ENGINEERING MATHEMATICS-IV
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17MAT41	IA Marks	40
Number of Lecture Hours/Week	04	Exam Marks	60
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Module 1	**Teaching Hours**
Numerical Methods: Numerical solution of ordinary differential equations of first order and first degree, Taylor's series method, modified Euler's method. Runge - Kutta method of fourth order, Milne's and Adams-Bashforth predictor and corrector methods (No derivations of formulae-single step computation only).	**10 Hours**
Module 2	
Numerical Methods: Numerical solution of second order ordinary differential equations, Runge-Kutta method and Milne's method. (No derivations of formulae-single step computation only). **Special Functions:** Series solution of Bessel's differential equation leading to $J_n(x)$-Bessel's function of first kind. Basic properties and orthogonality. Series solution of Legendre's differential equation leading to $P_n(x)$-Legendre polynomials. Rodrigue's formula, problems	**10 Hours**
Module 3	
Complex Variables: Review of a function of a complex variable, limits, continuity, differentiability. Analytic functions-Cauchy-Riemann equations in cartesian and polar forms. Properties and construction of analytic functions. Complex line integrals-Cauchy's theorem and Cauchy's integral formula, Residue, poles, Cauchy's Residue theorem (without proof) and problems. **Transformations:** Conformal transformations-Discussion of transformations: $w = z^2$, $w = e^z$, $w = z + (1/z)$ $(z \neq 0)$, Bilinear transformations-problems.	**10 Hours**
Module 4	
Probability Distributions: Random variables (discrete and continuous), probability functions. Poisson distributions**,** geometric distribution, uniform distribution, exponential and normal distributions, Problems. **Joint probability distribution:** Joint Probability distribution for two variables, expectation, covariance, correlation coefficient.	**10 Hours**
Module 5	
Sampling Theory: Sampling, Sampling distributions, standard error, test of hypothesis for means and proportions, confidence limits for means, student's t-distribution, Chi-square distribution as a test of goodness of fit. **Stochastic process:** Stochastic process, probability vector, stochastic matrices, fixed points, regular stochastic matrices, Markov chains, higher transition probability.	**10 Hours**

Course Outcomes: After studying this course, students will be able to:

- Solve first and second order ordinary differential equation arising in flow problems using single step and multistep numerical methods.
- Illustrate problems of potential theory, quantum mechanics and heat conduction by employing notions and properties of Bessel's functions and Legendre's polynomials.

- Explain the concepts of analytic functions, residues, poles of complex potentials and describe conformal and Bilinear transformation arising in field theory and signal processing.
- Develop probability distribution of discrete, continuous random variables and joint probability distribution occurring in digital signal processing, information theory and design engineering.
- Demonstrate testing of hypothesis of sampling distributions and illustrate examples of Markov chains related to discrete parameter stochastic process.

Question paper pattern:

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. B.V.Ramana "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.
2. B. S. Grewal," Higher Engineering Mathematics", Khanna publishers, 42nd edition, 2013.

Reference Books:

1. N P Bali and Manish Goyal, "A text book of Engineering mathematics" , Laxmi publications, latest edition.
2. Kreyszig, "Advanced Engineering Mathematics " - 9th edition, Wiley, 2013.
3. H. K Dass and Er. RajnishVerma, "Higher Engineering Mathematics", S. Chand, 1st ed, 2011.

OBJECT ORIENTED CONCEPTS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CS42	IA Marks	40
Number of Lecture Hours/Week	03	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Module 1	**Teaching Hours**
Introduction to Object Oriented Concepts: A Review of structures, Procedure–Oriented Programming system, Object Oriented Programming System, Comparison of Object Oriented Language with C, Console I/O, variables and reference variables, Function Prototyping, Function Overloading. **Class and Objects:** Introduction, member functions and data, objects and functions, objects and arrays, Namespaces, Nested classes, Constructors, Destructors. **Text book 1: Ch 1: 1.1 to 1.9 Ch 2: 2.1 to 2.6 Ch 4: 4.1 to 4.2**	**08 Hours**
Module 2	
Introduction to Java: Java's magic: the Byte code; Java Development Kit (JDK); the Java Buzzwords, Object-oriented programming; Simple Java programs. Data types, variables and arrays, Operators, Control Statements. **Text book 2: Ch:1 Ch: 2 Ch:3 Ch:4 Ch:5**	**08 Hours**
Module 3	
Classes, Inheritance, Exceptions, Packages and Interfaces: Classes: Classes fundamentals; Declaring objects; Constructors, this keyword, garbage collection. **Inheritance:** inheritance basics, using super, creating multi level hierarchy, method overriding. **Exception handling:** Exception handling in Java. Packages, Access Protection, Importing Packages, Interfaces. **Text book 2: Ch:6 Ch: 8 Ch:9 Ch:10**	**08 Hours**
Module 4	
Multi Threaded Programming, Event Handling: Multi Threaded Programming: What are threads? How to make the classes threadable ; Extending threads; Implementing runnable; Synchronization; Changing state of the thread; Bounded buffer problems, read-write problem, producer consumer problems. **Event Handling:** Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes. **Text book 2: Ch 11: Ch: 22**	**08 Hours**
Module 5	
The Applet Class: Introduction, Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Requesting repainting; Using the Status Window; The HTML APPLET tag; Passing parameters to Applets; getDocumentbase() and getCodebase(); ApletContext and showDocument(); The AudioClip Interface; The AppletStub Interface;Output to the Console. **Swings:** Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing Applet; Jlabel and ImageIcon; JTextField;The Swing Buttons;	**08 Hours**

JTabbedpane; JScrollPane; JList; JComboBox; JTable. **Text book 2: Ch 21: Ch: 29 Ch: 30**	

Course Outcomes: After studying this course, students will be able to

- Explain the object-oriented concepts and JAVA.
- Develop computer programs to solve real world problems in Java.
- Develop simple GUI int erfaces for a computer program to interact with users, and to **comprehend** the event-based GUI handling principles using Applets and swings.

Question paper pattern:

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Sourav Sahay, Object Oriented Programming with C++ , 2nd Ed, Oxford University Press,2006
   (Chapters 1, 2, 4)
2. Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007.
   (Chapters 1, 2, 3, 4, 5, 6, 8, 9,10, 11, 21, 22, 29, 30)

Reference Book:

1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806
2. Herbert Schildt, The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2003.
3. Stanley B.Lippmann, Josee Lajore, C++ Primer, 4th Edition, Pearson Education, 2005.
4. Rajkumar Buyya,S Thamarasi selvi, xingchen chu, Object oriented Programming with java,
   Tata McGraw Hill education private limited.
5. Richard A Johnson, Introduction to Java Programming and OOAD, CENGAGE Learning.
6. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies.

Note: Every institute shall organize a bridge organize on C++ either in the vacation or in the beginning of even semester.

DESIGN AND ANALYSIS OF ALGORITHMS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CS43	IA Marks	40
Number of Lecture Hours/Week	04	Exam Marks	60
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Module 1	**Teaching Hours**
Introduction: What is an Algorithm? **(T2:1.1),** Algorithm Specification **(T2:1.2),** Analysis Framework **(T1:2.1), Performance Analysis**: Space complexity, Time complexity **(T2:1.3)**. **Asymptotic Notations:** Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ), and Little-oh notation (o), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples **(T1:2.2, 2.3, 2.4). Important Problem Types:** Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. **Fundamental Data Structures:** Stacks, Queues, Graphs, Trees, Sets and Dictionaries. **(T1:1.3,1.4)**	**10 Hours**
Module 2	
Divide and Conquer: General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum **(T2:3.1, 3.3, 3.4)**, Merge sort, Quick sort **(T1:4.1, 4.2)**, Strassen's matrix multiplication **(T2:3.8)**, Advantages and Disadvantages of divide and conquer. **Decrease and Conquer Approach:** Topological Sort. **(T1:5.3)**	**10 Hours**
Module 3	
Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines **(T2:4.1, 4.3, 4.5). Minimum cost spanning trees:** Prim's Algorithm, Kruskal's Algorithm **(T1:9.1, 9.2)**. **Single source shortest paths:** Dijkstra's Algorithm **(T1:9.3)**. **Optimal Tree problem:** Huffman Trees and Codes **(T1:9.4)**. **Transform and Conquer Approach:** Heaps and Heap Sort **(T1:6.4)**.	**10 Hours**
Module 4	
Dynamic Programming: General method with Examples, Multistage Graphs **(T2:5.1, 5.2)**. **Transitive Closure:** Warshall's Algorithm, **All Pairs Shortest Paths:** Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem **((T1:8.2, 8.3, 8.4)**, Bellman-Ford Algorithm **(T2:5.4)**, Travelling Sales Person problem **(T2:5.9)**, Reliability design **(T2:5.8)**.	**10 Hours**
Module 5	
Backtracking: General method **(T2:7.1)**, N-Queens problem **(T1:12.1)**, Sum of subsets problem **(T1:12.1)**, Graph coloring **(T2:7.4)**, Hamiltonian cycles **(T2:7.5)**. **Branch and Bound:** Assignment Problem, Travelling Sales Person problem **(T1:12.2)**, **0/1 Knapsack problem (T2:8.2, T1:12.2):** LC Branch and Bound solution **(T2:8.2)**, FIFO Branch and Bound solution **(T2:8.2)**. **NP-**	**10 Hours**

Complete and NP-Hard problems: Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes **(T2:11.1)**.	

Course Outcomes: After studying this course, students will be able to

- Describe computational solution to well known problems like searching, sorting etc.
- Estimate the computational complexity of different algorithms.
- Develop an algorithm using appropriate design strategies for problem solving.

Question paper pattern:

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

T1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.

T2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI
2. Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education)

MICROPROCESSORS AND MICROCONTROLLERS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CS44	IA Marks	40
Number of Lecture Hours/Week	04	Exam Marks	60
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Module 1	**Teaching Hours**
The x86 microprocessor: Brief history of the x86 family, Inside the 8088/86, Introduction to assembly programming, Introduction to Program Segments, The Stack, Flag register, x86 Addressing Modes. **Assembly language programming:** Directives & a Sample Program, Assemble, Link & Run a program, More Sample programs, Control Transfer Instructions, Data Types and Data Definition, Full Segment Definition, Flowcharts and Pseudo code. **Text book 1: Ch 1: 1.1 to 1.7, Ch 2: 2.1 to 2.7**	**10 Hours**
Module 2	
x86: Instructions sets description**, Arithmetic and logic instructions and programs:** Unsigned Addition and Subtraction, Unsigned Multiplication and Division, Logic Instructions, BCD and ASCII conversion, Rotate Instructions. **INT 21H and INT 10H Programming :** Bios INT 10H Programming , DOS Interrupt 21H. 8088/86 Interrupts, x86 PC and Interrupt Assignment. **Text book 1: Ch 3: 3.1 to 3.5, Ch 4: 4.1 , 4.2 Chapter 14: 14.1 and 14.2**	**10 Hours**
Module 3	
Signed Numbers and Strings: Signed number Arithmetic Operations, String operations. **Memory and Memory interfacing: M**emory address decoding, data integrity in RAM and ROM, 16-bit memory interfacing. **8255 I/O programming:** I/O addresses MAP of x86 PC's, programming and interfacing the 8255. **Text book 1: Ch 6: 6.1, 6.2. Ch 10: 10.2, 10.4, 10.5. Ch 11: 11.1 to 11.4**	**10 Hours**
Module 4	
Microprocessors versus Microcontrollers, **ARM Embedded Systems :**The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software, **ARM Processor Fundamentals :** Registers , Current Program Status Register , Pipeline, Exceptions, Interrupts, and the Vector Table , Core Extensions **Text book 2:Ch 1:1.1 to 1.4, Ch 2:2.1 to 2.5**	**10 Hours**
Module 5	
Introduction to the ARM Instruction Set : Data Processing Instructions , Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants, Simple programming exercises. **Text book 2: Ch 3:3.1 to 3.6 (Excluding 3.5.2)**	**10 Hours**

Course Outcomes: After studying this course, students will be able to

- Differentiate between microprocessors and microcontrollers
- Develop assembly language code to solve problems
- Explain interfacing of various devices to x86 family and ARM processor

- Demonstrate interrupt routines for interfacing devices

Question paper pattern:

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey, The x86 PC Assembly Language Design and Interfacing, 5th Edition, Pearson, 2013.
2. **ARM system developers guide**, Andrew N Sloss, Dominic Symes and Chris Wright, Elsevier,Morgan Kaufman publishers, 2008.

Reference Books:

1. Douglas V. Hall: Microprocessors and Interfacing, Revised 2nd Edition, TMH, 2006.
2. K. Udaya Kumar & B.S. Umashankar : Advanced Microprocessors & IBM-PC Assembly Language Programming, TMH 2003.
3. Ayala : The 8086 Microprocessor: programming and interfacing - 1st edition, Cengage Learning
4. The Definitive Guide to the ARM Cortex-M3, by Joseph Yiu, 2nd Edition , Newnes, 2009
5. The Insider's Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition, 2005
6. ARM System-on-Chip Architecture, Steve Furber, Second Edition, Pearson, 2015
7. Architecture, Programming and Interfacing of Low power Processors- ARM7, Cortex-M and MSP430, Lyla B Das Cengage Learning, 1st Edition

SOFTWARE ENGINEERING
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CS45	IA Marks	40
Number of Lecture Hours/Week	04	Exam Marks	60
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Module 1	**Teaching Hours**
Introduction: Software Crisis, Need for Software Engineering. Professional Software Development, Software Engineering Ethics. Case Studies. **Software Processes**: Models: Waterfall Model **(Sec 2.1.1)**, Incremental Model **(Sec 2.1.2)** and Spiral Model **(Sec 2.1.3)**. Process activities. **Requirements Engineering**: Requirements Engineering Processes **(Chap 4)**. Requirements Elicitation and Analysis **(Sec 4.5).** Functional and non-functional requirements **(Sec 4.1)**. The software Requirements Document **(Sec 4.2)**. Requirements Specification **(Sec 4.3)**. Requirements validation **(Sec 4.6)**. Requirements Management **(Sec 4.7)**.	**12 Hours**
Module 2	
System Models: Context models **(Sec 5.1)**. Interaction models **(Sec 5.2)**. Structural models **(Sec 5.3)**. Behavioral models **(Sec 5.4)**. Model-driven engineering **(Sec 5.5)**. **Design and Implementation**: Introduction to RUP **(Sec 2.4)**, Design Principles **(Chap 17)**. Object-oriented design using the UML **(Sec 7.1)**. Design patterns **(Sec 7.2)**. Implementation issues **(Sec 7.3)**. Open source development **(Sec 7.4)**.	**11 Hours**
Module 3	
Software Testing: Development testing **(Sec 8.1)**, Test-driven development **(Sec 8.2)**, Release testing **(Sec 8.3)**, User testing **(Sec 8.4)**. Test Automation **(Page no 42, 70,212, 231,444,695)**. **Software Evolution**: Evolution processes **(Sec 9.1)**. Program evolution dynamics **(Sec 9.2)**. Software maintenance **(Sec 9.3)**. Legacy system management **(Sec 9.4)**.	**9 Hours**
Module 4	
Project Planning: Software pricing **(Sec 23.1)**. Plan-driven development **(Sec 23.2)**. Project scheduling **(Sec 23.3)**: Estimation techniques **(Sec 23.5)**. **Quality management**: Software quality **(Sec 24.1)**. Reviews and inspections **(Sec 24.3)**. Software measurement and metrics **(Sec 24.4)**. Software standards **(Sec 24.2)**	**10 Hours**
Module 5	
Agile Software Development: Coping with Change **(Sec 2.3)**, The Agile Manifesto: Values and Principles. Agile methods: SCRUM (Ref "**The SCRUM Primer, Ver 2.0**") and Extreme Programming **(Sec 3.3)**. Plan-driven and agile development **(Sec 3.2)**. Agile project management **(Sec 3.4),** Scaling agile methods **(Sec 3.5)**:	**8 Hours**

Course Outcomes: After studying this course, students will be able to:

- Design a software system, component, or process to meet desired needs within realistic constraints.
- Assess professional and ethical responsibility

- Function on multi-disciplinary teams
- Make use of techniques, skills, and modern engineering tools necessary for engineering practice
- Comprehend software systems or parts of software systems.

Question paper pattern:

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012.

(Listed topics only from Chapters 1,2,3,4, 5, 7, 8, 9, 23, and 24)

2. The SCRUM Primer, Ver 2.0, http://www.goodagile.com/scrumprimer/scrumprimer20.pdf

Reference Books:

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill**.**
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India

Web Reference for eBooks on Agile:

1. http://agilemanifesto.org/
2. http://www.jamesshore.com/Agile-Book/

DATA COMMUNICATION
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CS46	IA Marks	40
Number of Lecture Hours/Week	04	Exam Marks	60
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Contents	**Teaching Hours**
Module 1	
Introduction: Data Communications, Networks, Network Types, Internet History, Standards and Administration, **Networks Models**: Protocol Layering, TCP/IP Protocol suite, The OSI model, **Introduction to Physical Layer-1:** Data and Signals**,** Digital Signals, Transmission Impairment, Data Rate limits, Performance, **Digital Transmission**: Digital to digital conversion (Only Line coding: Polar, Bipolar and Manchester coding).	**10 Hours**
Module 2	
Physical Layer-2: Analog to digital conversion (only PCM), Transmission Modes, **Analog Transmission**: Digital to analog conversion, **Bandwidth Utilization**: Multiplexing and Spread Spectrum, **Switching**: Introduction, Circuit Switched Networks and Packet switching.	**10 Hours**
Module 3	
Error Detection and Correction: Introduction, Block coding, Cyclic codes, Checksum, Forward error correction**, Data link control**: DLC services, Data link layer protocols, HDLC, and Point to Point protocol (Framing, Transition phases only).	**10 Hours**
Module 4	
Media Access control: Random Access, Controlled Access and Channelization, **Wired LANs Ethernet**: Ethernet Protocol, Standard Ethernet, Fast Ethernet, Gigabit Ethernet and 10 Gigabit Ethernet, **Wireless LANs:** Introduction, IEEE 802.11 Project and Bluetooth.	**10 Hours**
Module 5	
Other wireless Networks: WIMAX, Cellular Telephony, Satellite networks, **Network layer Protocols** : Internet Protocol, ICMPv4,Mobile IP, **Next generation IP**: IPv6 addressing, The IPv6 Protocol, The ICMPv6 Protocol and Transition from IPv4 to IPv6.	**10 Hours**

Course Outcomes: After studying this course, students will be able to

- Illustrate basic computer network technology.
- Identify the different types of network topologies and protocols.
- List and explain the layers of the OSI model and TCP/IP model.
- Comprehend the different types of network devices and their functions within a network
- Demonstrate subnetting and routing mechanisms.

Question paper pattern:

The question paper will have ten questions.

There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Book:

Behrouz A. Forouzan, Data Communications and Networking 5E, 5th Edition, Tata McGraw-Hill, 2013. (Chapters 1.1 to 1.5, 2.1 to 2.3, 3.1, 3.3 to 3.6, 4.1 to 4.3, 5.1, 6.1, 6.2, 8.1 to 8.3, 10.1 to 10.5, 11.1 to 11.4, 12.1 to 12.3, 13.1 to 13.5, 15.1 to 15.3, 16.1 to 16.3, 19.1 to 19.3, 22.1 to 22.4)

Reference Books:

1. Alberto Leon-Garcia and Indra Widjaja: Communication Networks - Fundamental Concepts and Key architectures, 2nd Edition Tata McGraw-Hill, 2004.
2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
4. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007

DESIGN AND ANALYSIS OF ALGORITHM LABORATORY
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CSL47	IA Marks	40
Number of Lecture Hours/Week	01 I + 02 P	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 02

Description

Design, develop, and implement the specified algorithms for the following problems using Java language under LINUX /Windows environment.Netbeans/Eclipse IDE tool can be used for development and demonstration.

Experiments

1	A	Create a Java class called ***Student***with the following details as variables within it. (i) USN (ii) Name (iii) Branch (iv) Phone Write a Java program to create *nStudent* objects and print the USN, Name, Branch, and Phoneof these objects with suitable headings.
	B	Write a Java program to implement the Stack using arrays. Write Push(), Pop(), and Display() methods to demonstrate its working.
2	A	Design a superclass called ***Staff*** with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely ***Teaching*** (domain, publications), ***Technical*** (skills), and ***Contract*** (period). Write a Java program to read and display at least 3 *staff* objects of all three categories.
	B	Write a Java class called ***Customer*** to store their name and date_of_birth. The date_of_birth format should be dd/mm/yyyy. Write methods to read customer data as <name, dd/mm/yyyy> and display as <name, dd, mm, yyyy> using StringTokenizer class considering the delimiter character as "/".
3	A	Write a Java program to read two integers *a* and*b*. Compute *a*/*b* and print, when *b* is not zero. Raise an exception when *b* is equal to zero.
	B	Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number andprints; third thread will print the value of cube of the number.
4		Sort a given set of ***n*** integer elements using **Quick Sort** method and compute its time complexity. Run the program for varied values of *n*> 5000 and record the time taken to sort. Plot a graph of the time taken versus ***n***on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using

	Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.
5	Sort a given set of ***n*** integer elements using **Merge Sort** method and compute its time complexity. Run the program for varied values of *n*> 5000, and record the time taken to sort. Plot a graph of the time taken versus ***n***on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.
6	Implement in Java, the **0/1 Knapsack** problem using (a) Dynamic Programming method (b) Greedy method.
7	From a given vertex in a weighted connected graph, find shortest paths to other vertices using **Dijkstra's algorithm**. Write the program in Java.
8	Find Minimum Cost Spanning Tree of a given connected undirected graph using **Kruskal'salgorithm.** Use Union-Find algorithms in your program.
9	Find Minimum Cost Spanning Tree of a given connected undirected graph using **Prim's algorithm**.
10	Write Java programs to (a) Implement All-Pairs Shortest Paths problem using **Floyd's algorithm**. (b) Implement **Travelling Sales Person problem** using Dynamic programming.
11	Design and implement in Java to find a **subset** of a given set **S** = {Sl, S2,.....,Sn} of ***n*** positive integers whose SUM is equal to a given positive integer ***d***. For example, if S ={1, 2, 5, 6, 8} and ***d***= 9, there are two solutions {1,2,6}and {1,8}. Display a suitable message, if the given problem instance doesn't have a solution.
12	Design and implement in Java to find all **Hamiltonian Cycles** in a connected undirected Graph G of *n* vertices using backtracking principle.

Course Outcomes: The students should be able to:

- Design algorithms using appropriate design techniques (brute-force, greedy, dynamic programming, etc.)
- Develop variety of algorithms such as sorting, graph related, combinatorial, etc., in a high level language.
- Analyze and compare the performance of algorithms using language features.
- Apply and implement learned algorithm design techniques and data structuresto solve real-world problems.

Conduction of Practical Examination:

All laboratory experiments (Twelve problems) are to be included for practical examination. Students are allowed to pick one experiment from the lot.
To generate the data set use random number generator function.
Strictly follow the instructions as printed on the cover page of answer script for

breakup of marks
Marks distribution: Procedure + Conduction + Viva: 15 + 70 + 15 (100).
Change of experiment is allowed only once and marks allotted to the procedure

MICROPROCESSOR AND MICROCONTROLLER LABORATORY
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	17CSL48	IA Marks	40
Number of Lecture Hours/Week	01 I + 02 P	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 02

Description

Demonstration and Explanation hardware components and Faculty in-charge should explain 8086 architecture, pin diagram in one slot. The second slot, the Faculty in-charge should explain instruction set types/category etc. Students have to prepare a write-up on the same and include it in the Lab record and to be evaluated.

Laboratory Session-1: Write-up on Microprocessors, 8086 Functional block diagram, Pin diagram and description. The same information is also taught in theory class; this helps the students to understand better.

Laboratory Session-2: Write-up on Instruction group, Timing diagrams, etc. The same information is also taught in theory class; this helps the students to understand better.

Note: These TWO Laboratory sessions are used to fill the gap between theory classes and practical sessions. Both sessions are evaluated as lab experiments for 20 marks.

Experiments

- Develop and execute the following programs using 8086 Assembly Language. Any suitable assembler like MASM/TASM/8086 kit or any equivalent software may be used.
- Program should have suitable comments.
- The board layout and the circuit diagram of the interface are to be provided to the student during the examination.
- Software Required: Open source ARM Development platform, KEIL IDE and Proteus for simulation

SOFTWARE PROGRAMS: PART A

1. Design and develop an assembly language program to search a key element “X” in a list of ‘n’ 16-bit numbers. Adopt Binary search algorithm in your program for searching.
2. Design and develop an assembly program to sort a given set of ‘n’ 16-bit numbers in ascending order. Adopt Bubble Sort algorithm to sort given elements.
3. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.
4. Develop an assembly language program to compute nCr using recursive procedure. Assume that ‘n’ and ‘r’ are non-negative integers.
5. Design and develop an assembly language program to read the current time and Date

from the system and display it in the standard format on the screen.

6. To write and simulate ARM assembly language programs for data transfer, arithmetic and logical operations (Demonstrate with the help of a suitable program).
7. To write and simulate C Programs for ARM microprocessor using KEIL (Demonstrate with the help of a suitable program)

Note : To use KEIL one may refer the book: Insider's Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition, 2005

HARDWARE PROGRAMS: PART B

8. a. Design and develop an assembly program to demonstrate BCD Up-Down Counter (00-99) on the Logic Controller Interface.
   b. Design and develop an assembly program to read the status of two 8-bit inputs (X & Y) from the Logic Controller Interface and display X*Y.
9. Design and develop an assembly program to display messages "FIRE" and "HELP" alternately with flickering effects on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages (Examiner does not specify these delay values nor is it necessary for the student to compute these values).
10. Design and develop an assembly program to drive a Stepper Motor interface and rotate the motor in specified direction (clockwise or counter-clockwise) by N steps (Direction and N are specified by the examiner). Introduce suitable delay between successive steps. (Any arbitrary value for the delay may be assumed by the student).
11. Design and develop an assembly language program to
    a. Generate the Sine Wave using DAC interface (The output of the DAC is to be displayed on the CRO).
    b. Generate a Half Rectified Sine waveform using the DAC interface. (The output of the DAC is to be displayed on the CRO).
12. To interface LCD with ARM processor-- ARM7TDMI/LPC2148. Write and execute programs in C language for displaying text messages and numbers on LCD
13. To interface Stepper motor with ARM processor-- ARM7TDMI/LPC2148. Write a program to rotate stepper motor

Study Experiments:

1. Interfacing of temperature sensor with ARM freedom board (or any other ARM microprocessor board) and display temperature on LCD
2. To design ARM cortex based automatic number plate recognition system
3. To design ARM based power saving system

Course Outcomes: After studying this course, students will be able to

- Summarize 80x86 instruction sets and comprehend the knowledge of how assembly language works.
- Design and develop assembly programs using 80x86 assembly language instructions
- Infer functioning of hardware devices and interfacing them to x86 family
- Choose processors for various kinds of applications.

Conduction of Practical Examination:

- All laboratory experiments (all 7 + 6 nos) are to be included for practical examination.
- Students are allowed to pick one experiment from each of the lot.
- Strictly follow the instructions as printed on the cover page of answer script for breakup of marks
- PART –A: Procedure + Conduction + Viva: **08 + 35 +07 (50)**
- PART –B: Procedure + Conduction + Viva: **08 + 35 +07 (50)**
- Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.