook ', Wiley Publication ISBN-13: 978-1119092940
2) Jake VanderPlas, 'Python Data Science Handbook ESSENTIAL TOOLS FOR WORKING WITH DATA', O'REILLY ISBN:978-1-491-91205-8
3) Rachel Schutt and Cathy O'Neil, Doing Data Science, O'REILLY
4) Wes McKinney,Python for Data Analysis Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition , O'REILLY
5) Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012
6) John W. Foreman (Author), Data Smart: Using Data Science to Transform Information into Insight, WILEY
7) John Paul Mueller, Luca Massaron, Python for Data Science For Dummies , WILEY

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Java Web Technology (JWT)**
**Subject Code: 4649305**

**With effective from academic year 2018-19**

1. **Learning Objectives:**

- To learn and work with the web components of Java EE. i.e. the Servlet specification.
- Student will be able to learn MVC architecture and develop dynamic web application using Java Servlet and Java Server Pages technology.

2. **Prerequisites:** Programming Language of JAVA, HTML, JavaScript and JDBC

3. **Contents:**

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Servlet Basics, Handling the Client Request: Form Data, HTTP Request Headers**<br>Servlet Basics, Basic Servlet structure, Servlets Generating text/html content, Packaging Servlets, The servlet life-cycle. Handling Client Request Form Data, Reading Form Data from Servlets, Handling Client Request: Reading Request Headers, Understanding HTTP/1.1 Request Headers: Changing the page according to how the user got there and accessing the Standard CGI Variables. | **20%** |
| **Unit II** | **Server Response, HTTP Status Codes, HTTP Response Headers, Handling Cookies and Session Tracking**<br>Specifying Status Codes, HTTP / 1.1 Status Codes, Using Redirections, HTTP Response Headers: Setting Response Headers from Servlets, Understanding HTTP / 1.1 Response Headers, Using Servlets to Generate JPEG Images, Handling Cookies: Remembering Usernames and Passwords, Deleting Cookies, Sending and Receiving Cookies, Using Cookie Attributes, Differentiating Session Cookies from Persistent Cookies, Using Cookies to Remember User Preferences, Session Tracking: Need for Session Tracking, Session Tracking API, Encoding URLs Sent to the Client and accumulating a List of User Data. | **20%** |
| **Unit III** | **Listeners and Filters**<br>Using ServletContextListener, HttpSessionListener, Understanding of all the other Listeners viz. ServletRequestListener, ServletContextAttributeListener, ServletRequestAttributeListener, HttpSessionAttributeListener.<br>Using Filters for pre and post processing of request. | **10%** |
| **Unit IV** | **Overview of Java Server Pages, Java Code with JSP Scripting, JSP Page Directives, Files and Applets in JSP and Java Beans Components in JSP**<br>JSP Basic Syntax, HTML Text, HTML comments, Template Text, JSP Comment, JSPExpression, JSP Scriptlet, JSP Declaration, JSP 08 Directives, JSP Action, JSP Expression Language Element, Custom Tag (Custom Action), Escaped Template Text, Using JSP Scripting Elements, Using Predefined Variables, XML syntax for Expressions, Scriptlets, Declarations and Directives, Using Scriptlets, Using Declarations, Using Page Directive, Using Standard Actions Tags<jsp:plugin>, <jsp:forward>,<jsp:include>, | **20%** |

| | | |
|---|---|---|
| | Using JavaBeans in JSP pages – <jsp:useBean>, <jsp:getProperty>, <jsp:setProperty>,Sharing Beans, Use of Scopes and their Attributes. | |
| **Unit V** | **Model-View-Controller (MVC), JSP 2.0 Expression and Accessing Database with JDBC**<br>Integrating Servlets and JSP in a Web Application (MVC Architecture for Web Applications), Implementing MVC with Request Dispatcher, Understanding Data Sharing Between Servlets and JSP, JSP Expression Language, Accessing Scoped Variables, Bean Properties, Collections and Implicit Objects Using EL, Using EL Operators and Accessing Database with JDBC. | **20%** |
| **Unit VI** | **Declarative Security and Programmatic Security**<br>Form-based authentication, Basic authentication, example, configuration Tomcat to use SSL, programmatic Security, Example, Handling all security programmatically, Example, Using programmatic security with SSL. | **10%** |

**4. Text Book(s):**

**1)** Marty Hall, Larry Brown, "Core Servlets and JavaServer Pages Volume – 1", Pearson Education, 2nd ed.(2004)
**2)** Marty Hall, Larry Brown, Yaakov Chaikin, "Core Servlets and JavaServer Pages Volume – 2", Pearson Education, 2nd ed.(2004)

**5. Reference Books:**

**1)** Black Book "Java server programming" J2EE, 1st ed., Dream Tech Publishers, 2008.
**2)** Subrahmanyam Allamaraju, Cedric Buest, Professional Java Server Programming, Wiley Publication
**3)** Pravin Jain, The Class of Java, Pearson
**4)** Christian Bauer, Gavin King , Java Persistence with Hibernate, MANNING
**5)** Giulio Zambon, Beginning JSP, JSF and Tomcat, Apress
**6)** Cay S. Horstmann, "Core Java , Volume I – Fundamentals", Pearson Education , 10th Edition, 2017
**7)** Cay S. Horstmann, "Core Java , Volume II – Advanced Features", Pearson Education , 10th Edition, 2017
**8)** Cay Horstmann and Gary Cornell, Core Java, Volume II: Advanced Features, Pearson Publication
**9)** James Keogh ,Complete Reference J2EE, McGraw-Hill publication

**Web Resources**
http://docs.oracle.com/javaee/6/tutorial/doc/bnafd.html

**6. Chapter wise Coverage from the Text Book:**

| Unit # | Book# | Chapter |
|---|---|---|
| I | 1 | Chapter 3 (except 3.5, 3.7, 3.8), Chapter 4 (only 4.2), Chapter 5 (only 5.1, 5.3, 5.6 and 5.7), |
| II | 1 | Chapter 6 (except 6.4), Chapter 7 (only 7.1, 7.2 and 7.5), Chapter 8 |

| | | |
|---|---|---|
| | | (only 8.1, 8.3, 8.4, 8.6, 8.7, 8.11), Chapter 9 (only 9.1, 9.3, 9.5, 9.7) |
| III | 2 | Chapters 5, 6 |
| IV | 1 | Chapters 10 (only 10.6), 11, 12, 13, 14 |
| V | 1 | Chapters 15, 16, 17 |
| VI | 2 | Chapters 3 (3.1 to 3.5), 4 (4.1 to 4.5) |

## 7. Accomplishments

Students will understand advanced concepts related to MVC architecture, web services, servlet, spring and Hibernate. Students will be able to develop dynamic web applications using Java technology without need of other's help.

## Practical List

**1)** Write a Servlet to display "Hello World" on browser.
**2)** Write a Servlet to display all the headers available from request.
**3)** Write a Servlet to display parameters available on request.
**4)** Write a Servlet to display all the attributes available from request and context.
**5)** Write a Servlet which displays a message and also displays how many times the message has been displayed (how many times the page has been visited).
**6)** Assume that we have got three pdf files for the MCA-1 Syllabus, MCA-2 Syllabus and MCA-3 Syllabus respectively, Now write a Servlet which displays the appropriate PDF file to the client, by looking at a request parameter for the year (1, 2 or 3).
**7)** Assume that the information regarding the marks for all the subjects of a student in the last exam are available in a database, Develop a Servlet which takes the enrollment number of a student as a request parameter and displays the marksheet for the student.
**8)** Develop a Servlet which looks for cookies for username and password, and forwards to a home.jsp in case the cookies are valid and forwards to login.jsp, in case the cookies are not found or the cookies are not valid.
**9)** Write a servlet to implement Session tracking using all four methods.
**10)** Develop a Servlet to authenticate a user, where the loginid and password are available as request parameters. In case the authentication is successful, it should setup a new session and store the user's information in the session before forwarding to home.jsp, which displays the user's information like full name, address, etc.
**11)** Write a simple JSP page to display a simple message (It may be a simple html page).
**12)** Write a JSP page, which uses the include directive to show its header and footer.
**13)** Create a listener that notifies (through System.out) whenever a user adds a product to a shopping cart (i.e. adds an object to the session object) or removes it again. Hint: check out the class HttpSessionAttributeListener. Make it print the name and price of the object (hint: access the session through the HttpBindingEvent object). Also, let the listener print the total price of all objects saved in the session so far (one way to accomplish this could be to keep a collection of all objects saved to the session – or just their keys – in the listener or an associated class).
**14)** Create a servlet filter that logs all access to and from servlets in an application and prints the following to System.out:

1. the time the request was received
2. the time the response was sent
3. how much time it took to process the request
4. the URL of the resource requested
5. the IP address of the visitor

**15)** Develop a interest calculation application in which user will provide all information in HTML form and that will be processed by servlet and response will be generated back to the user.
**16)** Develop an application to demonstrate how the client (browser) can remember the last time it visited a page and displays the duration of time since its last visit. (Hint: use Cookie)
**17)** Develop an application to keep track of one user across several servlet invocations within the same browser session.
**18)** Develop an application to write a "page-composite" JSP that includes other pages or passes control to another page. (Hint: Use <jsp:include> or <jsp:forward>).
**19)** You want to reduce the amount of Java coding in your JSP using a JavaBean component. (Hint: Use <jsp:useBean> with the name of your bean).

**20)** Write a JSP page which uses tags available from the standard tag library JSTL.
**21)** Update the JSP page from above exercise to use tags availabe from the standard tag library JSTL.
**22)** Develop a JSP Page to display the personal information and result information of the student in two different tabular formats.
**23)** Create the filter that can add the time at which above jsp file called with appropriate message.
**24)** Design a Listener that loads the name of company as an init-parameter in context.
Use this name on the JSP – product.jsp and contactus.jsp of the website. (HINT: Implement ServletContextListener interface)
**25)** Create a filter to maintain the log of suspicious access of a particular JSP.
Access to the JSP on Sundays is suspicious.
Also design the JSP for the application. The JSP should display the details about sales history of the company for past 7 years.
The output should be shown in excel format.
**26)** Design an application where a user enters username and password, and requests for a servlet. Use filter to validate the password use database. If password is valid servlet is given as response, otherwise give appropriate message through filter.
**27)** Develop a program to perform the database driven operation like insert, Delete, Update and select. To perform the above operations create one table named Employee.

| **Field Name** | **Field Type** |
|---|---|
| EmpId | Integer |
| Empname | Varchar |
| Emp_desig | Varchar |
| Emp_J_Date | Varchar |
| Emp_Salary | Numeric |

**28)** Write a Java application to invoke a stored procedure using a CallableStatement. For this a stored procedure called increment Salary may be developed to increase all the employees salary by a percentage specified in the parameter.
**29)** Write a Servlet which uses the concept of Request forwarding & including external source in the current servlet context.
**30)** Write a JSP Page to which uses Session Tracking for online shopping.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Big Data Tools (BDT)**
**Subject Code: 4649306**

**With effective from academic year 2018-19**

**1. Learning Objectives:**

- To understand basics of Big Data
- To understand various Big Data Tools

**2. Prerequisites:** Working knowledge of Programming Language and Database Concepts

**3. Contents:**

| Unit | Course Content | Weightage percentage |
|---|---|---|
| **Unit I** | **Unit 1: Introduction to Big Data**<br><br>Types of Digital Data: Classification of Data (Structured, Semi-structured and Unstructured), Characteristics of Data, Evolution of Big Data, Definition of Big Data, Challenges of Big Data, Characteristics of Big Data (Volume, Velocity, Variety), Other characteristics of Big Data which are not Definitional Traits of Big Data, Why Big Data?, Are we Information Consumer or Producer? Traditional BI vs Big Data, Typical Data Warehouse Environment, Typical Hadoop Environment, What is Changing in Realms of Big Data? Terminologies used in Big Data Environments | **15%** |
| **Unit II** | **Unit 2: Introduction to NoSQL and Hadoop**<br><br>**NoSQL:** Introduction: Where is it used? What is it?, Types of NoSQL databases, Why NoSQL?, Advantages of NoSQL, Use of NoSQL in Industry, SQL vs NoSQL, NewSQL<br><br>**Hadoop:** Introduction, Distributed Computing Challenges, History of Hadoop, Overview of Hadoop and Hadoop Ecosystems, Features and key advantages of Hadoop, Versions of Hadoop, Hadoop distributions, RDBMS versus Hadoop, Hadoop vs SQL, Integrated Hadoop Systems offered by leading market vendors, Cloud based Hadoop solutions, HDFS, Processing data with Hadoop, Managing Resources and applications with Hadoop YARN, Interacting with Hadoop Ecosystem | **25%** |
| **Unit III** | **Unit 3: Introduction to MongoDB and MapReduce**<br><br>**MongoDB**: Introduction: What is MongoDB? Why MongoDb? (using JSON, Creating or generating a unique key, Support for Dynamic Queries, Storing Binary Data, Replication, Sharding, Updating information in –place), Terms used in RDBMS and MongoDb, Data types in MongoDb, MongoDB Query Language<br><br>**MapReduce:** Data Flow, Map, Shuffle, Sort, Reduce, Hadoop Streaming, mrjob, Installation, wordcount in mrjob, Executing mrjob | **25%** |

| | | |
|---|---|---|
| **Unit IV** | **Unit 4: Introduction to HIVE and Pig**<br><br>**HIVE:** Introduction: What is HIVE? HIVE Architecture, HIVE data Types, HIVE File Formats, HIVE Query Language, RCFile implementation, SerDe, User-Defined Functions (UDF)<br>**Pig:** Introduction: What is Pig? The anatomy of Pig, Pig on Hadoop, Pig philosophy, Use Case for Pig- ETL Processing, Pig Latin overview, Data types in Pig, Running Pig, Execution modes of Pig, HDFS commands, Relational operators, Eval function, Complex Data Types, Piggy Bank, User-defined Functions, Parameter substitution, Diagnostic Operator, Word Count Example using Pig, When to use and not use Pig? ,Pig at Yahoo, Pig vs HIVE. | **25%** |
| **Unit V** | **Unit 5: Overview of SPARK**<br>Introduction to Data Analysis with Spark, Downloading Spark and Getting Started, Programming with RDDs | **10%** |

**4. Text Book(s):**

**1)** Seema Acharya, Subhashini Chellappan, " Big Data and Analytics", Wiley India Pvt. Ltd.,2015
**2)** Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau ,"Learning Spark",O'Reilly Media,2015
**3)** Zachary Radtka and Donald Miner,"Hadoop with Python",O'Reilly Media,2016
(Free ebook is available on the following link)(As on 12-10-2018)
https://www.oreilly.com/programming/free/hadoop-with-python.csp

**5. Reference Books:**

**1)** Shashank Tiwari, " Professional NoSQL", Wiley India Pvt. Ltd.,2011
**2)** Kyle Banker,Peter Bakkum,Shaun Verch,Douglas Garrett,Tim Hawkins,"MongoDB in Action", DreamTech Press, 2nd Edition ,2016
**3)** Chris Eaton,Paul Zikopoulos,Tom Deutsch,George Lapis,Dirk Deroos,"Understanding Big Data : Analytics for Enterprise Class Hadoop and Streaming Data", Mcgraw Hill Education (India)Pvt.Ltd.,2012
**4)** Tom White,"Hadoop: The Definitive Guide",O'Reilly Media,4th Edition,2015
**5)** Vignesh Prajapati,"Big Data Analytics With R and Hadoop", Packt Pub Ltd ,2013
**6)** Dt Editorial Services,"Big Data - Black Book", Dreamtech Press,2016

**Web Resources:**

**a)** http://www.bigdatauniversity.com
**b)** http://www.mongodb.com
**c)** http://hadoop.apache.org/

**6. Unit wise coverage from Textbook(s):**

| Unit 1 | Book# | Topics |
|---|---|---|
| **I** | 1 | Chapter. 1, 2, 3.12 |

| | | |
|---|---|---|
| **II** | 1 | Chapter 4,5 |
| **III** | 1,3 | Chapter 6 (Book 1), Chapter 2 (Book 3) |
| **IV** | 1 | Chapter 9,10 |
| **V** | 2 | Chapter 1,2 and 3 (For Chapter 2 and 3, only Python, No Java, No Scala) |

**7. Accomplishment**

Student will understand fundamentals of Big Data, Tools and Techniques.

# Practical List

**Part I: MongoDb**

- Learn to Use MongoDB Atlas (The Cloud Version of MongoDB)
- Install and configure MongoDB

**MongoDB Shell Commands / Queries**: View all databases, Create new database, Drop existing database, View current database, Switch over to a given database, db.help(), Display statistics of a given database, Display current version of MongoDB Server, Display list of collections in current database, Create Collection, Drop Collection, CRUD operations (Create, Read, Update, Delete), Insert, Update else insert, save, update, remove, Find, Dealing with Using NULL Values, Count, Limit, Sort, Skip, Arrays and Array Operations, Aggregate

**1)** Create a StudentMaster database with a collection called "Student" containing documents with some or all of the following fields: StudentRollNo, StudentName, Grade, Hobbies, and DOJ. Perform the following operations on the database:

1. Insert 10 Records in the database.
2. Find the document wherein the "StudName" has value "Ajay Rathod".
3. Find all documents in proper format. (Without _Id field)
4. Retrieve only Student Name and Grade.
5. Retrieve Student Name and Grade of student who is having _id column is 1.
6. Add new field "Address" in Student Collection.
7. Find those documents where the Grade is set to 'VII'.
8. Find those documents where the Grade is not set to 'VII'.
9. Find those documents where the Hobbies is set to either 'Chess'or is set to 'Dancing".
10. Find those documents where the Hobbies is set neither to 'Chess' nor is set to 'Dancing".
11. Find those documents where the student name begins with 'M'.
12. Find those documents where the student name has an "e" in any position.
13. Find those documents where the student name ends in "a".
14. Find total number of documents.
15. Find total the number of documents where Grade is 'VII'.
16. Sort the documents in ascending order of student name.
17. Display the last two records.

**2)** Create a MovieMaker Database with a collection called "Movies "containing documents with some or all of the following fields: titles, directors, years, actors. Perform the following operations on the database (either in the console or using any programming language):

1. Retrieve all documents
2. Retrieve all documents with Director set to "Quentin Tarantino"
3. Retrieve all documents where actors include "Brad Pitt".
4. Retrieve all movies released before the year 2000 or after 2010.

5. Add a synopsis to "The Hobbit: An Unexpected Journey" : "A reluctant hobbit, Bilbo Baggins, sets out to the Lonely Mountain with a spirited group of dwarves to reclaim their mountain home - and the gold within it - from the dragon Smaug."
6. Add a synopsis to "The Hobbit: The Desolation of Smaug" : "The dwarves, along with Bilbo Baggins and Gandalf the Grey, continue their quest to reclaim Erebor, their homeland, from Smaug. Bilbo Baggins is in possession of a mysterious and magical ring."
7. Add an actor named "Samuel L. Jackson" to the movie "Pulp Fiction"
8. Find all movies that have a synopsis that contains the word "Bilbo"
9. Find all movies that have a synopsis that contains the word "Gandalf"
10. Find all movies that have a synopsis that contains the word "Bilbo" and not the word "Gandalf"
11. Find all movies that have a synopsis that contains the word "dwarves" or "hobbit"
12. Find all movies that have a synopsis that contains the word "gold" and "dragon".
13. Delete the movie "Pee Wee Herman's Big Adventure"

**3)** Create a database named "BookStore" in MongoDB with a collection called "Books" containing documents with some or all of the following fields: bookId, bookTitle, authors(containing fields: authorName), publicationYear, publisher, Orders(containing fields: OrderedId, orderDate, customerName, price, quantityOrdered, discount).
Note that a book may have one or more authors and orders. Also, the same OrdereId can be present in one or more books. Perform the following operations on the database (either in the console or using any programming language):

1. Insert records for 10 books from 5 authors, and at least 20 orders in total.
2. Update the title of a particular book.
3. Display all the books having less than 3 authors and sort by book name.
4. Display the number of books from each publisher.
5. Use MapReduce function to display the total quantity of books ordered for each date.
6. Use MapReduce function to display the discount offered to a particular customer.

**4)** Create a database named "Store" in MongoDB with a collection called "Sales" containing documents with some or all of the following fields: customerId, customerName, gender, dataOfBirth, contactNumber, address (containing fields: houseNo, street, area, city, pincode), orders(containing fields: orderId, orderDate, items(containing fields: itemId, itemName, itemPrice, quantityOrdered, discount)). Note that some customers may not provide their date of birth and/or contact number. Also, not all products would be sold at a discount. Perform the following operations on the database (either in the console or using any programming language):

1. Insert records for 3 customers and 5 items in at least 20 orders.
2. Update the contact number of a particular customer.
3. Display customerId, customerName, gender, contactNumber, of customers residing in "Ahmedabad".
4. Display city-wise count of customers
5. Use MapReduce function to display the number of times each item was sold.

**5)** Create a database "BookStore" with a collection called "Books" containing documents with some or all of the following fields: Category,BookName,Author,quantity, price,pages. Perform the following operations on the database:

1. Insert Records for 5 books.
2. Write Map & Reduce functions to split the books into the following two categories: Bigbooks, Smallbooks. (Books which have more than 300 pages should be in the Big books category. Books which have less than 300 pages should be in the Small books category.)
3. Count the number of books in each category
4. Store the output as follow as documents in a new collection called "Book_Result".

| **Book Category** | **Count of the Books** |
|---|---|
| Big books | 2 |
| Small books | 3 |

**Part II: Hadoop HDFS**

- Installation and configuration for: Apache Hadoop Stand-Alone Mode and Pseudo Distributed Mode
- Installation and configuration for: Apache Hadoop Real Cluster consisting of a single Master and Two Slave nodes.
- Test the above set-up with sample examples bundled along with the downloaded package.
- To develop and execute sample programs like word-count, maximum temperature, etc. Using Python with Map-Reduce in Hadoop
- HDFS Commands: -ls, -ls -R, -mkdir, -put, -get, -copyFromLocal, -copyToLocal, -cat, -cp, -rm-r

**1)** Create a file "Sample" in a local file system and export it to the HDFS File System.
**2)** Write the HDFS command for copying a "Sample" file from HDFS to local File System.
**3)** Write HDFS commands for creating "Test" directory in HDFS and then removing that directory.
**4)** Write HDFS command to display complete list of directories and files of HDFS.
**5)** Write HDFS command for displaying the contents of "Sample" text file in HDFS on screen.
**6)** Write HDFS command for copying an existing "Sample" file in a "Test" HDFS directory to some another HDFS directory.

**Part III: MapReduce**

**1)** Prepare an "input" folder containing multiple text files. Create a program using MapReduce that would accept the path to the "input" folder and generate an "output" folder having a text file containing the total number of occurrences of each single word present in text document. For example, if the text containing in input files is as follows:
"We thank you for your visit to Ahmedabad. We hope that you would visit us again."
The Output should be as follow:

| Word | Word Length | No of occurrences |
|---|---|---|
| We | 2 | 2 |
| To | 2 | 1 |
| Us | 2 | 1 |
| You | 3 | 2 |
| For | 3 | 1 |
| Your | 4 | 1 |
| That | 4 | 1 |
| Hope | 4 | 1 |
| Thank | 5 | 1 |
| Visit | 5 | 2 |
| Would | 5 | 1 |
| Again | 5 | 1 |
| Ahmedabad | 9 | 1 |

**2)** Write a program for Matrix Vector Multiplication using MapReduce.

**3)** Write a program to perform Union, Intersection and Difference operation using MapReduce on following files.
Input files:
- **a)** Content of file 1 (apple, orange, mango, apple, banana)
- **b)** Content of file 2 (apple, apple, plum, kiwi, kiwi, mango, mango)
- **c)** Content of file 3 (orange, orange, plum, grapes, kiwi, mango, apple)

**Part IV: Pig**

- Install and configure Apache Pig
- Test the Pig Installation for local and map-reduce mode execution
- Test the Pig Installation for Interactive (Grunt Shell) and Batch Mode (.pig file) Execution
- Develop UDF (User Defined Function) in Python for Pig

Working with Pig Operators/Functions (LOAD, DUMP, FOREACH, GROUP, DISTINCT, LIMIT, ORDER BY, JOIN, UNION, SPLIT, SAMPLE, AVG, MAX, COUNT, TUPLE, MAP,PIGGY BANK, PARAMETER SUBSTITUTION, DESCRIBE, Simple Problems like Word Count using PIG

**1)** Write a pig script to load and store "Student data".(Student file contain Roll no, Name, Marks and GPA).
- **a)** Filter all the students who are having GPA>5.
- **b)** Display the name of all Students in Uppercase.
- **c)** Group tuples of students based on their GPA.
- **d)** Remove duplicates tuple of Student list.
- **e)** Display first three tuples from "student" relation.
- **f)** Display the names of students in ascending order.
- **g)** Join two relation namely Student and department (Rno,DeptNo,DeptName) based on the values contain in the roll no column.
- **h)** Merge content of two relation Student and department.
- **i)** Partition a relation based on the GPA's acquired by students.

**j)** To calculate the average marks for each student.
**k)** Calculate maximum marks of each student.
**l)** Count the number of tuples in a bag.

**2)** Load the file menu.csv (Category,Name, Price) and write one Pig script

**a)** Which meals cost more than 30.00?
**b)** Which meals contain the word "Panner "?
**c)** Which are the 10 most expensive meals?
**d)** For every day, what's the average price for a meal?
**e)** For every day, what's the most expensive meal?

**3)** Write a program to count Word on Pig.
**4)** Write a pig script to spilt customers for reward program based on their life time values.
If Life time values is >1000 and < =2000 then Silver Program
If Life time values is >20000 then Gold Program
**Input:**

| Customers | Lifetime value |
|---|---|
| Jack | 25000 |
| Smith | 8000 |
| David | 12000 |
| John | 15000 |
| Scott | 12000 |
| Lucy | 28000 |
| Ajay | 12000 |
| Vinay | 30000 |
| Joseph | 21000 |
| Joshi | 25000 |

**5)** Create a data file for below schemas:
Order: CustomerId, ItemId,ItemName,OrderDate,DelivaryDate
Customer: CustomerId,CustomerName, Address,City,State,Country
a) Load Order and Customer Data.
b) Write a pig latin Script to determine number of items bought by each customer.

**6)** Do the Following:
1. Create a file which contains bag dataset as shown below.

| User Id | From | To |
|---|---|---|
| user1001 | user1001@sample.com | {(user003@sample.com),(user004@sample.com), (user006@sample.com)} |
| user1002 | user1002@sample.com | {(user005@sample.com),(user006@sample.com)} |
| user1003 | user1003@sample.com | {(user001@sample.com),(user005@sample.com)} |

2. Write a pig latin statement to display the names of all users who have sent emails and also a list of the people that have sent the email to.
3. Store the result in a file.

**7)** Create a UDF to convert name into uppercase.

**Part V: Hive**

- Install and configure Apache Hive
- SerDe and User Defined Function Creation in Hive using Java

Create database, display list of existing databases, describe database, describe extended database, alter database properties, to make a given database as current database, drop database, create managed table, create external table, loading data into a table, working with collection data types, querying a table using select, querying collection data types, create static partition and load data into it from original table, static partition creation using alter, create dynamic partition, load data into dynamic partition, create bucket, create view, query view, drop view, sub-query, joins, Aggregation, Group By and Having, RC File Implementation

1. **Create a data file for below schemas**
   **Order**: CustId,ItemId,ItemName,OrderDate,Delivary Date
   **Customer**:CustId,CustName,Address,City,State,Country

   a) Create a table for Orders and Customer Data.
   b) Write Hive Query Language to find number of items bought by each customer.

2. Create a partition table for Customer Schema to reward customer based on their life time value.

| Customer Id | Customers | Lifetime value |
|---|---|---|
| 1001 | Jack | 25000 |
| 1002 | Smith | 8000 |
| 1003 | David | 12000 |
| 1004 | John | 15000 |
| 1005 | Scott | 12000 |
| 1006 | Lucy | 28000 |
| 1007 | Ajay | 12000 |
| 1008 | Vinay | 30000 |
| 1009 | Joseph | 21000 |
| 1010 | Joshi | 25000 |

   a) Create partition table if life time value is 12000.
   b) Create partition table for all life time values.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Data Mining and Data Visualization (DMDV)**
**Subject Code: 4649307**

**With effective from academic year 2018-19**

1. **Learning Objectives:**
   - To understand the need for Data Mining and advantages to the business world.
   - To get a clear idea of various classes of Data Mining techniques, their need, scenarios (situations) and scope of their applicability.
   - To learn the algorithms used for various type of Data Mining problems
   - To understand how to explore and communicate data using data visualization techniques

2. **Prerequisites:** Knowledge of RBMS and OLTP

3. **Contents:**

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction : Data Mining**<br><br>An Overview; What is Data Mining; Why Data Mining?, What Kind of Data can be mined?, What Kind of Patterns Can be Mined? Major Issues in Data Mining, Data Objects and Attribute Keys<br><br>**Data Pre-processing:** An Overview of Data Cleaning methods, Data Integration, Data Reduction and Data Transformation. | 10% |
| **Unit II** | **Mining Frequent Patterns, Associations, and Correlations**<br><br>**Basic Concepts:** Market Basket Analysis; Frequent Item sets, Closed Item sets, and Association Rules;<br><br>**Frequent Pattern Mining Methods:** Apriori Algorithm: Finding Frequent Itemsets Using Candidate Generation; Generating Association Rules from Frequent Item sets<br><br>**Which Patterns Are Interesting**?—Pattern Evaluation Methods (Strong Rules Are Not Necessarily Interesting, From Association Analysis to Correlation Analysis, A Comparison of Pattern Evaluation Measures)<br><br>**Overview:** Improving the Efficiency of Apriori | 15% |

| | | |
|---|---|---|
| **Unit III** | **Classification & Prediction**<br><br>**Classification :** Basic Concepts<br><br>**Decision Tree Induction:** Decision Tree Induction, Attribute Selection Measures; Tree Pruning; Scalability and Decision Tree Induction, Visual Mining for Decision tree induction<br><br>**Rule-based Classification:** Using IF-THEN Rules for Classification; Rule Extraction from a Decision Trees; An Overview of Rule Induction Using a Sequential Covering Algorithm<br><br>**Bayesian Classification:** Bayes' Theorem, Naive Bayesian Classification; Bayesian Belief Networks<br>**Model Evaluation and Selection:** Metrics for Evaluating Classifier Performance, Holdout Methods and Random Sub sampling, Cross-validation, Bootstrap, Model Selection Using Statistical Tests of Significance, Comparing Classifiers Based on Cost–Benefit and ROC Curves<br><br>**Techniques to Improve Classification Accuracy:** Introducing Ensemble Methods, Bagging, Boosting and Adaboost, An Overview of Random Forests, Improving Classification Accuracy of Class-imbalanced data | 20% |
| **Unit IV** | **Cluster Analysis**<br><br>Cluster Analysis (What is Cluster Analysis? Requirement for Cluster Analysis, Overview of Basic Clustering Methods); Distance Measures in Algorithmic Methods<br>Partitioning Methods (K-Means, K-Medoids)<br><br>**Hierarchical Methods:** Agglomerative versus Divisive Hierarchical Clustering, An Overview of BIRCH Multiphase Hierarchical Clustering using clustering feature tree, Chameleon Multiphase Hierarchical Clustering using Dynamic Modeling, Probabilistic Hierarchical Clustering,<br><br>**Density-Based Methods (An Overview of DBSCAN:** Density-Based Clustering Based on Connected Regions with High Density)<br><br>**Outliers Analysis:** What are Outliers? Types of Outliers, Challenges of Outliers Detection. | 20% |

| | | |
|---|---|---|
| **Unit V** | **Introduction to Data Visualization**<br><br>Acquiring Data, Visualizing Data, Simultaneous Acquisition and Visualization, Applications of Data Visualization (Uses in the Public Sector, Business-to-Business and inter-Business Uses, Business-to-Consumer Uses), Control of Presentation, Faster and Better JavaScript Processing, Rise of HTML5, Lowering the Implementation Bar<br><br>**Exploring the Visual Data Spectrum:** Charting Primitives (Data Points, Line Charts, PerfChart, Seasonality Chart, Bar Charts, Pie Charts, Area Charts), Exploring advanced Visualizations (Candlestick Charts, Bubble Charts, Surface Charts, Map Charts, Infographics).<br><br>Making use of HTML5 CANVAS, Integrating SVG | 20% |
| **Unit VI** | **Visualizing Data Programmatically**<br><br>Creating HTML5 CANVAS Charts (HTML5 Canvas Basics, Linear Interpolations, A Simple Column Chart, Adding Animations),<br><br>Startig with Google Charts (Google Charts API Basics, A Basic Bar Chart, A Basic Pie Chart, Working with Chart Animations) | **15%** |

**4. Text Books:**

**1)** Han, J., Kamber, M., Pei, J. Data mining concepts and techniques. Morgan Kaufmann, 3rd Edition, 2011
**2)** Jon Raasch, Graham Murray, Vadim Ogievetsky, Joseph Lowery, "JavaScript and jQuery for Data Analysis and Visualization", WROX

**5. Reference Books:**

**1)** Field Cady, 'The Data Science Handbook The Data Science Handbook', Wiley Publication ISBN-13: 978-1119092940
**2)** Pieter Adriaans & Dolf Zentinge, "Data Mining", Addison-Wesley, Pearson (2000)
**3)** Vikram Pudi & P. Radhakrishnan, "Data Mining", Oxford University Press (2009)
**4)** G. K. Gupta, "Introduction to Data Mining with Case Studies", EEE, PHI (2006)
**5)** A Julie Steele and Noah Iliinsky, Designing Data Visualizations: Representing Informational Relationships, O'Relly
**6)** Scott Murray, Interactive Data Visualization for Web, O'Relly

**6. Chapter wise Coverage from Main Book(s):**

| Unit# | Book# | Topics |
|---|---|---|
| **I** | **1** | **Chapter 1 (1.1,1.2,1.3,1.4,1.7),**<br>**Chapter 2-2.1,2.2,**<br>**Chapter 3- 3.1,3.2,3.3,3.4.1, 3.5** |
| **II** | **1** | **Chapter 6 ( 6.1, 6.2.1-6.2.3,6.3)** |

| | | |
|---|---|---|
| **III** | **1** | **Chapter 8, 9.1** |
| **IV** | **1** | **Chapter 10.1,10.2,10.3,10.4.1, 12.1** |
| **V** | **2** | **Chapter 1,3** |
| **VI** | **2** | **Chapter 9** |

**7. Accomplishments of the student after completing the course:**

- Student will understand fundamentals of Data Mining.
- Ability to apply Apriori algorithm for Association Mining
- Ability to apply Decision Tree and Bayesian algorithms for Classification
- Ability to differentiate between Classification & Clustering, and similarly between Supervised Learning & Unsupervised Learning
- Ability to understand Data visualization and visualize data programmatically.

# Practical List

## PART-I Data Mining Practical List

**Tools**: Data Mining using 'R' Programming / Python

1. Create your own data set (like customer, weather, agriculture etc.), load it and apply any pre-processing technique and clean the data, show results.
    a. Clean missing values
    b. Remove Data
    c. Data Smoothing using Binning
2. Pre-process and classify any data set like customer, agriculture, weather etc.
3. Implement Decision Tree algorithm by taking appropriate data set and predict the result. Calculate entropy and information gain.
4. Implement Association mining algorithm by taking appropriate data set and find support and confidence. Also show confusion matrix.
5. Implement Naive Bayesian algorithm taking any dataset of your choice and predict the result.
6. Implement K-means clustering algorithm by taking appropriate data set and predict the result.
7. Implement K-medoids clustering algorithm by taking appropriate data set and predict the result.
8. Implement CART algorithm by taking appropriate data set and predict the result.
9. Implement DBSCAN clustering algorithm by taking appropriate data set and predict the result.
10. Implement K-Nearest Neighbor algorithm by taking appropriate data set and predict the result.

Suggested Data set URLs:

i) www.kdnuggets.com/**datasets**/
ii) http://archive.ics.uci.edu/ml/

## PART-II Data Visualization Practical List

**Tools**: HTML5(Canvas and SVG tags), D3.js, Google API, Canvas.js

### A) Installation

1. Setup Environment for All the Tools

### B) Develop the following Program Using HTML5 CANVAS and SVG TAG

**1)** Develop the Different basic Graphical Shapes using HTM5 CANVAS
**2)** Develop the Different Advanced Graphical Shapes using HTM5 CANVAS
**3)** Develop the Different basic Graphical Shapes using HTM5 SVG
**4)** Develop the Different Advanced Graphical Shapes using HTM5 SVG

### C) Develop Following Program Using HTML5 and JavaScript

**5)** Develop the simple bar chart usingTML5 CANVAS
**6)** Read the data .txt file and draw Data Table
**7)** Read the data .txt file and draw Simple Bar Chart
**8)** Read the data .csv file and draw Data Table
**9)** Read the data .csv file and draw Column Bar Chart
**10)** Read the data XML file and draw Data Table
**11)** Read the data XML file and draw Simple Chart
**12)** Read JSON Data and draw Data Table
**13)** Read JSON Data and draw Simple Chart

**D) Develop Following Program Using HTML5 and D3.js and Canvas.js**

**14)** Showing the data as a column chart (simple)
**15)** Showing the data as a stacked column chart
**16)** Showing the Data as a column chart for four age group
**17)** Showing the data as a Line chart (single, fewer and multiple lines)
**18)** Showing the data as a Pie Chart (single and multiple pie)
**19)** Showing the data as a Bar Chart (Simple and multiple)

**E) Develop Following Program Using HTML5 and Google Chats API and Map API**

**20)** Using Google Charts API Basics draw charts like a Bar chart
**21)** Using Google Charts API Basics draw charts like a Line chart
**22)** Using Google Charts API Basics drawPieChart.
**23)** Using Google Charts API Basics draw Donut Chart.
**24)** Using Google Charts API Basics draw Candle Chart.
**25)** Using Google Charts API Basics draw other types of Chart.
**26)** Using Google API read JSON file and create Google Map.

**F) Develop the Dashboard using all types of Graphics charts using any one graphical Tools (desirable)**

**27)** Building interconnected Dashboard

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Software Quality Assurance (SQA)**
**Subject Code: 4649308**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To understand the basic view of software quality and quality factors.
- To understand the Software Quality Assurance (SQA) architecture and the details of its components.
- To understand of how the SQA components can be integrated into the project life cycle.
- To be familiar with the software quality infrastructure.

## 2. Prerequisites: Software Engineering Basics, Basics of Java Programming

## 3. Contents:

| Unit | Chapter Details | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to software Quality and Assurance**<br>The software quality challenge, Software quality, Software quality factors<br>Management and its role in software quality assurance | **10%** |
| **Unit II** | **Components of SQA**<br>The components of the software quality assurance system – overview<br>**Pre-project Software Quality Components**<br>Contract review, Development and quality plans | **15%** |
| **Unit III** | **SQA Components in the Project Life Cycleand Strategies**<br>Integrating quality activities in the project life cycle, Reviews, Software testing – strategies | **25%** |
| **Unit IV** | **Software Testing – Implementation**:<br>Software Quality Implementation, Assuring the quality of software maintenance components, Assuring the quality of external participants' contributions, CASE tools and their effect on software quality | **25%** |
| **Unit V** | **Software Quality Infrastructure Components**<br>Procedures and work instructions, Staff training and certification,Corrective and preventive actions, Documentation control | **10%** |
| **Unit VI** | **Software Quality Metrics**<br>Software Quality metrics, Cost of Quality | **15%** |

**Desirable Topics:**

**1)** Quality and Management Standards (ISO, CMMi, ISO/IEC, IEEE, EIA).

## 4. Text Book:

Daniel Galin, "Software Quality Assurance", Pearson Publication, 2009.

## 5. Reference Books:

**1)** KshirsagarNaik and PriyadarshiTripathy, Software Testing & Quality Assurance-Theory and Practice, Wiley Student edition
**2)** William E. Perry, Effective Methods for Software Testing, WILLEY, . 3rd Edition
**3)** Alan C. Gillies, "Software Quality: Theory and Management", International Thomson Computer Press, 1997.
**4)** M G Limaye, Software Testing, Tata McGraw-Hill Education, 2009

## 6. Chapter wise Coverage from Main Reference Book(s):

| Unit No. | Text Books | Topics/Subtopics |
|---|---|---|
| I | Book-1 | 1.1 to 1.2, 2.1 to 2.6, 3.1 to 3.6, 26.1 to 26.4 |
| II | Book-1 | 4.1 to 4.7,5.2 to 5.5, 6.1 to 6.4 |
| III | Book-1 | 7.1 to 7.4, 8.1 to 8.5, 9.1 to 9.5, |
| IV | Book-1 | 10.1 to 10.4, 11.1 to 11.4, 12.2 to 12.3, 13.1 to 13.4 |
| V | Book-1 | 14.1 to 14.4, 16.2 to 16.10,17.2 to 17.8 ,19.1 to 19.6 |
| VI | Book-1 | 21.1 to 21.6, 22.1 to 22.3 |

## 7. Accomplishments of the student after completing the course:

- Utilize the concepts in software development life cycle.
- Demonstrate their capability to adopt quality standards.
- Assess the quality of software product.
- Apply the concepts in preparing the quality plan & documents.

# Practical List

**A) Setup:**

1. Download and install the **Java Software Development Kit (JDK)**

   http://www.oracle.com/technetwork/java/javase/downloads/index.html
2. Download "**Eclipse IDE for Java Developers**"

   http://www.eclipse.org/downloads/
3. **Download the Selenium Java Client Driver**

   http://seleniumhq.org/download/
4. **Configure Eclipse IDE with WebDriver**

**B) Suggested Applications for QA & QC:**

| | |
|---|---|
| 1 | **Leave Management System with following modules:**<br>a. Login – Two types of User: Admin and User<br>b. Admin Functionalities:<br>i. Manage Leave Types<br>ii. Manage User Leaves<br>iii. Manage Users<br>iv. Manage Different Shifts<br>v. Manage Reporting Groups and Team Structure<br>c. Time and Attendance<br>i. User can view his/her attendance detail<br>ii. Admin can view user's attendance log<br>iii. Admin can generate various report like LateIn, EarlyOut, etc..<br>d. Leaves<br>i. User can apply leave and Admin can reject/approve<br>ii. User can view his leave request log, can modify and cancel as well<br>** Many other functionalities can be added to make it more complex |
| 2 | **Recruitment System**<br>a. Manage Positions and vacancies within an organization<br>b. Manage Advertisement on the positions<br>c. Manage vacancy process from application to acceptance or rejection<br>d. Manage and Setup Interviews<br>e. Maintain Database |
| 3 | **Consider the below module which is responsible for Package purchase and Payment in different currencies:**<br>– The User can purchase one out of three packages namely, Basic, Moderate, & Premium.<br>– The User can purchase a particular package in three currencies.<br>– The available currencies are Dollar, which is by default, Pound, & INR.<br>– User can change the currency only on first step of 3 which is while selecting the package.<br>a. The User Selects the Package.<br>b. The User calculates the VAT amount applicable with respect to the country selection. |

| | |
|---|---|
| | c. The User pays online through debit/ credit card only.<br>– List out all the possible Test Cases for above module.<br>– List out critical step, which if not properly handled can deviate the user from the system.<br>– List out the enhancement point which can avail user more flexibility about one of the above three steps. |
| 4 | **In Airline reservation system, the following features need to be tested namely,**<br>a. Login<br>b. Search and book flights<br>c. Search and book packages<br>d. Register<br>Feature not in scope,<br>e. Search and book hotels<br>– Pre-requities: Database & Payment gateway's sanbox enviroment access should be available.<br>– Prepare the Test Plan for the above with all the possible criteria need to be considered.<br>– Prepare the Test Cases for the features in scope to be tested.(At least one for each above mentioned feature)<br>– Prepare the Defect Report. |
| 5 | **Healthcare Web application with following modules:**<br>a. Patient Registration<br>b. Scheduling<br>c. Treatment<br>d. Billing |
| 6 | **Consider a School Management System,** which allows Parent to download Children's Progress Report and Results from School's website So How can we authenticate the User (parent)? & it has dual Authentication system integrated on LogIn page. |
| 7 | **Notepad "Save As" Functionality**<br>What are the different scenarios that can be identified for testing a simple notepad save as functionality? The Save as does navigate to the file name and file path. Given these two fields what are the various scenario that can be applied for testing? |

**Part 1: Test Planning (Desirable)**

a) Prepare Quality Plan for any Application like online shopping etc.
b) Prepare Test Plan for any Application like Railway Reservation System etc.

**Output:** Test plan and Quality Plan

**Part 2: Software Testing (Manual) - Mandatory**

a) Create Test cases (Unit, Integration, System and Acceptance Test Cases) for Application
b) Perform manual testing using test case created and prepare test Metrics

**Suggested Template for Test case creation.**

| Sr# | Test condition / Steps | Input | Expected Result | Actual Result | Pass/Fail |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |

Write test cases using following techniques (Suggested)
i) Coverage
ii) Boundary Value Analysis (BVA)
iii) Equivalence Partition (EP)
iv) State Transition Technique
v)Error Guessing Technique

c) Test Case Management Tools like JIRA ( desirable)

**Part 3: Software Testing (Automated) - Mandatory**

Test automation – script creation and execution

**Tools: Selenium**

**A) Concepts**

| | |
|---|---|
| 1 | Introduction to Selenium , Installation and Setup |
| 2 | Selenium WebDriver Commands<br>• Browser Commands<br>• Navigation Commands<br>• WebElement Commands<br>• FindElement and FindElements Command<br>• CheckBox & Radio Button Operations<br>• DropDown & Multiple Select Operations<br>• Handle Dynamic WebTables in Selenium Webdriver |
| 3 | Navigate back/forwards, get, refresh<br>I_ loading a page in current window / New window<br>2. Move back and forward<br>3. Refresh Page |
| 4 | Interrogation:<br>i) get window title<br>ii) current url<br>iii) Page source |
| 5 | Locating web elements by Id, ClassName,LinkText,PartialLinkText,Name,TagName, CssSelector , XPath |
| 6 | Inspecting elements in web browsers |
| 7 | Element interrogation |
| 8 | Manipulation :<br>Click, submit, shift-click, special actions, type text, clear text, list box selection and manipulation commands |
| 9 | Synchronization: Page load time out, implicit wait, explicit wait , ExpectedConditions class |
| 10 | Window handling: size, position, handles, switch to |
| 11 | Screenshot/capture |

| | |
|---|---|
| 12 | Browser profile: set preferrred language, changing user agent, enable extension |
| 13 | Cookies: reading, creating and deleting |
| 14 | Data Driven Testing ; Use pre-stored data as input and expected output ( possibel data source file Excel) |
| 15 | Page Object Model : test scenario like shopping cart from login |

B) Practical List

Write script and perform Following list of activities / test scenarios using offline and online web application ( Permissible)

| | |
|---|---|
| 1 | Open URL in different browsers (Chrome, Firefox, IE) |
| 2 | Perform mouse hovers and other events |
| 3 | To take snapshots. |
| 4 | Getting current time-stamp |
| 5 | To Handle a drop-down. |
| 6 | Handling 2 3 steps at a time like Website Registration+login+logout at a time in single run. |
| 7 | To Upload File. |
| 8 | Storing and fetching data from excel (E.g Registration data) |
| 9 | Radio button selection. |
| 10 | Checkbox Selection |
| 11 | Selection |
| 12 | Absolute/Relative XPaths |
| 13 | Basic Selenium framework (Advance). |
| 14 | Generate Reports (Advance) |
| 15 | Logging (Advance). |
| 16 | Handling multiple browser tabs. |
| 17 | Handling multiple div./frames of page. (Advance) |
| 18 | Handling browser's parent and child window.(Advance) |
| 19 | verifies an expected page title, UI Element, Text and Table content |
| 20 | Handling mouse over menu/sub-menu. |
| 21 | To Provide meaningful messages in assertions! |
| 22 | Test Login of any web application |
| 23 | Test Shopping cart |
| 24 | Test using excel file data as input |
| 25 | Generate test report |

**References:**
https://www.softwaretestinghelp.com/selenium-webdriver-commands-selenium-tutorial-17/
http://toolsqa.com/selenium-webdriver/

**Part 4: Advanced Testing - (Desirable)**

**a) Perform Security Testing**

1) Security testing for architecture, source code and user interface
2) Vulnerability testing with help of open source and licensed tools

**References: www.owasp.org: OWASP guideline compliance verification**

**Type: Manual / Automated (using Tool) - (Desirable)**

**b) Perform Performance Testing**
Performance testing for functionality for 50 concurrent users

**Type: Manual / Automated (using Tool) - (Desirable)**

**Part 5: Issue Tracking (**desirable**)**
Tools: JIRA, BugZilla

**Web References:**
1) http://www.opensourcetesting.org/
2) http://www.onestoptesting.com/

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 4th Semester**
**Subject Name: Cyber Security & Forensic (CSF)**
**Subject Code: 4649309**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To understand the major concepts of Cyber Security and Forensics and to create the awareness through simple practical tips and tricks and to educate the students to learn how to avoid becoming victims of cyber crime
- To understand the major concepts of Cyber Security and Forensics and to create the awareness through simple practical tips and tricks and to educate the students to learn how to avoid becoming victims of cyber-crimes.
- The subject and the course content will help to the student who wish to take up cyber forensics as career as well as those who want to seek careers in cyber security.
- To gain experience of doing independent study and research in the field of cyber security and cyber forensics.

## 2. Prerequisites:

Basic fundamental knowledge of Networking, Web Application, Mobile Application and Relational Database Management System

## 3. Contents

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **UNIT- I: Introduction to Cybercrime:**<br><br>Introduction, Classifications of Cybercrimes: E-Mail Spoofing, Spamming, Cyber defamation, Internet Time Theft, Newsgroup Spam/Crimes from Usenet Newsgroup, Industrial Spying/Industrial Espionage, Hacking, Online Frauds, Pornographic Offenses , Software<br>Piracy, Password Sniffing, Credit Card Frauds and Identity Theft.<br><br>Cyber offenses: How Criminals Plan that attack, Categories of Cybercrime, How Criminals Plan the Attacks: Passive Attack, Active Attacks, Scanning/Scrutinizing gathered Information, Attack (Gaining and Maintaining the System Access), Social Engineering, Cyberstalking, Cybercafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector and Cloud Computing. | **15%** |
| **Unit II** | **Cybercrime: Mobile and Wireless Devices**<br><br>Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era and Laptops. | **10%** |
| **Unit III** | **Tools and Methods Used in Cybercrime**<br>Introduction, Proxy Servers and Anonymizers, Phishing, Password | **25%** |

| | | |
|---|---|---|
| | Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks. Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft): Types of Identity Theft, Techniques of ID Theft, Identity Theft- Countermeasures, How to Protect your Online Identity. | |
| **Unit IV** | **Cybercrimes and Cybersecurity: The Legal Perspectives**<br><br>Introduction, Why Do We Need Cyberlaws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act , Amendments to the Indian IT Act, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario. | **10%** |
| **Unit V** | **Understanding Computer Forensics**<br><br>Introduction, Historical Background of Cyberforensics, Digital Forensics Science, The Need for Computer Forensics, Cyberforensics and Digital Evidence, Forensics Analysis of E-Mail : RFC282, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Setting up a Computer Forensics Laboratory: Understanding the Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics, Forensics and Social Networking Sites: The Security/Privacy Threats, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing and Antiforensics. | **25%** |
| **Unit VI** | **Forensics of Hand-Held Devices**<br>Introduction, Hand-Held Devices and Digital Forensics, Toolkits for Hand-Held Device Forensics: EnCase, Device Seizure and PDA Seizure, Palm DD, Forensics Card Reader, Cell Seizure, MOBILedit!, ForensicSIM, Organizational Guidelines on Cell Phone Forensics: Hand- Held Forensics as the Specialty Domain in Crime Context . | **15%** |

**4. Text Book:**

**1)** Nina Godbole, Sunit Belpure, "Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", Wiley, 2011

**5. Reference Books:**

**1)** Dafydd Stuttard , The Web Application Hacker′s Handbook: Finding and Exploiting Security Flaws Paperback – Wiley, 2nd Edition, .
**2)** Wade Alcorn , Christian Frichot, Michele Orru, , The Browser Hacker's Handbook Book, WIley

**3)** James Graham, Richar Howard,Ryan Olson, "Cyber Security Essentials", CRC Press, Tailor and Francis Group, 2011
**4)** Robert Jones, "Internet Forensics: Using Digital Evidence to Solve Computer Crime",
**5)** O'Reilly Media, October, 2005
**6)** Chad Steel, "Windows Forensics: The field guide for conducting corporate computer investigations", Wiley India Publications, December, 2006
**7)** Nelson Phillips, Enfinger Steuart, "Computer Forensics and Investigations", Cengage Learning, New Delhi, 2009.
**8)** Kenneth J. Knapp, "Cyber Security and Global Information Assurance: Threat Analysis and Response Solutions", IGI Global, 2009.
**9)** Peter Wayner, "Disappearing Cryptography – Information Hiding: Steganography & Watermarking", Morgan Kaufmann Publishers, New York, 2002.

**Practical/ Tools:**

**1)** Mike Shema, Anti-Hacker Tool Kit (Indian Edition), Mc Graw Hill.
**2)** Christian Martorella , Learning Python Web Penetration Testing,PAKT
**3)** Vijay Kumar Velu , Mastering Kali Linux for Advanced Penetration Testing, PAKT, Book 2017
**4)** Nipun Jaswal, Mastering Metasploit,: Take your penetration testing and IT security skills to a whole new level with the secrets of Metasploit, 3rd Edition Paperback – Import, 28 May 2018 by
**5)** Gilberto Najera-Gutierrez, Juned Ahmed Ansari, Web Penetration Testing with Kali Linux - PAKT, Third Edition February 2018

## 6. Chapter Wise Coverage from Text Book:

| Unit | Book# | Topics |
|---|---|---|
| I | 1 | 1.1 to 1.5, 2.1 to 2.8 |
| II | 1 | 3.1 to 3.12 |
| III | 1 | 4.1 to 4.12 , 5.1, 5.2, 5.3 |
| IV | 1 | 6.1, 6.3, 6.4, 6.5, 6.6, 6.8, 6.9, 6.10 |
| V | 1 | 7.1 to 7.14, 7.16, 7.17, 7.18, 7.19 |
| VI | 1 | 8.1, 8.3, 8.4, 8.8 |

**Additional Topics:**

Cybercrime: Illustrations, Examples and Mini-Cases, Scams
(Only for the referential context should not be asked in the examination)

**Real-Life Examples:**

Example 1: Official Website of Maharashtra Government Hacked
Example 2: E-Mail Spoofing Instances
Example 3: I Love You Melissa – Come Meet Me on the Internet
Example 4: Ring-Ring Telephone Ring: Chatting Sessions Turn Dangerous
Example 5: Young Lady's Privacy Impacted
Example 6: Indian Banks Lose Millions of Rupees
Example 7: "Justice" vs. "Justice": Software Developer Arrested for Launching Website Attacks
Example 8: Parliament Attack
Example9: Pune City Police Bust Nigerian Racket

Mini-Cases:
- Mini-Case 1: Cyberpornography Involving a Juvenile Criminal
- Mini-Case 2: Cyberdefamation: A Young Couple Impacted
- Mini-Case 12: Internet Used for Murdering
- Mini-Case 13: Social Networking Victim – The MySpace Suicide Case
- Mini-Case 16: NASSCOM vs. Ajay Sood and Others

Online Scams:
- Scam No. 1 – Foreign Country Visit Bait
- Scam No. 2 – Romance Scam
- Scam No. 3 – Lottery Scam
- Scam No. 4 – Bomb Scams
- Scam No. 5 – Charity Scams
- Scam No. 6 – Fake Job Offer Scam

Financial Crimes in Cyber Domain:
- Financial Crime 1: Banking Related Frauds
- Financial Crime 2: Credit Card Related Frauds

**7. Accomplishment:**

After learning the course the students should be able to: student should understand cyber-attack, types of cybercrimes, cyber laws and also how to protect them self and ultimately society from such attacks

# Practical List

**Part I - Commands**

1. Study of following network emulators:
   i) WHOIS Search
   ii) Whois CLI Command
   iii) Nslookup
   iv) Host
   v) Ping
   vi) Traceroute
   vii) Netstat
   viii) Tcpdump and Windump
2. Create a malicious program that is (Atleast one program):
   i) Virus
   ii) Worm
   iii) Trojan
   iv) Dropper
3. TCP / UDP connectivity using Netcat
4. TCP scanning using NMAP.
5. Port scanning using NMAP.
6. TCP / UDP connectivity using Netcat.

**Part II - Exploits**

7. Exploit Web application Security using DVWA (Manual).
   Command Execution
   - SQL Injection
   - File Inclusion
   - XSS /CSIRF
   - Brute Force

8. Exploit Web application Security using DVWA
   Automated SQL injection with SqlMap .

**Part III - Forensics**

9. Perform a forensic analysis through autopsy sleuth kit.
10. Perform forensic analysis through helix.
11. Study of Forensic Tools ( Study any TWO)
    - Password Clearing
    - File Recovery
    - Data Hiding Techniques
    - Steganography
    - CheckSum
    - Hiren's BootCD

**Note: Above list is a suggestive, you may selective from Internet**

**Part IV: Desirable (add on knowledge)**

1. Network vulnerability using OpenVAS.
2. Perform image acquisition of the first partition carry out a dead analysis on image.
3. Study "omni peek "and perform live network analysis to capture packets.
4. Perform forensic data recovery through (Icare) a disk drill.
5. Perform forensic hash analysis and integrity check of evidence through FCIV and windiff.
6. Securely deleting file permanently (use tool like File shradder).
7. Install Kali-Linux on a PC for using it as an attacklaunching/vulnerability exploiting machine.
8. Create an intentionally vulnerableLinux Machine using MetaSploitable2 on another machine.
9. Perform Scanning/Reconnaissance testing on above mentioned machine in 5) using the machine mentioned in 4) using tools like NMAPand OpenVAS.
10. Study and Use MetaSploit Framework (already bundled with Kali Linux) present in machine to exploit vulnerabilities in the target vulnerable machine mentioned in5) using both command line and Armitage GUI utility.
11. Verify the integrity of a downloaded .tar.gz fileusing the shasum command. Eg. Hadoop Installation files can be taken as an example.Visit Hadoop Downloads Homepage:http://hadoop.apache.org/releases.html

**Evaluation Parameters:**

- Group Size : ( 2-3 Persons)
- Evaluation of the projects would be done considering Report ( Pahse I,II and III). The main parameter of assessment would be the ability of the students to understand Cyber Security and Forensic concepts and process
- Though the project and domain specific knowledge would be not be assessed for, the evaluation would predominantly depend on the students' ability to explain, modify or execute security testing.
- Though the project would be evaluated for the entire team, the examiner should emphasize on the contribution of each team member in the project
- Documentation
    - Outcome: Report ( Document) Minimum Pages : 50 Pages
    - The documentation should also include description related to Tools and methodologies used in.
    - Topics

| | | |
|---|---|---|
| I - Basics | a) Study, run and document ( Part I) | 20% |
| II - Exploit | Select an application and Exploit Web application Security using DVWA ( Manual and Automate)<br><br>Document work done | 30% |
| III _ Forensic | A) Perform a forensic analysis through autopsy sleuth kit. | 30% |

| | | |
|---|---|---|
| | or<br>Perform forensic analysis through helix. | |
| | B) Study and document any two Forensic Tools ( refer List Part 3 #11) | |
| IV - VIVA | VIVA | 20% |

- Following **is expected to be demonstrated**
  - Understanding of Basic Commands, Threats working
  - The execution of the Security Tools

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Design and Analysis of Algorithms (DAA)**
**Subject Code: 4659301**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

To understand alternate methods of writing algorithms under various categories, such as Divide-and-Conquer, Dynamic Programming, Greedy Methods, Backtracking, Branch & Bound, etc. To be able to analyze algorithms by working out complexity of algorithms. To understand the basics of P, NP, and NP Complete problems

**2. Prerequisites:** Programming Language: C, Data Structure

## 3. Course Contents:

| Sr. No. | Course Content | Percentage Weightage |
|---|---|---|
| 1 | **Basic Concepts of Analysis and Design of Algorithms**<br>Introduction; Characteristics of iterative algorithms; Efficiency of algorithms; Estimating and specifying execution time; Order notation: Big-Oh, Theta, Omega, Small-Oh, Small-Omega notations; Algorithm strategies | **7%** |
| 2 | **Algorithms Using Divide-and-Conquer Strategy**<br>Introduction; Examples: x**n; Multiplication algorithm and its analysis; Binary search and its analysis; Closest pair; Merge sort and its analysis; Limitations of Divide-and-Conquer strategy; Decrease-and-Conquer approach: Permutation generation | **20%** |
| 3 | **Greedy Methods**<br>Introduction; Knapsack problem; Job sequencing with deadlines; Minimum spanning trees: Prim's algorithm, and Kruskal's algorithm; Shortest path, Dijkstra's shortest path algorithm, Optimal merge patterns | **20%** |
| 4 | **Dynamic Programming**<br>Introduction; Examples: Coin exchange problem; Principle of Optimality; Rod cutting problem, Multistage graphs, Traveling salesman problem, Matrix multiplication, Longest common sub-sequence, Maximum flow problem | **20%** |
| 5 | **Backtracking, Branch and Bound Algorithms**<br>Combinatorial search; Search and traversal: Breadth First Search (BFS), Depth First Search (DFS); 8-Queen problem; M-Coloring problem; Hamiltonian circuits; Branch-and-Bound algorithms, Examples: Shortest path; 16-Puzzle and 8-Puzzle; Scale balancing, 0/1 Knapsack problem, Traveling salesman problem; Limitations of Branch-and-Bound | **23%** |

| | | |
|---|---|---|
| 6 | **Efficiency of Algorithms; Complexity Calculation and Categorization**<br>Polynomial-time and Non-polynomial-time algorithms; Worst and average case behavior, Probabilistic average case analysis; Time analysis of algorithms; Examples: Matrix multiplication; Efficiency of recursion; Complexity: Notion of complexity, Profiling, Suppressing multiplicative constants, Counting dominant operations, Growth rate, Upper bounds, Asymptotic growth rate; The 'O' notation; Simplified definition of 'O'; 'O' notation rules<br>**Examples of Complexity Calculation**: Sorting examples: Bucket sort, Radix sort, Simple Insertion sort, Quick sort, Heap sort, Merge sort, Counting Sort; Binary, Binomial and Fibonacci Heaps; Binomial Heap; Dijkstra's shortest path algorithm;<br>**Complexity Categorization of Problems**: Introduction: P, NP, NPC, NPH; Upper and lower bounds; Four levels of algorithmic behavior; summary | **10%** |

**4. Text Book(s):**

1. Parag H Dave, Himanshu B Dave, "Design and Analysis of Algorithms", Pearson, 2nd Edition (2014)

**5. Reference Books:**

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, "Introduction to Algorithms", PHI, 2nd Edition
2. Anany Levitin, "Introduction to Design and Analysis of Algorithms", Pearson (2014)
3. S. Baase, "Computer Algorithms: Introduction to Design and Analysis", Pearson (2002)
4. Kleinberg, "Algorithm Design", Pearson (2013)
5. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Universities Press, 2nd Edition (2008)
6. Thomas H. Cormen, "Algorithms Unlocked", MIT Press (2013)
7. Sanjay Dasgupta, "Algorithms", McGraw-Hill (2006)
8. Gerard Tel, "Introduction to Distributed Algorithms", Cambridge University Press (2004)

**6. Unit wise coverage from Text book(s):**

| Unit 1 | Topics |
|---|---|
| **I** | Chapter-1; Chapter-4 |
| **II** | Chapter-9 (9.1: Ex-1; 9.2: Ex-5, Ex-7, Ex-9; 9.4; 9.6.1, 9.6.2) |
| **III** | Chapter-10 (10.1, 10.2, 10.3, 10.4.1-10.4.4, 10.5, 10.6) |
| **IV** | Chapter-11 (11.1, 11.2, 11.3, 11.4, 11.5, 11.6, 11.9) |
| **V** | Chapter-12 (12.1; 12.2; 12.3; 12.4.2, 12.4.3; 12.6.1, 12.6.3, 12.6.4, 12.6.6, 12.6.7) |
| **VI** | Chapter-14 (14.1, 14.2, 14.3, 14.3.1, 14.4, 14.5, 14.5.1-14.5.11); Chapter-15 (15.1, 15.2, 15.4, 15.5); Chapter-17 (17.1, 17.2, 17.3, 17,18) |

**7. Accomplishment of the student after completing the course:**

The student will be able to decide on an appropriate category of algorithms for solving a given problem. With an understanding of more than one method of solving problems using algorithms, (s)he will be able to carry out complexity of algorithms and decide on an efficient algorithm for the task on hand. (S)he will also have an idea about categorization of problems into P, NP, NPC, NPH.

## Practicals List

**Objectives**: To develop ability to write algorithms (often more than one different algorithms) for various problems and implement them using C language.

**Prerequisites**: C Language, Data Structures

**Advice (Note) to Teachers:**

.The list of exercises given below is an indicative list. More than one algorithm is possible for many problems. Teachers should encourage students to try out multiple algorithms for the given problems and discuss among them about relative efficiency (and complexity) of algorithms.

.All the algorithms are to be implemented in C language

.First list of exercises have been labeled as "Mandatory" while the exercises in the second list have been marked as "Desirable". Teachers should encourage bright students to complete desirable exercises as well.

**A) List of Mandatory Lab Exercises (Write Algorithms and Implement in C Language)**

For the following problems, students are expected to write one or more (as the case may be) algorithms along with the complexity of these algorithms, and implement them in C Language.

1. Find square root of a number. Can we use Divide & Conquer approach for this problem? Can we have a still better algorithm to solve the problem?
2. Determine smallest divisor of an integer.
3. For a given value of n, generate prime numbers <= n (more than one algorithms are possible)
4. Find $X^n$. Iterative and recursive algorithms are possible with complexity $\log_2 n$
5. Determine product of 2 integers (a * b) as repeated sums. Iterative and recursive algorithms are possible.
6. Determine product of 2 large integers using multiplication of their digits. For simplicity, assume both numbers to have same number of digits. This assumption can be relaxed subsequently.
7. Binary Search of an ordered array. Iterative and Recursive algorithms are possible.
8. Sort a given sequence of numbers using (a) Bubble Sort, and (b) Merge Sort
9. Knapsack problem using Greedy algorithm.
10. Solution of Rod-cutting problem using Dynamic Programming algorithm.
11. Multiplication of n Matrices using Dynamic Programming algorithm.
12. Breadth First Search (BFS) in a binary tree.
13. Depth First Search (DFS) in a binary tree.
14. 8-Puzzle and 16-Puzzle
15. Solve 8 Queens problem.

16. Solve Scale Balancing problem.
17. Prim's algorithm to find minimum spanning (cost) tree (shortest path in a tree).

**B) List of Lab Exercises ( Write optimized code)**

18. **Number of ways:**
    You are given three strings A,B and C. From the strings A and B, you can create all possible strings X such that X contains atleast one character from both the strings, and the order of all the selected characters in individual strings is preserved.

    For example:
    A = "ab", B = "cd"
    All possible strings are: { abc, abcd, abd, ac, acb, acbd, acd, acdb, ad, adb, bc, bcd, bd, ca, cab, cabd, cad, cadb, cb, cbd, cda, cdab, cdb, da, dab, db }

19. **Compute** the sum of the Bitwise OR of all the subarrays present in the array.
    Input Array: 1 2 3
    all possible subsets are {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}
    ` Bitwise OR of these subsets are, 1 + 2 + 3 + 3 + 3 + 3 + 3 = 9.
    Output : 18

20. **Given** an array, we need to calculate the Sum of Bit-wise AND of all possible subsets of given array.
    Input Array: 1 2 3
    all possible subsets are {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}
    ` Bitwise AND of these subsets are, 1 + 2 + 3 + 0 + 1 + 2 + 0 = 9.
    Output : 9
21. Without using divide operator, write a function to divide two integers.
22. Find if a number is divisible by 17 using bitwise operators
23. Compute Subtract using plus operator.
24. Find maximum of two numbers without any comparison.
25. De Bruijn Sequence : Given a string (like AB), generate shortest String containing all combinations of the given string.

    Example:
    For given string "AB", all combinations are {AA, AB, BA, BB}
    One string containing all these combinations is AAABBABB.
    But this is not the shortest.
    Shortest string containing all combinations is AABBA
    Such a string is called
26. Given a number N, find all pairs of numbers such that $N^2 = X^2 + Y^2$
27. Given some integer N, find all the prime factors of that number.

**C) List of Desirable Lab Exercises (Write Algorithms and Implement in C Language)**
For the following problems, students are expected to write one or more (as the case may be) algorithms along with the complexity of these algorithms, and implement them in C Language.

28. Generate pseudo-random numbers.
29. Strassen matrix multiplication.

30. Find the closest pair out of given n points in 2-dimensional space.
31. Unique partitions of a positive integers.
32. Generate permutations of given n numbers. Iterative and recursive algorithms are possible.
33. Generate Gray Code.
34. Kruskal's algorithm to find minimum spanning (cost) tree (shortest path in a tree).
35. Determine largest common subsequence
36. Implement Twister game

**Reference Books:**

.Parag H Dave, Himanshu B Dave, "Design and Analysis of Algorithms", Pearson (2014)
.Anany Levitin, "Introduction to Design and Analysis of Algorithms", Pearson (2014)
.Thomas H. Cormen, "Algorithms Unlocked", MIT Press (2013)

**Accomplishment of the student after completing the course:**

The student will be able to write one (and sometimes more than one) algorithm to solve a given problem. (S)he will be able to determine complexity of algorithms and select the most efficient algorithm for a given task.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Machine Learning (ML)**
**Subject Code: 4659302**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- Basic concepts of various learning methods
- To learning algorithms used in machine learning

**2. Prerequisites:** Basics of computer science including algorithms, data structure, Basic Linear algebra and Probability theory.

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to Machine Learning, Model Preparation, Modelling and Evaluation**<br>Human learning versus machine learning, types of machine learning, applications of machine learning, tools for machine learning, Machine Learning Activities, Data structures for machine learning, Data Pre-processing, selecting a model, training a model, model representation and interpretability, evaluating performance of a model, improving performance of a model | 25% |
| **Unit II** | **Feature Engineering, Bayesian Concept Learning**<br>Introduction to feature engineering, feature transformation, feature subset selection, Importance of Bayesian methods, Bayes' theorem, concept learning through Bayes' theorem, Bayesian Belief Network | 20% |
| **Unit III** | **Supervised Learning – Classification, Regression**<br>Example of supervised learning, classification model, classification learning steps, common classification algorithms, example of regression, common regression algorithms,. | 20% |
| **Unit IV** | **Unsupervised Learning – Clustering, pattern finding using association rules**<br>Unsupervised learning versus supervised learning, applications of unsupervised learning, clustering and its types, Apriori algorithm for association rule learning | 17% |
| **Unit V** | **Neural Network**<br>Understanding the biological neuron, exploring artificial neuron, types of activation functions, early implementation of artificial neural network, architectures of neural network, learning process in artificial neural network, backpropagation, Overview of Deep Learning | 18% |

**Desirable Topics:**

Representation Learning, Active Learning, Instance-based learning, Ensemble Learning

## 4. Text Book:

**1)** Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, "Machine Learning", Pearson Education

## 5. Reference Books:

**1)** Tom M Mitchell, "Machine Learning", McGraw Hill
**2)** Anuradha Srinivarasaraghavan, Vincy Joseph, "Machine Learning", Wiley India

**3)** Peter Harrington, "Machine Learning in Action", DreamTech
**4)** Henrik Brink, Joseph Richards, Mark Fetherolf, "Real-World Machine Learning", DreamTech
**5)** Christopher Bishop, "Pattern Recognition and Machine Learning"
**6)** Jiawei Han and Michelline Kamber, "Data Mining: Tools and Techniques", 3rd Edition.

**6. Chapter wise Coverage from the Text Book:**

| Unit # | Chapter |
|---|---|
| I | Chapter 1: 1.1 to 1.8, Chapter 2: 2.1 to 2.6 and Chapter 3: 3.1 to 3.6 |
| II | Chapter 4: 4.1 to 4.3, Chapter 6: 6.1 to 6.5 |
| III | Chapter 7: 7.1 to 7.5 and Chapter 8: 8.1 to 8.3 |
| IV | Chapter 9: 9.1 to 9.5 |
| V | Chapter 10: 10.1 to 10.9 |

**7. Accomplishment of the student after completing the course**:

**1)** Student will be able to understand the concept of Machine learning and range of problems that could be solved by machine learning. They will be able to compare different types of learning algorithms and apply machine learning concepts in real life problems.

## Practical List

**1.** Write a python code to implement **decision tree** for below given dataset. Identify the root node and all subpart or children of node and draw the tree.

| **Item no** | **Age** | **Income** | **Student** | **Credit Rating** | **Buys-Computer** |
|---|---|---|---|---|---|
| 1 | Youth | High | No | Fair | No |
| 2 | Youth | High | No | Excellent | No |
| 3 | Middle | High | No | Fair | Yes |
| 4 | Senior | Medium | No | Fair | Yes |
| 5 | Senior | Low | Yes | Fair | Yes |
| 6 | Middle | Low | Yes | Excellent | No |
| 7 | Senior | Low | Yes | Excellent | Yes |
| 8 | Youth | Medium | No | Fair | No |
| 9 | Youth | Low | Yes | Fair | Yes |
| 10 | Senior | Medium | Yes | Fair | Yes |
| 11 | Youth | Medium | Yes | Excellent | Yes |
| 12 | Middle | Medium | No | Excellent | Yes |
| 13 | Middle | High | Yes | Fair | Yes |
| 14 | Senior | Medium | No | Excellent | No |

**2.** Write a python code to implement **K-nearest neighbourhood** program for the given dataset. Assume that value of K=19.

| Item no | Outlook | Temp | Humidity | Windy | Play |
|---|---|---|---|---|---|
| 1 | Sunny | Hot | High | FALSE | No |
| 2 | Sunny | Hot | High | TRUE | No |
| 3 | Overcast | Hot | High | FALSE | Yes |
| 4 | Rainy | Mild | High | FALSE | Yes |
| 5 | Rainy | Cool | Normal | FALSE | Yes |
| 6 | Rainy | Cool | Normal | TRUE | No |
| 7 | Overcast | Cool | Normal | TRUE | Yes |
| 8 | Sunny | Mild | High | FALSE | No |
| 9 | Sunny | Cool | Normal | FALSE | Yes |
| 10 | Rainy | Mild | Normal | FALSE | Yes |
| 11 | Sunny | Mild | Normal | TRUE | Yes |
| 12 | Overcast | Mild | High | TRUE | Yes |
| 13 | Overcast | Hot | Normal | FALSE | Yes |
| 14 | Rainy | Mild | High | TRUE | No |

**3.** Write a python code to implement Apriori **a**lgorithm, apply join and prune method and find frequent itemset

| **Sr#** | **Item no** | **Name** |
|---|---|---|
| 1 | T1 | Bread, butter, milk, soda |
| 2 | T2 | Coke, egg, milk |
| 3 | T3 | Bread, butter, egg |
| 4 | T4 | Break, coke, jam |
| 5 | T5 | Bread, butter |
| 6 | T6 | Potato chips, soda |
| 7 | T7 | Coke, fruit, juice |
| 8 | T8 | Bread, coke, milk |
| 9 | T9 | Coke, soda, jam, milk |
| 10 | T10 | Bread, butter, egg, milk, soda |
| 11 | T11 | Bread, milk |
| 12 | T12 | Bread, jam |

**4.** Write a python code to apply **Naive Bayesian** algorithm to clasify that whether a person can buy computer or not based on given test data :

| **Item no** | **Age** | **Income** | **Student** | **Credit Rating** | **Buys-Computer** |
|---|---|---|---|---|---|
| 1 | Youth | High | No | Fair | No |
| 2 | Youth | High | No | Excellent | No |
| 3 | Middle | High | No | Fair | Yes |
| 4 | Senior | Medium | No | Fair | Yes |
| 5 | Senior | Low | Yes | Fair | Yes |
| 6 | Middle | Low | Yes | Excellent | No |
| 7 | Senior | Low | Yes | Excellent | Yes |

| | | | | | |
|---|---|---|---|---|---|
| 8 | Youth | Medium | No | Fair | No |
| 9 | Youth | Low | Yes | Fair | Yes |
| 10 | Senior | Medium | Yes | Fair | Yes |
| 11 | Youth | Medium | Yes | Excellent | Yes |
| 12 | Middle | Medium | No | Excellent | Yes |
| 13 | Middle | High | Yes | Fair | Yes |
| 14 | Senior | Medium | No | Excellent | No |

Test Data
Age : Youth Income : LOW Student : No Credit Rating : Fair Buy Computer - ??

**5.** Write a python code to calculate gini index and draw the regression tree using **CART** algorithm for the below data set of ASCII characters:

511 975 511 975 496 966 480 967 450 981 420 993 376 1023 337 1053 297 1094 269 1135 248 1177 241 1215 252 1245 280 1264 323 1270 373 1263 431 1238 492 1205 550 1168 604 1126 648 1087 684 1056 710 1031 725 1013 725 1013 728 1025 719 1053 707 1095 690 1150 673 1214 654 1288 636 1368 616 1448 595 1523 574 1595 550 1656 529 1704 503 1731 484 1743 469 1734 463 1705 471 1661 493 1595 528 1524 581 1442 645 1364 709 1294 776 1231 838 1186 747 921 755 903 757 882 744 869 717 866 686 866 637 891 586 915 525 960 471 997 418 1045 379 1090 351 1132 340 1168 347 1195 369 1209 409 1211 459 1202 514 1185 572 1162 632 1127 688 1092 735 1053 777 1014 810 987 835 965 851 957 861 957 862 974 857 999 847 1043 833 1097 816 1159 797 1225 776 1300 753 1377 731 1452 708 1521 685 1581 660 1634 636 1674 611 1704 589 1716 568 1714 552 1704 539 1686 538 1657 538 1622 558 1575 583 1531 622 1485 672 1437 728 1387

**6.** Given the below training dataset of petal size and flower type, predict flower type for petal of size 2.5 cm. using K-Nearest-Neighbor classification.

| Petal Size | Flower Type |
|---|---|
| 1 | a |
| 2 | b |
| 1 | a |
| 2 | b |
| 3 | c |
| 4 | d |
| 3 | c |
| 2 | b |
| 5 | a |

**7.** Consider the below training data and determine if jacket is to be worn when the temperature is 12-degree Celsius using Linear Regression Model.

| **Outside Temperature** | **Wear a Jacket** |
|---|---|
| 30°C | No |
| 25°C | No |
| 20°C | No |

| | |
|---|---|
| 15°C | Yes |
| 10°C | Yes |

**8.** Implement a python program that takes interest rate (x), finds the equation that best fits the data and is able to forecast out median home price for given interest rate using the data given below. (Use linear regression)

| interest rate (%) (x) | Median home price (y) |
|---|---|
| 10.3 | $183,800 |
| 10.3 | $183,200 |
| 10.1 | $174,900 |
| 9.3 | $173,500 |
| 8.4 | $172,900 |
| 7.3 | $173,200 |
| 8.4 | $173,200 |
| 7.9 | $169,700 |
| 7.6 | $174,500 |
| 7.6 | $177,900 |
| 6.9 | $188,100 |
| 7.4 | $203,200 |
| 8.1 | $230,200 |
| 7 | $258,200 |
| 6.5 | $309,800 |
| 5.8 | $329,800 |

**9.** Write a python code to predict profit of hotel chain given the population of the area (city) using the data at https://docs.google.com/spreadsheets/d/1Ks20skBgEefHFU36sFqVzozoFtz2EZE2rxB_IgXOrUg/edit?usp=sharing.

**10.** Write a python code to predict the price of house given square feet and number of bed rooms in the house for the dataset available at https://docs.google.com/spreadsheets/d/1DHVK7gKo4TSyj7mFLwofHamj1Sl4SOZma2q51w1ZvyE/edit?usp=sharing

**11.** Build a classification model in python that classifies if a student gets admission in a course or not given his last two examination scores for the dataset available at https://docs.google.com/spreadsheets/d/1g0mjTUZ9Ado5prXA1UnAvNjmdzTrV0TzkFkIoU-Lpbk/edit?usp=sharing

**12.** Build a multivariate logistic regression model to classify glass type of glass given different glass mixture features using the Glass Identification Dataset from UCI Machine Learning Repository.

**13.** Implement supervised machine learning algorithm (Classification – K Nearest Neighbourhood) in python to classify breast tumour data into malignant breast tumour or benign breast tumour (use breast tumour dataset) and obtain its accuracy level.

**14.** Implement supervised machine learning algorithm (Classification – K Nearest Neighbourhood) in python to classify iris data into setosa, virginica, versicolor using iris dataset and obtain its accuracy level.

15. Implement supervised machine learning algorithm (Classification – Support Vector Machine) in python to classify breast tumour data into malignant breast tumour or benign breast tumour (use breast tumour dataset) and obtain its accuracy level.
16. Write a python program to build an email spam classifier using support vector machines for the Spam base dataset from UCI machine learning repository.
17. Implement unsupervised machine learning algorithm (Clustering – K Means) in python on Titanic dataset to cluster data (use Titanic dataset) by removing the class label.
18. Implement unsupervised machine learning algorithm (Clustering – K Means) in python on Breast Tumour dataset to cluster data (use Breast Tumour dataset) by removing the class label.
19. Implement unsupervised machine learning algorithm (Clustering – Hierarchical) in python on Titanic dataset to cluster data (use Titanic dataset).
20. Implement Apriori algorithm in python to find rules which explain association between different products for given transactions at a retail store. (The data is available at https://drive.google.com/file/d/1NUXoptUlHY8z4KcFKpFA6sQN5KnWzk3p/view?usp=sharing )
21. Implement text classification using neural network in python/R on Twenty Newsgroup dataset from UCI machine learning repository.
22. Implement supervised machine learning algorithm (Classification - Naïve Bayes algorithm) in python/R on Pima Indians Diabetes dataset and obtain its accuracy level.
23. classification and prediction algorithms on UCI dataset using Python's scikit-learn library

**Desirable Practical Lists**

1. For the sentiment analysis dataset given in link https://drive.google.com/file/d/1x6H7_KJjkbDrpgZFS7I2wjsZsILeSJ4S/view?usp=sharing, implement the following in python,
    a. Clean and pre-process the dataset by removing URL, removing HTML tags, handling negation words which are split into two parts, converting the words to lower cases, removing all non-letter characters
    b. Split the dataset into training and testing set
    c. Implement feature extraction technique (to convert textual data to the numeric form)
    d. Build the classification model using Logistic Regression that classifies if a given sentiment text is positive or negative
    e. Obtain the accuracy score of the built model.
2. Implement a content based recommender system in python that recommends movies that are similar to a particular movie using movielens-20m-dataset available at https://kaggle.com.

**The practical exercises should be performed in python.**

**References:**

**1)** Peter Harrington, "Machine Learning in Action", DreamTech
**2)** Michael Bowles, "Machine Learning in Python", Wiley
**3)** Gavin Hackeling, Mastering Machine Learning with scikit-learn, Packt
**4)** Giuseppe Bonaccorso, Machine Learning Algorithms - Second Edition, Packt

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Software Engineering (SE)**
**Subject Code: 4659303**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To understand the concepts of software Engineering
- To understand how to Select and apply Appropriate Process Model to All Stages of Software Development Life Cycle (SDLC)
- To understand how to manage user's Requirement
- To understand how to Analyze, Design, Build and test software
- To understand agile methodology.

## 2. Prerequisites: Systems & Object Oriented Design Methodologies

## 3. Contents:

| Unit | Chapter Details | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to Software Engineering & Process Models**<br><br>Software Engineering, Software Process<br><br>Process Models – Waterfall, Incremental, Evolutionary Process Model –<br>Prototype, Spiral and concurrent Development Model<br><br>Agile Process; Extreme Programming (XP); Brief Overview of Other Agile Process Models: Adaptive Software Development, Scrum | **10%** |
| **Unit II** | **Requirement Engineering**<br><br>Requirements Engineering; Groundwork for Understanding of Software<br>Requirements; Overview of Eliciting Requirements, Developing Use Cases, Building the Requirements Model; Negotiating Requirements; Validating Requirements;<br><br>Requirement Modelling Strategies; Overview of Flow-Oriented Modelling, Behavioural Modelling; | **20%** |
| **Unit III** | **Design Concepts**<br><br>Design Concepts, Design Model;<br><br>Architectural Styles, Architectural Design, Assessing Alternative architectural Designs, Architectural mapping Using Data Flow<br><br>User Interface Design: Golden Rules of User Interface Design; User<br>Interface Analysis and Design; Interface Analysis; Interface Design steps | **20%** |
| **Unit IV** | **Software Testing**<br><br>Software; Test Strategies for Object Oriented Software; Test Strategies for WebApps; System Testing; Debugging; | **10%** |

| | | |
|---|---|---|
| | Software Testing Fundamentals; White-Box Testing; Basic Path Testing;<br>Control Structure Testing; Black-Box Testing;<br><br>**Case Study: Prepare Test Cases for Online Shopping Application** | |
| **Unit V (*)** | **Introduction to Agile Methodology**<br><br>Agile Principles: 12 principles of Agile software, The customer is always right, Delivering the project, Communicating and working togather, Project execution - Moving the project Along,Constantly Improving the Project and the Team, Agile Project: Bringing all teh principles Together<br><br>Scrum and Self organizing Teams: The rules of Scrum, Everyone on a Scrum Team Owns Project, The whole team uses the daily Scrum, Sprints, planning and retrospectives<br><br>Scrum Planning and collective commitment: User stories, Velocity and Accepted Scrum Practices, Scrum Values revisited.<br><br>Case study: For any Application (e.g. Payroll, Online Shopping etc.), create Agile documents using SCRUM.<br><br>i) Agile Project Charter<br>ii) Agile Roadmap / Schedule<br>iii) Agile Project Plan<br>iv) Agile User Story ( Minimum 3 Tasks)<br>v) Agile Release Plan<br>vi) Agile Sprint Backlog<br>vii) Agile Test Plan<br>viii) Earned-value and burn charts | **20%** |
| **Unit VI (*)** | **HIGH LEVEL DESIGN**<br>**Overview:** What to specify: Security, Hardware (External) , User Interface, Internal Interfaces, External Interfaces, Architecture, Reports, Other Outputs, Database ( Audit trails, User Access, Database Maintenance), Configuration Data, Data Flows and States, Training, UML Diagrams ( Structure Diagram, Behavior Diagrams ( Use case, Activity, State Diagram), Interaction Diagrams, Sequence Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram<br><br>Case study: For any application (e.g. Payroll, Online Shopping etc.) System, create<br>i) Data Flow Diagram ( 0 Level)<br>ii) UML Diagrams : Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram | **20%** |

(*) Only application / case study to be asked in theory exam from Unit 5 and 6

**Desirable Topics:**

i) Reference for Case Study of SRS:
- Chapter 3 of Pankaj Jalote, "Software Engineering – A Precise Approach", Wiley India
- Chandramouli Subramanian, , Saikat Dutt,, Chandramouli Seetharaman, B G Geetha,Software Engineering, Pearson

ii) Agile Methodology XP, Lean and Kanban

iii) Project Estimation and Scheduling

**4. Text Book:**

1) Roger S. Pressman, "Software Engineering – A Practitioner's Approach", 7th Edition, McGraw Hill Publications

2) Andrew Stellman, Greene Jennifer, Beginning Agile, O'Reilly

3) Rods Stephen, Beginning Software Engineering, WROX

**5. Reference Books:**

**1)** Sommerville, "Software Engineering", 8th Edition, Pearson Education
**2)** Chandramouli Subramanian, , Saikat Dutt,, Chandramouli Seetharaman, B G Geetha,Software Engineering, Pearson
**3)** Waman S. Jawadekar, "Software Engineering – Principles and Practices", TMGH Publication
**4)** Pankaj Jalote, "Software Engineering – A Precise Approach", Wiley India
**5)** Waman S. Jawadekar, "Software Engineering – A Primer", TMGH Publication
**6)** M G Limaye, Software Testing, Tata McGraw-Hill Education, 2009

**6. Chapter wise Coverage from Main Reference Book(s):**

| Unit No. | Text Books | Topics/Subtopics |
|---|---|---|
| I | Book-1 | Chapter : 1.3,1.4, 2.1 to 2.3, 3.3, 3.4, 3.5.1 ,3.5.2 |
| II | Book-1 | Ch-5, 7.1 to 7.3 |
| III | Book-1 | Ch-8.3, 8.4, 9.1.1, 9.3,9.4,9.5, 9.6, 11.1 to 11.4 |
| IV | Book-1 | Ch-17, 18.1 to 18.6, |
| V | Book-2 | Chapter 3,4,5 |
| VI | Book-3 | Chapter 5 ( Overview) |

**7. Accomplishments of the student after completing the course:**

- Students will understand a high-level overview of the software development process.
- Student will understand various process models available for software engineering, activities of software engineering like software requirements, software design, software construction, software management, and software quality etc.
- Student will understand agile methodology.

**8. Suggested case studies for Unit 6:**

Prepare Data Flow Diagram (0 Level) and ii) UML Diagrams (Use Case Diagram, Activity Diagram, Sequence Diagram, and Class Diagram) for following definitions:

1. Consider a book store in a shopping mall. The customer selects the books from racks to purchase. Prepare a sequence diagram for bookstore checkout system. The customer brings selected books to cashier. The cashier scans each item with checkout system to prepare an order.

2. Consider an Online shopping web site, where customer selects the items and adds into cart. At the customer will proceed for payments. Prepare a Use case, Activity and sequence diagram for online shopping checkout system. The customer brings selected books to cashier. The cashier scans each item with checkout system to prepare an order.

3. The case study 'Online Mobile Recharge' gives us the information about all the mobile service providers. This application provides us the complete information regarding any mobile service provider in terms of their plans, options, benefits, etc. Suppose, any Airtel customer wants to have the information of all the schemes and services provided by the company, he/she can have the information and according to his convenience he can recharge the mobile from the same application. The major advantage of this proposed system is to have the recharging facility of any service provider under same roof.

4. In tour management system, System will check whether the customer is existing or new. New user will enter his personal and tour details for reservation. This login information could be used for further transactions. When customer is satisfied with tour package he/she would request for reservation of tour. Personal details of new customer is stored in cust_info while the details regarding the tour selected by particular customer is stored in tour_info and the details regarding it would be restructured in Tour Information System. Existing customer can update his/her personal details in cust_info and cancel reservation for tour from tour_info and changes regarding it are also reflected in Tour Information System. After confirming the tour package the customer will make payment either online or through staff by personally going at the office. Customer can make payment by cash, credit card or by cheque. System checks for the validity of staff. Once the payment is done by customer, valid staff will make Ticket Reservation System. Reserved customer will be able to view details about reservation by providing login information from cust_info and tour_info system. Administrator can add, delete or modify tour schemes from Tour Information System.

5. "Railway Reservation System is a system used for booking tickets over internet. Any Customer Can book tickets for different trains.Customer can book a ticket only if the tickets are available. Customer searches for the availability of tickets then if the tickets are available he books the tickets by initially filling details in a form. Tickets can be booked in two ways by i-ticket or by e-ticket booking. In case of i-ticket booking customer can book the tickets online and the tickets are couriered to Particular customer at their address. But in case of e-ticket booking and cancelling.
   Tickets are booked and cancelled online sitting at the home and customer himself has to take print of the ticket but in both the cases amount for tickets are deducted from customers account. For cancellation of ticket the customer has to go at reservation office than fill cancellation form and ask the clerk to cancel the ticket than the refund is transferred to customer account. After booking ticket the customer has to checkout by paying fare amount to clerk".

**6.** The **University** runs various diploma, graduation and post-graduation courses such as DE, Dpharm, BE, MBA. MCA, ME, Mpharm etc. The courses follows semester patterns and under each course various subjects are taught. Students seek admissions to these courses and if found elligible, the student is enrolled for the requested course. There are several faculty members in the university who teach the various subjects of these courses. The subject teacher conducts semester examinations for the concerned subject at the end of the semester and the student's performance is recorded. Even if a student is unable to pass a subject, he is promoted to the next semester but has to reappear for the subject examination again and clear his backlog.

**7.** Others:
a) Facebook Application.
b) WhatsApp Application.
c) Library Management System
d) Online Quiz System
e) ATM System

**PS: Above is a suggestive list. You may select any other relevant Application**.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Software Project-III (SP-III)**
**Subject Code: 4659304**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To be able to develop Project using Emerging Framework
- To solve industrial (or society or research) problems
- To plan, schedule, and monitor the software project
- Development, coding, and testing of a large project cohesively
- Documentation of project

**2. Prerequisites:** Programming Language, Basic Concepts of Software Engineering, RDBMS

**3. Suggested Frameworks** : IONIC / Xamarin / DJANGO

## 4. Guidelines for Project using Framework

- It is recommended that the team should be of 2-3 students.
- The project should be free from plagiarism of any kind.
- Project must have proper documentation.
- It is mandatory that the project should be developed using One of the above mentioned framework on Linux or Windows Platform. Linux is the preferred platform.
- This may not be a live project.
- Coding standards should be followed meticulously. At the minimum, the code should be self documented, modular, and should use the meaningful naming convention.
- It is advisable that object-oriented methodology is used with reusability of classes and code, etc.
- The output reports must include MIS reports, if applicable.
- The documentation should include a chapter on "Learning during Project Work", i.e. "Experience of Journey during Project Duration".
- Student may be asked to write the code related to the project during examination.
- Mentor/ Internal guides (i.e. the faculty members) must devote time, allocated as per the time table to guide the students for the project. The time allocation will be in accordance with the teaching scheme for 5th semester project.

## 5. Documentation:

- The project has to be well-documented in the form of a Project Report (at least 50 pages comprising of the relevant description of the project including design, data dictionary, source code, screenshots, etc.).
- Format : The Project report Print out should be taken on both the side of page with single line spacing. Use Times New Roman of size 10 for normal text. A typical Table of content will be as follows.

TABLE OF CONTENTS

1. Introduction
    - 1.1. Existing System
    - 1.2. Need for the New System
    - 1.3. Objective of the New System
    - 1.4. Problem Definition
    - 1.5. Core Components
    - 1.6. Project Profile
    - 1.7. Assumptions and Constraints
    - 1.8. Advantages and Limitations of the Proposed System

2. Requirement Determination & Analysis
    2.1. Requirement Determination
    2.2. Targeted Users
3. System Design
    3.1. Use Case Diagram
    3.2. Class Diagram
    3.3. Interaction Diagram
    3.4. Activity Diagram
    3.5. Data Dictionary
4. Development
    4.1. Coding Standards
5. Agile Documentation
    5.1 Agile Project Charter
    5.2 Agile Roadmap / Schedule
    5.3 Agile Project Plan
    5.4 Agile User Story ( Minimum 3 Tasks)
    5.5 Agile Release Plan
    5.6 Agile Sprint Backlog
    5.7 Agile Test Plan
    5.8 Earned-value and burn charts
6. Proposed Enhancements
7. Conclusion
8. Bibliography

## 6. Knowledge about the following is expected to be demonstrated.

- The objective of the Project Development is to make students aware about the industry based process and workings using Framework. As a result, Project must meet with the industry standards.
- Proper knowledge about the purpose of the application.
- Use of justifiable application for group of 2-3 members.
- Project must include features like MIS Reports, Advance Search, File based processing etc.

## 7. Evaluation

| Sr. No | Particulars | Weightage |
|---|---|---|
| 1 | Project | 30% |
| 2 | Documentation | 20% |
| 3 | Code Changes | 25% |
| 4 | VIVA | 25% |

**Web References:**

**1)** ionicframework.com/docs
**2)** https://visualstudio.microsoft.com/xamarin/
**3)** https://github.com/xamarin
**4)** https://www.djangoproject.com/\
**5)** https://www.djangoproject.com/

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Application Development using SPRING (ADS)**
**Subject Code: 4659305**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To understand the Spring framework.
- To understand Spring container, Modules, Dependency Injection and aspect oriented programming.
- To understand how to design and develop application using Spring.

## 2. Prerequisites: Basics of JAVA Programming, Exposure to J2EE Technology.

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to Spring:**<br><br>What is Spring?, Initializing a Spring application (Initializing a Spring project with Spring Tool Suite, Examining the Spring project structure),<br>Writing a Spring application (Handling web requests, Defining the view,<br>Testing the controller, Building and running the application, Getting to know Spring Boot DevTools), The core Spring Framework (Spring Boot, Spring Data, Spring Security, Spring Integration and Spring Batch, Spring Cloud)<br><br>**Developing Application:** Displaying information(Establishing the domain, Creating a controller class, Designing the view), Processing form submission, Validating form input (Declaring validation rules, Performing validation at form binding, Displaying validation errors), Working with view controllers, Choosing a view template library | **10%** |
| **Unit II** | **Working with Data:**<br><br>Reading and writing data with JDBC (Adapting the domain for persistence, Working with JdbcTemplate, Defining a schema and preloading data, Inserting data), Persisting data with Spring Data JPA(Adding Spring Data JPA to the project, Annotating the domain as entities, Declaring JPA repositories, Customizing JPA repositories) | **15%** |
| **Unit III** | **Securing Spring**<br><br>Enabling Spring Security, Configuring Spring Security(In-memory user store, JDBC-based user store, LDAP-backed user store, Customizing user, authentication), Securing web requests (Securing requests, Creating a custom login page, Logging out, Preventing cross-site request forgery), Knowing your user<br><br>Fine-tuning autoconfiguration (Understanding Spring's environment abstraction, Configuring a data source, Configuring | **15%** |

| | | |
|---|---|---|
| | the embedded server, Configuring logging, Using special property values), Creating your own configuration properties(Defining configuration properties holder , Declaring configuration property metadata), Configuring with profiles(Defining profile-specific properties, Activating profiles, Conditionally creating beans with profiles) | |
| **Unit IV** | **Advanced Spring**<br><br>Creating REST services:Writing RESTful controllers( Retrieving data from the server,Sending data to the server, Updating data on the server,Deleting data from the server),Enabling hypermedia ( Adding hyperlinks,Creating resource assemblers, Naming embedded relationships), Enabling data-backed services (Adjusting resource paths and relation names,Paging and sorting, Adding custom endpoints, Adding custom hyperlinks to Spring Data endpoints)<br><br>Consuming REST services: Consuming REST endpoints with RestTemplate( GETting resources, PUTting resources, DELETEing resources, POSTing resource data), Navigating REST APIs with Traverson<br><br>Sending messages with JMS (Setting up JMS, Sending messages with JmsTemplate, Receiving JMS messages), Working with RabbitMQ and AMQP (Adding RabbitMQ to Spring, Sending messages with RabbitTemplate,Receiving message from RabbitMQ), Messaging with Kafka(Setting up Spring for Kafka messaging, Sending messages with KafkaTemplate, Writing Kafka listeners)<br><br>Declaring a simple integration flow(Defining integration flows with XML,Configuring integration flows in Java, Using Spring Integration's DSL configuration), Surveying the Spring Integration landscape( Message channels, Filters, Transformers, Routers, Splitters, Service activators,Gateways , Channel adapters, Endpoint Modules), Creating an email integration flow | **30%** |
| **Unit V** | **Reactive Spring**<br><br>**Understanding reactive programming**: Defining Reactive Streams, Getting started with Reactor( Diagramming reactive flows, Adding Reactor Dependencies), Applying common reactive operations(Creating reactive types, Combining reactive types, Transforming and filtering reactive streams, Performing logic operations on reactive types)<br><br>**Developing reactive APIs:** Working with Spring WebFlux (Introducing Spring WebFlux, Writing reactive controllers), Defining functional request handlers, Testing reactive controllers (Testing GET requests, Testing POST requests, Testing with a live | **30%** |

| | | |
|---|---|---|
| | server), Consuming REST APIs reactively (GETting resources, Sending resources, Deleting resources, Handling errors, Exchanging requests), Securing reactive web APIs (Configuring reactive web security, Configuring a reactive, user details service)<br><br>**Persisting data reactively**: Understanding Spring Data's reactive story (Spring Data reactive distilled, Converting between reactive and non-reactive types, Developing reactiverepositories), Working with reactive Cassandra repositories, Enabling Spring Data Cassandra,Understanding Cassandra data modeling, Mapping domain types for Cassandra persistence, Writing reactive Cassandra repositories, Writing reactive MongoDB repositories (Enabling Spring Data MongoDB,Mapping domain types to documents, Writing reactive MongoDB repository interfaces) | |

**4. Text Book:**

1) Craig Walls, "Spring in Action", Fifth Edition, Manning, ISBN 9781617294945

**5. Reference Books:**

1) Santosh Kumar K., "Spring and Hibernate", Tata McGraw-Hill Publishing, 2009,ISBN 978-0070680111
2) Paul Tepper Fisher and Brian D. Murphy, "Spring persistence with Hibernate", Apress,2010, ISBN 978-1-4302-2632-1
3) Amritendu De, "Spring 4 and Hibernate 4: Agile Java Design and Development", McGraw-Hill Education,2015, ISBN: 9780071845113
4) Chris Schaefer, Clarence Ho, and Rob Harrop ,Pro Spring. Apress

**6. Chapter wise Coverage from Text Book(s):**

| Unit No. | Text Books | Topics/Subtopics |
|---|---|---|
| I | Book-1 | Chapter : 1,2 |
| II | Book-1 | Chapter : 3 |
| III | Book-1 | Chapter : 4,5 |
| IV | Book-1 | Chapter : 6,7,8,9 |
| V | Book-1 | Chapter : 10,11,12 |

**7. Accomplishments of the student after completing the course:**

- Students will understand Spring framework.
- Student will understand how to design and develop web application using Spring.

## 8. Laboratory Exercises

**A.** General Guidelines

- Group: 2-3 Person.
- The project should be free from plagiarism of any kind.
- It is mandatory that the project should be developed using Spring 3 or later version on Linux or Windows Platform.
- This may not be a live project

### B. Expected Outcome

- The objective of the Web Application Development using Spring is to make students aware about the industry based process and workings. As a result, Project must meet with the industry standards.
- There will not be any compulsion to prepare a project report for the students but an application should be demonstrated, so that evaluator may get the detail about the Project developed and can evaluate the students as per the evaluation criteria.

### C. Evaluation

| Sr. No. | Particulars | Weightage |
|---|---|---|
| 1 | **Project** | **40%** |
| 2 | **Code Changes** | **40%** |
| 3 | **VIVA** | **20%** |

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Big Data Analytics (BDA)**
**Subject Code: 4659306**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To be able to understand various techniques of Big Data Analytics
- To understand how to explore and communicate data using data visualization techniques

## 2. Prerequisites: Knowledge of R/Python and Database concepts

## 3. Recommended content knowledge

| Sr. No. | Course Content | Weightage |
|---|---|---|
| 1 | **Introduction**<br>Big Data Overview, BI versus Data Science, Current analytical architecture, Drivers of Big Data, Emerging Big Data Ecosystem and a New Approach to Analytics, Key Roles for the New Big Data Ecosystem, Examples of Big Data Analytics<br><br>Data Analytics Life Cycle Overview, Phases ( Discovery, Data Preparation, Model Planning, Mode Building, Communicate Results, Operationalize)<br><br>**Case Study: Global Innovation Network and Analysis ( GINA)** | 15% |
| 2 | **Mining Relationships Among Records**<br><br>**Association Rules :** Discovering Association rules in transaction Databases, Generating Candidate Rules, The apriori algorithm, Selecting strong rules, Data Formats, The process of Rule selection, Interpreting results, Rules and chance<br><br>**Collaborating Filtering:** Data and Format, User based collaborative filtering "People like you", Item-based Collaborative Filtering, Advantages and weaknesses of Collaborative filtering, Collaborative filtering vs Association Rules<br><br>**Cluster Analysis:** Introduction, measuring distance between two records, Measuring distances between two clusters**,** Hierarchical (Agglomerative) Clustering, Non-Hierarchical Clustering: The k-Means Algorithm | **25%** |
| 3 | **Forecasting Time Series**<br><br>**Handling Time Series:** Introduction, Descriptive vs. Predictive Modeling, Popular Forecasting Methods in Business, Time Series Components, Data-Partitioning and Performance Evaluation<br><br>**Regression-Based Forecasting :** A Model with Trend, A Model with Seasonality, A Model with Trend and Seasonality, Autocorrelation and ARIMA Models<br><br>**Smoothing Methods:** Introduction, Moving Average, Simple Exponential Smoothing, Advanced Exponential Smoothing | 25% |

| | | |
|---|---|---|
| 4 | **Social Network Analysis**<br><br>Introduction, Directed vs. Undirected Networks, Visualizing and Analyzing Networks, Social Data Metrics and Taxonomy, Using Network Metrics in Prediction and Classification, Collecting Social Network Data with R, Advantages and Disadvantages | 15% |
| 5 | **Text Mining**<br><br>Introduction, The Tabular Representation of Text: Term-Document Matrix and "Bag-of-Words" , Bag-of-Words vs. Meaning Extraction at Document Level, Preprocessing the Text, Implementing Data Mining Methods ,<br><br>Example: Online Discussions on Autos and Electronics | **20%** |

## 4. Main Reference Books:

1. EMC Education Services, Data Science and Big Data Analytics, WILEY
2. Galit Shmueli, Peter C Bruce, Inbal Yahav, Nitin R Patel, Kenneth C, Linchtendahl Jr, Data Mining for Business Analytics- concepts, techniques and Application in R, WILEY

## 5. Recommended Book(s):

1. Glenn J Myatt, Wayne P Johnson, Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining, Wiley, 2nd Edition
2. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press (Wiley India) , 2nd Edition
3. Mehmed Kantardzic, Data Mining: Concetps, Models, Methods and Algorithms, Wiley-IEEE, 2nd Edition
4. Field Cady, 'The Data Science Handbook The Data Science Handbook', Wiley Publication ISBN-13: 978-8126573332
5. Han, J., Kamber, M., Pei, J. Data mining concepts and techniques. Morgan Kaufmann, 2011
6. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007
7. Vincent Granville, Developing Analytic Talent: Becoming a Data Scientist, wiley, 2014
8. John W. Foreman (Author), Data Smart: Using Data Science to Transform Information into Insight, WILEY

## 6. Chapter wise Coverage from Main Book(s):

| Unit# | Book# | Topics |
|---|---|---|
| **1** | **1** | **Chapter 1,2** |
| **2** | **2** | **Chapter 14,15** |
| **3** | **2** | **Chapter 16, 17,18** |
| **4** | **2** | **Chapter 19** |
| **5** | **2** | **Chapter 20** |

## 7. Suggested Practical

**Tool: Python, Libraries of Python like Pandas, Sci-kit Learn etc., R ( R Studio and required packages)**

**Part 1: Data Pre-processing:**

Dataset:https://www.analyticsvidhya.com/blog/2016/07/practical-guide-data-preprocessing-python-scikit-learn/

1. Download loan data set (https://www.analyticsvidhya.com/blog/2016/07/practical-guide-data-preprocessing-python-scikit-learn/) and perform following operations
   i. Write program to read dataset ( Text,CSV,JSON,XML)
   ii. Performing Data Cleaning
      a. Handling Missing Data
      b. Removing Null data
      c. Rescaling Data
   iii. Dimensionality Reduction
   iv. Encoding Data
   v. Feature Selection
   vi. Implement Principle Component Analysis,
2. Use Loan data (above) and Fit KNN model to find out accuracy of model for prediction of loan.
3. Write a python code to predict profit of hotel chain given the population of the area (city) using the data at https://docs.google.com/spreadsheets/d/1Ks20skBgEefHFU36sFqVzozoFtz2EZE2rxB_IgXOrUg/edit?usp=sharing.
4. Write a python code to predict the price of house given square feet and number of bed rooms in the house for the dataset available at https://docs.google.com/spreadsheets/d/1DHVK7gKo4TSyj7mFLwofHamj1Sl4SOZma2q51w1ZvyE/edit?usp=sharing

**Part 2: Mining Relationships among Records**

5. Implement Apriori algorithm in python to find rules which explain association between different products for given transactions at a retail store. (The data is available at https://drive.google.com/file/d/1NUXoptUlHY8z4KcFKpFA6sQN5KnWzk3p/view?usp=sharing )
6. Implement text classification using neural network in python/R on Twenty Newsgroup dataset from UCI machine learning repository.
7. Generating Association rule mining e.g "Sythentic Data on Purchase of Phone faceplate"
   a. Recommender algorithms: Generating rules for Similar Book Purchases
8. Collaborative Filtering (use movielens dataset):
   a. Find similar items by using a similarity metric
   b. For a user, recommend the items most similar to the items (s)he already likes

**Part 3: Implement Clustering**

9. Implement clustering algorithm for grouping news articles.

10. Implement unsupervised machine learning algorithm (Clustering – K Means) in python on Titanic dataset to cluster data (use Titanic dataset) by removing the class label.
11. Implement unsupervised machine learning algorithm (Clustering – K Means) in python on Breast Tumour dataset to cluster data (use Breast Tumour dataset) by removing the class label.
12. Implement unsupervised machine learning algorithm (Clustering – Hierarchical) in python on Titanic dataset to cluster data (use Titanic dataset).

**Part 4: Various types of Text Analysis**

13. For the sentiment analysis dataset given in link https://drive.google.com/file/d/1x6H7_KJjkbDrpgZFS7I2wjsZsILeSJ4S/view?usp=sharing, implement the following in python,
    b. Clean and pre-process the dataset by removing URL, removing HTML tags, handling negation words which are split into two parts, converting the words to lower cases, removing all non-letter characters
    c. Split the dataset into training and testing set
    d. Implement feature extraction technique (to convert textual data to the numeric form)
    e. Build the classification model using Logistic Regression that classifies if a given sentiment text is positive or negative
    f. Obtain the accuracy score of the built model.

14. Implement a content based recommender system in python that recommends movies that are similar to a particular movie using movielens-20m-dataset available at https://kaggle.com.

**Part 5: Advanced Data Visualization**

15. Write a program to plot Chi square distribution
16. Write a program to plot Normal distribution
17. Write a program to plot Poisson distribution
18. Write a program to plot T distribution
19. Write a program to plot Binomial Distribution
20. Write a program to plot Central limit theorem
21. Write a program to plot Uniform distribution

**Part 6: Text pre-preprocessing using Python**

**Tools: NLTK (** http://www.nltk.org/) sci-kitlearn etc.

22. Removing stop words (the most common words in a language like “the”, “a”, “on” etc.)
23. Write a python code to perform spell check (edit distance algorithm)
24. Write a python code for finding the root words ( Stemming algorithm : A stemming algorithm reduces the words "fishing", "fished", and "fisher" to the root word, "fish)
25. Write a python code to implement Tokenized algorithm for text processing
26. Write a python code Part of speech (PoS) tagging

**Part 7. Desirable:**

Abstractive/ Extractive text Summarization (single document, multi document)
Time series algorithm

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Application Development using LARAVEL (ADL)**
**Subject Code: 4659307**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To understand the LARAVEL framework
- To understand how to Implement LARAVEL
- To understand how to design and develop responsive web application using LARAVEL

## 2. Prerequisites: HTML, CSS and Basics of PHP

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to LARAVEL**<br><br>Why Use a Framework? A Short History of Web and PHP Frameworks, What's So Special About Laravel?, How It Works, Why Laravel?<br><br>Setting Up a Laravel Development Environment: System Requirements, Composer, Local Development Environments, Creating a New Laravel Project, Laravel's Directory Structure, Configuration<br><br>An Introduction to Artisan, Basic Artisan commands, Writing custom artisan commands, calling artisan commands in Normal code, Tinker | **15%** |
| **Unit II** | **LARAVEL Basics**<br><br>**Router and Controllers:** Route Definitions, Route Groups, Views, Controllers, Route Model Binding, Route Caching, Form Method Spoofing, CSRF Protection, Redirects, Aborting the Request, Custom Responses<br><br>**Blade Template :** Echoing Data, Control Structures ( Conditionals, Loops, Or), Template Inheritance, View Composers and Service Injection, Custom Blade Directives<br><br>**Front End Components:** Elixir, Pagination, Message Bags, String Helpers, Pluralization and localization | **25%** |
| **Unit III** | **Collecting and Validating User Data**<br><br>Injecting a Request Object, Route Data, Uploaded Files, Validation, Form Requests, Eloquent Model Mass Assignment, {{ versus {!! Auth Controller<br><br>**Database Eloquent:** configuration, Migration, Seeding, Query Builder, Introduction to Eloquent, Eloquent Events<br><br>**User Authentication and Authorization :** The User Model and | **30%** |

| | | |
|---|---|---|
| | Migration, Using the auth() Global Helper and the Auth Facade | |
| **Unit IV** | **Advanced LARAVEL**<br><br>**Request and Response:** Laravel's Request Lifecycle, The Request Object, The Response Object, Laravel and Middleware<br><br>**Writing APIs :** The Basics of REST - Like JSON APIs, Controller Organization and JSON Returns, Reading and Sending Headers, Eloquent Pagination, Sorting and Filtering, Transforming Results | **15%** |
| **Unit V** | **Mail , Notifications**<br><br>**Storage and Retrieval:** Basic File Uploads and Manipulation, Sessions, Cache, Cookies, Full-Text Search with Laravel Scout | **15%** |

**Desirable Topics:**

**1)** GitHub - akaunting/akaunting: Free and Online Accounting Software (url: https://github.com/akaunting/akaunting)

**4. Text Book:**

1) Matt Stauffer, "LARAVEL Up and Running, A framework for building modern PHP Apps", O'REILLY , 3rd Indian Reprint ( ISBN: 978-93-5213-485-4)

**5. Reference Books:**

1) Martin Bean, "Laravel 5 Essentials", Packet Publishing, ISBN 978-1-78528-301-7
2) Fernando Monteiro, "Hands-On Full-Stack Web Development with Angular 6 and Laravel 5 ", Packt Publishing, ISBN 9781788833912
3) Web Technologies : HTML,CSS3, JAVASCRIPT, jQUERY,AJAX, PHP,XML, MVC and LARAVEL), Black Book, 2018, Dreamtech , ISBN 9789386052490

Webliography:

1) Online Laravel 5.2 Documentation (https://laravel.com/docs/5.7)
2) Nathan Wu, Learning Laravel 5 Cookbook (https://learninglaravel.net/laravelbook)
3) https://laravel-news.com/category/laravel-tutorials
4) https://laravel.com

**6. Chapter wise Coverage from Text Book(s) :**

| Unit | Text Books | Topics |
|---|---|---|
| I | Book-1 | Chapter : 1,2,7 |
| II | Book-1 | Chapter : 3,4,5 |
| III | Book-1 | Chapter : 6,8, Chapter : 9 ( page 197 to 205) |
| IV | Book-1 | Chapter : 10,13( Page 283 to 295) |
| V | Book-1 | Chapter : 14( Page 318 to 331) |

## 7. Accomplishments of the student after completing the course:

- Students will understand LARAVEL framework.
- Student will understand how to design and develop responsive web application using LARAVEL.

## 8. Laboratory Exercises

A. General Guidelines

- Group: 2-3 Person.
- The project should be free from plagiarism of any kind.
- It is mandatory that the project should be developed using LARAVEL 5.6 or later version on Linux or Windows Platform.
- This may not be a live project
- Use any database for storing data.

B. **Expected Outcome**

- The objective of the LARAVEL Project Development is to make students aware about the industry based process and workings. As a result, Project must meet with the industry standards.
- There will not be any compulsion to prepare a project report for the students but an application should be demonstrated, so that evaluator may get the detail about the Project and can evaluate the students as per the evaluation criteria.

**C. Evaluation**

| Sr. No. | Particulars | Weightage |
|---|---|---|
| **1** | **Project** | **40%** |
| **2** | **Code Changes** | **40%** |
| **3** | **VIVA** | **20%** |

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Advanced Networking (AN)**
**Subject Code: 4659308**

**With effective from academic year 2019**

## 1. Learning Objectives:

- Develop strong analysis, design, implementation, testing and troubleshooting skills in students regarding TCP/IP based networks and services as relevant to the computer networking needs of the IT industry.
- Establish a strong conceptual foundation of the TCP/IP protocol stack, services and related tools/technologies so as to facilitate the development of the above mentioned skills.
- Design and implement customized TCP/IP based application layer services.
- To familiarize with security and performance issues in TCP/IP networks
- To familiarize with Wireless Networks, WiFi and Mobile Networks, Browser Networking,XMLHttpRequest and Server-Sent Events (SSE) and WebSocket and WebRTC
- Create a strong conceptual foundation and offer maximum possible development of required theoretical and practical skills for students aspiring to make a career in Computer Networking like Network Designer, Network administrator, etc.
- To learn how to listen to local network traffic and analyze different protocols.

## 2. Prerequisites:

Digital Data communication concepts, Layered architecture as per OSI and TCP/IP model, Functionality of all layers in the OSI and TCP/IP model, Concepts of LAN, WAN, Internet, HTTP, Ethernet, General concepts in routing and basic routing algorithms like Djkistra's shortest path, distance vector routing, link state routing, etc., Overview of popular application layer services like HTTP, DNS, FTP etc.

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Primer on Latency and Bandwidth, Building Blocks of TCP and UDP and Transport Layer Security (TLS).** Speed Is a Feature, Components of Latency, Speed of Light and Propagation Latency, Last-Mile Latency, Bandwidth, Delivering Higher Bandwidth and Lower Latencies, TCP Three Way Handshaking, Congestion Avoidance and Control, Bandwidth Delay Product, Optimization for TCP,UDP and Network Address Translator, NAT Traversal, STUN, TURN and ICE. TLS Handshake, TLS Session Resumption, Chain of Trust and Certificate Authorities, Certificate Revocation, TLS Record Protocol and Optimizing for TLS | 30 % |
| **Unit II** | **Wireless Networks, WiFi** Ubiquitous Connectivity, Types of Wireless Networks, Performance Fundamentals of Wireless Networks, From Ethernet to a Wireless LAN, WiFi Standards and Features, Measuring and Optimizing WiFi Performance, Optimizing for WiFi Networks. | 10 % |
| **Unit III** | **Browser Networking, XMLHttpRequest and Server-Sent Events (SSE)** Primer on Browser Networking, XMLHttpRequest: Brief History of XHR, Cross-Origin Resource Sharing (CORS), Downloading and | 30 % |

| | | |
|---|---|---|
| | uploading Data with XHR, Monitoring Download and Upload Progress, Streaming Data with XHR, Server-Sent Events (SSE): EventSource API and Event Stream Protocol. | |
| **Unit IV** | **WebSocket and WebRTC**<br>Introduction to WebSocket, WebSocket API, WebSocket Protocol, WebSocket Use Cases and Performance, WebRTC: Standards and Development of WebRTC, Audio and Video Engines, Real-Time Network Transports, Establishing a Peer-to-Peer Connection, Delivering Media and Application Data, DataChannel, WebRTC Use Cases and Performance. | 30 % |

**4. Text Book:**

**1)** Ilya Grigorik, "High-Performance Browser Networking", 2013: First Edition, O'Reilly E-book book also available https://hpbn.co/

**5. Reference Books:**

**1)** Douglas E. Comer, "Internetworking with TCP/IP - (Vol. 1) Principles, Protocols, and Architecture", 6th Edition, Prentice Hall of India (PHI) Publishers.
**2)** Behrouz A. Forouzan, "TCP/IP Protocol Suite", 4th Edition, McGraw-Hill
**3)** W. Richard Stevens, G. Gabrani, "TCP/IP- Illustrated, Vol. 1 (The Protocols)", Pearson Publishers.
**4)** Chris Sanders, "Practical Packet Analysis using Wireshark to solve real world problems", No Starch Press, Inc.

**6. Chapter wise Coverage from the Text Book:**

| Unit No. | Text Books | Topics/Subtopics |
|---|---|---|
| 1 | Book-1 | Chapter 1 to Chapter 4 |
| 2 | Book-1 | Chapter 5, Chapter 6 |
| 3 | Book-1 | Chapter 14 to Chapter 16 |
| 4 | Book-1 | Chapter 17 to Chapter 18 |

**7. Accomplishment of the student after completing the course**:

**1)** Have thorough understanding of TCP/IP based systems, services and related tools and technologies
**2)** Be fluent in design and developing TCP/IP socket based networking solutions
**3)** Effectively use available OS commands/utilities as well as popular third party tools for TCP/IP networking depending upon the needs
**4)** Be geared to adapt to more sophisticate networking related packages and hence develop relatively complex applications more reliably and faster.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Advanced Networking (AN)**
**Subject Code: 4659308**

**With effective from academic year 2019**

**Practical List**

1. Download Wireshark is a network packet analyzer from its official webpage https://www.wireshark.org/
2. Install Wireshark under Windows/Linux/MAC/Solaris platform
3. Wireshark Installation, understanding the Wireshark Environment / Menu System, to start and stop live capture of traffic from given wired ethernet network interface, capturing options, store captured data in different supported file formats, to open already stored captured data file.
4. Analyze Transmission Control Protocol (TCP) Traffic Established of TCP Connections, Termination of TCP Connections, How TCP Tracks Packets Sequentially, Review the Trace File: Packet Loss Detected by the Receiver – Fast Recovery, Packet Loss Detected by the Sender – RTO Timeout, Improve Packet Loss Recovery with Selective Acknowledgments, TCP Flow Control, The TCP Window Size > Zero Can Still Stop Data Transfer
5. Analyze User Datagram Protocol (UDP) Traffic : Normal UDP Traffic, UDP Problems, Dissect the UDP Packet Structure, Filter on UDP Traffic
6. Analyze IEEE 802.11 (WLAN) : Wireless LANs (WLANs) Traffic, Signal Strength and Interference, Capture WLAN Traffic, Monitor Mode vs. Promiscuous Mode, 802.11 Traffic Basics like Data Frame, Management Frame and Control Frames etc., Normal 802.11 Communications, Dissect the 802.11 Frame Structure, Filter on All WLAN Traffic, Frame Control Types and Subtypes
7. Capture Wi-FI and Bluetooth Traffic and Interpret/ Analyze the corresponding header and payload using Wireless Traffic Sniffing tools like WireShark-USB/AirCrackng/Kismet, etc.
8. Write a java socket program for implementation of echo (Algorithms: Client Side: Create a socket which binds the IP address of server and the port address to acquire service, After establishing connection send a data to server, Receive and print the same data from server, Close the socket and End the program, Server Side: Create a server socket to activate the port address, Create a socket for the server socket which accepts the connection, After establishing connection receive the data from client, Print and send the same data to client, Close the socket and End the program).
9. Write a java client-server application for chat using TCP/IP (Algorithms: Client Side: Start writing the program, include all necessary package in java, create a socket in client to server, client establishes a connection to the server, client accept the connection and to send the data from client to server, client communicates the server to send the end of the message and Stop the program, Server Side: Include all necessary package in java, create a socket in server to client, server establishes a connection to the client, server accept the connection and to send the data from server to client and vice versa, server communicate the client to send the end of the message and Stop the program).
10. Write a java program to Perform File Transfer in Client & Server Using TCP/IP (Algorithms: Client Side: Establish a connection between the Client and Server, Socketss=new Socket(InetAddress.getLocalHost(),1100), Implement a client that can send two requests (i) To get a file from the server (ii) To put or send a file to the server, After getting approval from the server ,the client either get file from the server or send, file to the server, Server Side: Implement a server socket that listens to a particular port number, Server reads the filename and sends the data stored in the file for the 'get' request, reads the data from the input stream and writes it to a file in the server for the 'put' instruction, Exit upon client's request and Stop.

11. Write a java program to implement simple client-server application using UDP (Algorithms: Client Side: Create a datagram socket with server's IP address, Create datagram packets with data, data length and the port address, Send the datagram packets to server through datagram sockets, Receive the datagram packets from server through datagram sockets and Close the socket, Server Side: Create a datagram socket with port address, Create datagram packets with data, data length and the port address, Send the datagram packets to client through datagram sockets,Receive the datagram packets from client through datagram sockets and Close the socket.
12. Implement real-time video streaming and data exchange using Web RTC (i) Get video from your webcam (ii) Stream video with RTCPeerConnection (iii) Stream data with RTCDataChannel (iv) Set up a signaling service to exchange messages (v) Combine peer connection and signalling and (vi) Take a photo and share it via a data channel.
    Download the sample source code and Learn Step by Step:
    https://codelabs.developers.google.com/codelabs/webrtc-web/#9
    The practical examination exercises would be based on lab exercises above list except Lab # 12. For Lab # 12 a documentation/report/file should be maintained.

**References:**

https://www.wireshark.org/
https://www.wireshark.org/download/docs/user-guide.pdf
https://www.aircrack-ng.org/
https://www.kismetwireless.net/
https://webrtc.org/

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Image Processing (IP)**
**Subject Code: 4659309**

**With effective from academic year 2018-19**

1. **Learning Objectives:**
   - To understand basic concepts and methodologies for digital image processing
   - To develop a foundation that can be used as the basis for further study and research in this field.
   - To provide understanding of the different types of image representations, enhancing image characteristics, image filtering, and reducing the effects of noise and blurring in an image.
   - To understand image processing needed for extracting information from an image.

2. **Prerequisites:** Knowledge of Computer Graphics is desirable.

3. **Course Contents:**

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction to Digital Image Processing and Fundamental**<br>Introduction: What is Digital Image Processing? Fundamental steps in Digital Image processing, Components of Image Processing system<br>Digital Image Fundamentals- Some basic relationships like Neighbours, Connectivity, and Distance Measures between pixels. Overview of mathematical tools used in digital image processing | 10% |
| **Unit II** | **Transformations, Histogram Processing and Spatial Filtering**<br>Image Enhancement in the spatial domain: Background, Some basic Gary Level Transformations, Histogram Processing, Fundamentals of spatial filtering, Smoothening and Sharpening Spatial Filters<br>Filtering in the frequency Domain: Background, Introduction to Fourier Transform (FT) and frequency domain, Computing and Visualizing the Discrete Fourier Transform (DFT) of one variable, Extension to functions of two variables - 2D DFT, Image Smoothing and Sharpening Using Frequency Domain Filters | 30% |
| **Unit III** | **Image Restoration**<br>Image Restoration: A model of the Image Degradation/Restoration process, Noise Models, Restoration in the presence of noise only - Spatial filtering | 15% |
| **Unit IV** | **Morphological Image Processing**<br>Morphology: Dilation, Erosion, Opening and Closing, The Hit-or-Miss Transformation, Morphological Algorithms: Boundary Extraction, Region filling, Extraction of connected components, Convex Hull, Thinning, Thickening, Skeletons, Pruning, Morphological reconstruction | 20% |
| **Unit V** | **Image Segmentation and Object Recognition**<br>Image Segmentation: Fundamentals, Point, Line and Edge Detection, Region Based Segmentation<br>Object Recognition : Pattern and Pattern Classes, Recognition Based on Decision Theoretic Methods- Matching, Optimal Statistical Classifier, Neural Networks, Object recognition based on structural methods – Matching Shape Numbers | 25% |

**4. Text Book:**

**1)** Richard E Woods, Rafael C Gonzalez , "Digital Image Processing", Pearson, 3rd Edition

**5. Reference Books:**

**1)** Anil K. Jain, "Fundamentals of Digital Image Processing", Prentice-Hall India, 1989
**2)** B. Chanda and D. Datta Majumder, "Digital Image Processing and Analysis", Prentice-Hall India, 2nd edition (October 30, 2011)
**3)** Madhuri A. Joshi, "Digital Image Processing", Prentice-Hall India, 2006

**6. Chapter wise Coverage from the Text Book:**

| Unit # | Chapter |
|---|---|
| I | Chapter 1: 1.1, 1.4,1.5<br>Chapter 2: 2.5, 2.6 |
| II | Chapter 3: 3.1, 3.2, 3.4, 3.5, 3.6<br>Chapter 4 : 4.1,4.2,4.4,4.5,4.8,4.9 |
| III | Chapter 5: 5.1 ,5.2,5.3 |
| IV | Chapter 9: 9.1,9.2,9.3,9.4, 9.5 |
| V | Chapter 10: 10.1, 10.2, 10.4<br>Chapter 12: 12.1, 12.2, 12.3.1 |

**7. Accomplishment of the student after completing the course**:

**1)** Understanding of the principals the Digital Image Processing and terminology used to describe features of images.
**2)** Understanding of the mathematical foundations for digital manipulation of images; image acquisition; pre-processing; segmentation; Fourier domain processing.
**3)** Be able to write programs for implementing image processing tasks.
**4)** Learn and understand the Image Enhancement techniques.
**5)** Learn and understand Image Segmentation and Recognition concepts.

## Practical List

**Objective:**

Learning the use of Python and OpenCV to implement basic image processing algorithms and to build and execute image processing based projects to solve real life problems

**Prerequisites**: Knowledge of OpenCV and Python

**Lab Experiments:**

1. Getting started with images
    a. Learn to load an image, display it and save it back
2. Drawing functions in OpenCV
    a. Draw lines, rectangles, ellipses, circles, ellipses, polygons, adding text to images
3. Perform Basic operations on images
    Read and edit pixel values, working with image -other basic operations.
4. Perform Arithmetic operations on images
5. For a sample images change images between different color spaces
6. Showing images in an OpenCV window
    Read, write, view images and conversion between different formats.
7. Write code to perform following:
    - Loads 2 images (Image 1 say I1 and Image 2 say I2)
    - Computes the pixel-wise difference between the two images
    - Computes an output image where each pixel of coordinates (x,y) contains the absolute difference of the corresponding pixels on I1 and I2
    - Out(x,y) = abs(I1(x,y) – I2(x,y))
    - Displays output image in a window
8. Write code to change brightness of the colour image and show negative of an image.
9. Histograms-1: Find, Plot, Analyze
    Find and draw contours
10. Histograms-2: Histogram Equalization
    Equalize histograms to get better contrast for images
11. Histograms-3: 2-D Histograms
    Find and plot 2-D histograms
12. Apply different Geometric transformations to images like rotation, translation, crop
13. Apply various Scaling operations on the image – resize, down size & upsize (preserve aspect ratio) , resize only width, resize only height, resize to fixed height and width
14. Convert images to binary images using global thresholding, adaptive thresholding, Otsu's binarization.
15. Blur the images, filter the images with custom kernels.
16. Find the Fourier Transform of images using OpenCV using the FFT functions available in Numpy.
17. OpenCV provides variations to remove Noise
    cv2.fastNlMeansDenoising()—works with a single grayscale images
    cv2.fastNlMeansDenoisingColored()—works with a color image.
    Use these functions to denoise grayscale and colour images.
18. Do you have an old degraded photo with many black spots and strokes on it? Take it. Try to restore it with a technique called image inpainting.
19. Perform Morphological Transformations - Erosion, Dilation, and Opening, Closing on a sample image.

20. Find Image Gradients.
21. Find edges with Canny Edge Detection.
22. Apply Hough Line Transform to Detect lines in an image.
23. Apply Hough Circle Transform to Detect circle in an image.
24. Apply Watershed Algorithm and k-means algorithm for Image Segmentation.
25. Search for an object in an image using Template Matching.
26. Detect QR code.
27. Detect text in natural scenes.

**References:**

1) Alexey Spizhevoy, Aleksadr Rybnikov, "OpenCV3 Computer Vision with Python Cookbook", Packt Publishing Ltd., 2018
2) https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_colorspaces/py_colorspaces.html
3) OpenCV-Python Tutorials Documentation Release 1 , Alexander Mordvintsev & Abid K, Nov 05, 2017, https://media.readthedocs.org/pdf/opencv-python-tutroals/latest/opencv-python-tutroals.pdf
4) https://codewords.recurse.com/issues/six/image-processing-101
5) https://pythonprogramming.net/image-operations-python-opencv-tutorial/
6) http://www.imageprocessingplace.com/root_files_V3/image_databases.htm ( to obtain sample images)

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Operations Research (OR)**
**Subject Code: 4659310**

**With effective from academic year 2018-19**

1. **Learning Objectives:**

- To get a basic understanding of Operation Research Techniques for optimum utilization of constrained resources in wide range of areas including industry, business, commerce, administration, management, service supply, maintenance, agriculture, medicines and healthcare, defense etc.
- For a given problem statement, students will be able to develop ability to
  - Classify the class of problem (LPP or Transportation or Assignment ... etc),
  - Formulate the appropriate OR model,
  - Find the solution, and
  - Interpret the results

**NOTE: Mathematical derivations are not included for any topic identified.**

2. **Prerequisites:** Basic knowledge of Mathematics and Probability Distributions

3. **Contents:**

| Unit | Chapter Details | Weightage Percentage |
|---|---|---|
| **Unit I** | **Basics of Operations Research, Transportation Problem & Assignment Problem**<br>**(a) Basics of Operation Research**<br>Introduction, definitions, features, advantages and applications<br>**(b) Transportation problem (T.P.)**<br>Formulation of a T.P., Methods to find initial basic feasible solution: NorthWest corner rule, Least cost cell entry method, Vogel's Approximation method, Test of optimality for finding an optimum solution – MODI method. Variations of Transportation Problems<br>**(c) Assignment problem (A.P.)**<br>Formulation of an Assignment Problem, Method to find an optimum solution - Hungarian Assignment Method, Variations of assignment problem | **20%** |
| **Unit II** | **Management of Inventory and Replacement**<br>**(a) Management of Inventory**<br>Introduction and terminology of the inventory management problem including Objective(s) and Constraints; Single Item Inventory Control without Shortages Model –I: EOQ model with constant rate of demand Model – II: EOQ model with different rates of demand.<br>**(b) Management of Replacement**<br>Definition, replacement of items that deteriorates, replacement of item that fails completely | **15%** |
| **Unit III** | **(a) Theory of Games**<br>Introduction, Two-Person Zero Sum game, Pure strategies (Minimax & Maximin principles) Games with Saddle Point, Rules to determine Saddle Point, Mixed Strategies, Rules of Dominance, Solution methods games without Saddle Point | **25%** |

| | | |
|---|---|---|
| | **(b) Queuing Theory**<br>Introduction, Queuing system and problem, transient and steady states, traffic intensity, probability distributions in queuing systems, single service queuing model(s).<br>**(c) Simulation**<br>Introduction, applications, Monte-Carlo Method, Simulation using Computers, Simulation of Inventory Problems, Queuing Problems, Investment problems | |
| **Unit IV** | **Project Management and Scheduling**<br>**(a) Project Management (CPM & PERT)**<br>Network concepts, components, rules for network construction, critical path method (CPM) and Project evaluation and Review Techniques (PERT)<br>**(b) Production Scheduling (Job Sequencing)**<br>Introduction, Johnson's algorithm for N jobs on 2 machines, Johnson's algorithm for N jobs on M machines | **20%** |
| **Unit V** | **Linear Programming Problem (L.P.P.)**<br>Linear Programming Problem (L.P.P.), Formulation of a L.P.P. with its components: objective function and constraints, optimal solution, slack, surplus and artificial variables, Graphical method, Simplex method, Big-M method, Primal & Dual problem definition | **20%** |

**Desirable Topic:** Decision Thoery (EMV criteria, EVPI, EPPI), Max-min principle, Minmax principle, Hurwitz's principle, Laplace principle (Chapter 11 from Main Text Book)

**4. Text Book:**

1. J. K. Sharma, "Operations Research – Theory and Application", 4th Edition, Macmillan Publishers India Ltd.

**5. Reference Books:**

1. Kanti Swarup, Gupta P.K. , Man Mohan, "Operations Research", Sultan Chand & Sons, New Delhi
2. Shah, Gor, Soni, "Operations Research", PHI
3. V. K. Kapur, "Operations Research – Problems & Solutions", Sultan Chand & Sons, New Delhi

**6. Chapter wise Coverage from Main Reference Book(s):**

| Unit No. | Text Books | Topics/Subtopics |
|---|---|---|
| 1 | Basics of Operation Research | Ch – 1 (1.1 to 1.5, 1.10, 1.13) |
| | Transportation Problem | Ch – 9 (9.1 to 9.5) |
| | Assignment Problem | Ch – 10 (10.1 to 10.4) |

| | | |
|---|---|---|
| 2 | Inventory Management | Ch – 14 (14.1 to 14.7) |
| | Replacement | Ch – 17 (17.1 to 17.4 ) |
| 3 | Game theory | Ch. 12 ( 12.1 to 12.6) |
| | Queuing theory | Ch. 16 ( 16.1 to 16.6) |
| | Simulation | Ch – 19 (19.1 to 19.12) |
| 4 | Project Management (CPM and PERT) | Ch – 13 (13.1 to 13.6) |
| | Job Sequencing | Ch – 20 (20.1 to 20.3, 20.5) |
| 5 | Basics of Linear Programming | Ch – 2 (2.1, 2.2,2.3,2.4, 2.6,2.7, 2.8) |
| | Graphical Method of LPP | Ch – 3 (3.1, 3.2, 3.3.1 to 3.3.3, 3.4) |
| | Simplex Method of LPP | Ch. 4 ( 4.1 to 4.6) |
| | Duality in LPP | Ch. 5 (5.1 and 5.2) |

**7. Accomplishments of the student after completing the course:**

- Ability to classify and formulate Operation Research problems.
- Ability to design and construct suitable optimization models and to find solution of real life problems from diverse fields.
- Ability to interpret results.

**8. Practical List**

Tools: R /R Studio

1. A company is involved in the production of two items (X and Y). The resources need to produce X and Y are twofold, namely machine time for automatic processing and craftsman time for hand finishing. The table below gives the number of minutes required for each item:
   Machine time Craftsman time

   Item X 13 20
   Y 19 29

   The company has 40 hours of machine time available in the next working week but only 35 hours of craftsman time. Machine time is costed at £10 per hour worked and craftsman time is costed at £2 per hour worked. Both machine and craftsman idle times incur no costs. The revenue received for each item produced (all production is sold) is £20 for X and £30 for Y.
   The company has a specific contract to produce 10 items of X per week for a particular customer.
   - Formulate the problem of deciding how much to produce per week as a linear program.
   - Solve this linear program graphically

2. Solve using the Simplex method the following problem:
   Maximize $Z = f(x,y) = 3x + 2y$
   subject to: $2x + y \leq 18$

$2x + 3y \leq 42$
$3x + y \leq 24$
$x \geq 0 , y \geq 0$

3. Solve using the Simplex method the following problem:
   Maximize $p = 2x - 3y + z$ Objective function
   subject to $x + y + z \leq 10$
   $4x - 3y + z \leq 3$
   $2x + y - z \leq 10$
   $x \geq 0, y \geq 0, z \geq 0$

4. Develop a generalized program to solve optimized **Transportation problem**. First develop the program for a balanced problem, make a copy of that program and then modify to take care of unbalanced problem. Ask number of sources and destinations and the costs of transportation from every source to every destination. Show allocation at every step, final allocation and total transportation cost.

5. Develop a generalized program to solve optimized **Assignment problem**. First develop the program for a balanced problem, make a copy of that program and then modify to take care of unbalanced problem. Ask number of sources and destinations and the costs of transportation from every source to every destination. Show allocation at every step, final allocation and total transportation cost.

6. A certain item costs Rs. 75 per tonne. The requirement is 8,000 tonnes per year and each time the stock is replenished there is a set – up cost of Rs. 600. The cost of carrying inventory has been estimated at 12.8 per cent of the value of the stock per year. Find out the optimal order quantity, number of orders required to be placed in a year, number of days between two successive orders and total variable inventory cost. Assume 360 days in a year.

7. A television repairman finds that the time spent on repairing each TV has an exponential distribution with a mean of 15 minutes. He repairs the sets in the order in which they arrive.
   The arrival of sets follows a Poisson distribution approximately with an average rate of 16 per 8 hour day. Find out for how many hours would the repairman be busy in a day, what is the average number of TV sets in the system and the average waiting time of a TV set in the system.

8. There are 5 workers and their work time to complete their jobs on different machines are given below. Develop a program to solve **Assignment problem** for minimum solution

| | Machine 1 | Machine 2 | Machine 3 | Machine 4 | Machine 5 |
|---|---|---|---|---|---|
| Worker 1 | 8 | 5 | 7 | 7 | 8 |
| Worker 2 | 9 | 5 | 6 | 7 | 8 |
| Worker 3 | 6 | 8 | 5 | 6 | 9 |
| Worker 4 | 8 | 10 | 7 | 6 | 5 |
| Worker 5 | 4 | 6 | 5 | 6 | 4 |

9. There are 5 salesman and each of them can work on any one of 5 districts. Table below shows average revenue generated by each of them. Develop a program to solve **Assignment problem** for maximization.

| | District 1 | District 2 | District 3 | District 4 | District 5 |
|---|---|---|---|---|---|
| Salesman 1 | 250 | 198 | 206 | 220 | 210 |
| Salesman 2 | 240 | 220 | 196 | 208 | 212 |
| Salesman 3 | 260 | 240 | 198 | 220 | 220 |
| Salesman 4 | 240 | 250 | 194 | 208 | 200 |
| Salesman 5 | 240 | 220 | 198 | 200 | 204 |

10. A television repairman finds that the time spent on his jobs has an exponential distribution with mean of 30 minutes. If he repairs sets in the order in which they came in, and if the arrival of sets follows a Poisson distribution approximately with an average rate of 10 per 8-hour day, what is the repairman's expected idle time each day? How many jobs are ahead of the average set just brought in?

11. On an average 96 patients per 24-hour day require the service of an emergency clinic. Also on an average, a patient requires 10 minutes of active attention. Assume that the facility can handle only one emergency at a time. Suppose that it costs the clinic Rs 100 per patient treated to obtain an average servicing time of 10 minutes, and that each minutes of decrease in this average time would cost Rs. 10 per patient treated. How much would have to be budgeted by the clinic to decrease the average size of the queue from one and one-third patients to half patient.

12. Students arrive at the head office according to a Poisson input process with a mean rate of 40 per hour. The time required to serve a student has an exponential distribution with a mean of 50 per hour. Assume that the students are served by a single individual, find the average waiting time of a student.

13. Develop a program to Find Critical Path, completion time, float time for following activity table.

| Activity | Duration |
|---|---|
| 1-2 | 6 |
| 1-3 | 8 |
| 2-4 | 3 |
| 2-5 | 5 |
| 3-5 | 9 |
| 4-5 | 6 |
| 5-6 | 8 |

**Desirable:**

14. Develop a generalized **sequencing** program for n jobs and m machines. First develop a program for n jobs two machines, make a copy and then make it general for n jobs m machines. Show the sequence after every iteration, final sequence, total elapsed time and idle times for every machine

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Web Data Management (WDM)**
**Subject Code: 4659311**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To understand data models, syntax, query languages, schemas, query analysis, type-checking, publishing, indexes, and storage methods for semi structured data represented in XML.

## 2. Prerequisites:

- Database Programming, Database Management Systems

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Introduction: Modelling Web Data**<br>Database Technology and Web Applications, Semi structured data, Web Data Management with XML, XML and syntax, XML Data Model, XLink and XPointer | 20 % |
| **Unit II** | **XPath and XQuery**<br>Regular Path Expressions, XPath Basics, XPath Steps and expressions, path evaluations, axes, node tests, predicates, XQuery Syntax, Flwor, expression, advanced features, Xupdate | 20 % |
| **Unit III** | **Typing**<br>Automata on ranked trees, unranked trees, XML schema, other schema languages, Graph Semi structured data, graph bisimulation, data guides, XML query evaluation, XML identifiers, XML evaluation technique. | 20 % |
| **Unit IV** | **Ontologies, Querying and Data Integration**<br>RDF, RDF Schema, OWL, Description Logic, Querying data through ontologies, Querying RDF data, querying through RDFS, Answering queries though DL, Global-as-view (GAV) and Local-as-view (LAV) mediation, Ontology based mediation, Peer-to-peer data management Systems | 20 % |
| **Unit V** | **Building Web scale applications**<br>Web search, web crawlers, web information retrieval, Web graph mining and current topics in web search, Distributed systems, failure management, Required properties of a distributed system, P2P networks,Hash-based structures, distributed indexing, Distributed computing with MapReduce | 20 % |

## 4. Text Book:

**1)** S. Abiteboul, I. Manolescu, P. Rigaux, M. Rousset and P. Senellart, Web Data Management, Cambridge University Press, 2012

## 5. Reference Books:

**1)** S. Abiteboul, P. Buneman and D. Suciu, Data on the Web: From Relational to Semistructured Data to XML, Morgan Kaufman Publisher
**2)** Research papers and W3C web site

**6. Accomplishment of the student after completing the course**:

**1)** Have thorough understanding about state of the art in designing and building web applications and services, largely focusing on issues and challenges that revolve around the management and processing of web data.

**2)** Students will be able to use (i) current web technologies to develop dynamic web sites (ii) develop web sites that use dynamic content generated from a database

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 5th Semester**
**Subject Name: Wireless Communication & Mobile Computing (WCMC)**
**Subject Code: 4659312**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To acquire conceptual knowledge of Mobile Computing, OS and its Architecture.
- To familiarize with the RFID, GSM, GPRS Technology.
- To familiarize with the WAP Protocol.

## 2. Prerequisites:

- Overview of Basic Networking, its protocol suite, Data Communications, Overview of Database and Distributed Systems.

## 3. Course Contents:

| Unit | Course Content | Weightage Percentage |
|---|---|---|
| **Unit I** | **Mobile Computing Introduction**<br>History of Wireless Communications, Types, propagation modes Wireless network architecture, Applications, Security, Concerns and Standards, Benefits, Future. Evolution of mobile computing, Needs of mobile users, SOC and AOC client, Mobile computing OS, Architecture for mobile computing, Three tier architecture, design considerations for mobile computing, mobile computing Through internet, making existing applications Mobile-Enabled. | 20 % |
| **Unit II** | **Mobile Technologies**<br>Bluetooth, Radio frequency identification (RFID),Wireless Broadband, MobileIP: Introduction, Advertisement, Registration, TCP connections, two level addressing, abstract mobility management model, performance issue, routing in mobile host, Adhoc networks, Mobile transport layer: Indirect TCP, Snooping TCP, Mobile TCP, Time out freezing, Selective retransmission, transaction Oriented TCPIpv6, Global system for mobile communication, GSM architecture, GSM entities, call routing in GSM,PLMN interface, GSM addresses and identifiers, network aspects in GSM, GSM frequency allocation, authentication and security, Short message services, Mobile computing over SMS, SMS, value added services accessing the SMS bearer. | 30% |
| **Unit III** | **General packet radio service(GPRS):**<br>GPRS and packet data network, GPRS network architecture, GPRS network operation, data services in GPRS, Applications of GPRS, Billing and charging in GPRS. | 20 % |
| **Unit IV** | **Wireless Application Protocol(WAP):**<br>Introduction to WebSocket, WebSocket API, WAP, MMS, GPRS application CDMA and 3G Spread-spectrum Technology, CDMA versus GSM, Wireless data, third generation networks, applications in 3G Wireless LAN, Wireless LAN advantages, IEEE802.11 standard Wireless LAN architecture, Mobility in Wireless LAN, Deploying Wireless LAN, Mobile adhoc networks and sensor networks, wireless LAN security, W iFi v/s 3G Voice over Internet protocol and convergence, Voice over IP,H.323 framework for | 30% |

| | | |
|---|---|---|
| | voice over IP,SIP, comparison between H.323 ad SIP, Real time protocols, convergence technologies, call routing, call routing, voice over IP applications, IMS, Mobile VoIP, Security issues in mobile Information security, security techniques and algorithms, security framework for mobile environment. | |

**4. Text Book:**

**1)** Mobile Computing , Asoke K Telukder, Roopa R Yavagal, TMH
**2)** The complete reference J2ME, TMH
**3)** Programming for Mobile and Remote Computers, G. T. Thampi, dreamtech
**4)** Handbook of Wireless Networks and Mobile Computing, Ivan Stojmenovic ,Wiley

**5. Reference Books:**

**1)** Principles of Mobile Computing, - Hansmann, Merk, Nicklous and Stober, Springer
**2)** Mobile Communications, Jochen Schiller, Pearson
**3)** Mobile Computing, Raj Kamal, Oxford
**4)** Mobile Computing, Wandra & Wandra, Akshat Pub.
**5)** Android Wireless Application Development, Shane Conder, Lauren Darcey, Pearson
**6)** Professional Android 2 Application development, Reto Meier, Wrox, Wiley India

**6. Accomplishment of the student after completing the course**:

**1)** Have thorough understanding about state of the art in Mobile Technologies, General packet radio service and Wireless Application Protocol.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Syllabus for Master of Computer Applications, 6th Semester**
**Subject Name: Software Project - IV**
**Subject Code: 4669301**

**With effective from academic year 2018-19**

## 1. Learning Objectives:

- To solve industrial (or society or research) problems.
- To plan, schedule, and monitor the software project
- Development, coding, and testing of a large project cohesively.
- Documentation of project

## 2. Prerequisites:

Software Engineering, Programming / Coding language, RDBMS

## 3. Guidelines for Project

- It is recommended that the team should be about 2-3 students.
- The project should be free from plagiarism of any kind.
- Internal guides (i.e. The regular faculty members) must be allocated to projects.
- Project plan along with the division of work amongst teammates would have been prepared and got approved within a maximum of 5 days of the start of the project.
- Coding standards should be followed meticulously. At the minimum, the code should be self documented, modular, and should use the meaningful naming convention.
- It is advisable that object-oriented methodology is used with the reusability of classes and code, etc..
- The output reports must include MIS reports, if applicable.
- The documentation should include a chapter on "Learning during Project Work", i.e. "Experience of Journey during Project Duration".
- It is strongly recommended that Data structure/Database design is included in the report. At least portions of code (preferably full code) are mandatory. The student may be asked to write the code related to the project during the examination.
- If a student is compelled to follow certain instructions (by the external, i.e. organization's Guide) which he/she does not agree to, such a student must prepare a supplementary report to document his/her version and present it to the examiners if such a need arises.
- Internal guides (i.e. The regular faculty members) must devote the time allocated as per the timetable to guide the students for the project. The time allocation will be in accordance with the scheme for the 6th semester project as given.
- Internal guides should preferably visit external guide to track the project
- Project document should be printed on both sides of paper.

## 4. Accomplishments of the student after completing the course:

- Doing the project will enable the student to go through rich experience in developing large projects. Such an experience will include encountering various technical issues, finding sources to resolve the issues and finally finding the solution of all these issues satisfactorily.
- Thinking analytically, analyzing, and synthesizing requirements and complicated information for getting a good comprehension of the solution methodology to be adopted.
- Ability to document and write well.
- Organizing the time effectively.
- Working with teammates and generating substantial output of the efforts.
- It will prepare the students for analyzing and programming for industrial problem and large projects work in future

## 5. Documentation:

- The project has to be well-documented in the form of a Project Report (at least 50 pages comprising of the design, data dictionary, source code, screenshots, etc.).
- Format: Print out on both the side of page with single line spacing. Use Times New Roman of size 10 for normal text.

**TABLE OF CONTENTS**

1. Introduction
    - 1.1. Existing System
    - 1.2. Need for the New System
    - 1.3. Objective of the New System
    - 1.4. Problem Definition
    - 1.5. Core Components
    - 1.6. Project Profile
    - 1.7. Assumptions and Constraints
    - 1.8. Advantages and Limitations of the Proposed System
2. Requirement Determination & Analysis
    - 2.1. Requirement Determination
    - 2.2. Targeted Users
3. System Design
    - 3.1. Use Case Diagram
    - 3.2. Class Diagram
    - 3.3. Interaction Diagram
    - 3.4. Activity Diagram
    - 3.5. Data Dictionary
4. Development
    - 4.1. Coding Standards
    - 4.2. Screen Shots
5. Agile Documentation
    - 5.1 Agile Project Charter
    - 5.2 Agile Roadmap / Schedule
    - 5.3 Agile Project Plan
    - 5.4 Agile User Story (Minimum 3 Tasks)
    - 5.5 Agile Release Plan
    - 5.6 Agile Sprint Backlog
    - 5.7 Agile Test Plan
    - 5.8 Earned-value and burn charts
6. Proposed Enhancements
7. Conclusion
8. Bibliography

## 6. Evaluation Parameters :

- Evaluation of the projects would be done considering the framework available at the Institute. The main parameter of assessment would be the ability of the students to code.
- Though the project and domain specific knowledge would be assessed for, the evaluation would predominantly depend on the students' ability to explain, modify or revise of code.
- Coding standards should have been implemented.

- Though the project would be evaluated for the entire team, the examiner should emphasize on the contribution of each team member in the project development
- Total Marks ( 700 = 500 External + 200 Internal )

| | |
|---|---|
| Documentation (more specifically, Correctness and completeness of UML diagrams, and relationship between Class diagram & Database structure) | 150 |
| Explanation of Analysis & Design | 150 |
| Explanation of Code (To test the ability to explain how to code few functionalities used in the project) | 200 |
| Presentation | 100 |