|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code**  **20CST-313** | **Operating Systems** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type** |
|  |  | 3 | 0 | 0 | 0 | 3 | 3 | CR |
|  | |  | | | | Course Code(s)  **20CST-313** | | | |
| **PRE-REQUISITE** | | 20CST-252 | | | |  | | | |
| **CO-REQUISITE** | | 20CSP-314,20CST-315,20CST-316,20CST-319,20CSP-321 | | | |  | | | |
| **ANTI-REQUISITE** | | 20CST-463 | | | |  | | | |

**Course Objectives**

* To grasp a fundamental understanding of operating systems and its functionalities.
* To understand process management, various scheduling algorithms, concurrency and synchronization.
* To understand memory management and virtual memory concepts in modern Operating systems.

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | Explain the structure of the operation system and illustrates the process management mechanism and applications |
| CO2 | Identify and solve the problems of deadlocks |
| CO3 | Student will be able to outline the various features such as  memory management, device management and file management of operating system |
| CO4 | Categorize the security threats |
| CO5 | Apply the knowledge of distributed and Network Operating Systems on the complex problems related to computer network |

**Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **INTRODUCTION** | **Contact Hours:15** |
| **Introduction to the Operating System** | Introduction to Operating Systems, Operating System Structure, Main Functions and characteristics of Operating Systems, Types of Operating Systems, System calls, Types of system calls, System programs. | |
| **Process Management** | Process Concept, Process Control Block, Process Scheduling, Threads, CPU Scheduling : Preemptive/ Non Preemptive Scheduling, Scheduling Criteria, Scheduling Algorithms, inter-process communication, remote procedure calls, Process Synchronization | |
| **Deadlocks** | Deadlock characterization and conditions for deadlock, deadlock prevention, Deadlock avoidance-safe state, resource allocation graph algorithm, Banker’s algorithms-Safety algorithm, Deadlock detection, Recovery from deadlock. | |
| **Unit-2** | **MEMORY AND DEVICE MANAGEMENT** | **Contact Hours:15** |
| **Memory Management** | Address binding, logical versus physical address space, dynamic loading, Swapping, contiguous memory allocation, Fragmentation, Paging, Segmentation, Segmentation with Paging, Virtual Memory Concept, Demand Paging, Page Replacement, Page Replacement Algorithms | |
| **Device Management** | Disk Structure, Disk formatting, Disk Scheduling Algorithms, RAID structure-RAID levels, problems with RAID. | |
| **File Management** | File Concepts, Access Methods, Directory Structure, Allocation Methods, Free Space Management. | |
| **Unit-3** | **SECURITY AND TYPES OF OS** | **Contact Hours:15** |
| **System Protection and Security** | Goals, principles and domain of protection, Access matrix, implementation of access matrix, the security problem, program threats, system and network threats. | |
| **Distributed and Network Operating Systems** | Overview: Topology, connection strategy, network operating system types: Peer to Peer & Client server, Distributed message passing. | |

**Text Books:**

1. Galvin, Peter B., Silberchatz, A., “Operating System Concepts”, Addison Wesley, 8th Edition.
2. Flynn, “Operating Systems”, Cengage Learning.
3. Dhamdhere, D.M., "Operating System: A Concept Based Approach", Tata Mc-Graw-Hill.

**Reference Books:**

1. Madnick , Stuart E., Donovan, John J. “ *Operating System*”, McGrawHill.
2. Stalling, William, “*Operating Systems*”, Pearson Education, Fifth Edition.

**Mode of Evaluation: The performance of students is evaluated as follows:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | **Theory** | | |
| **Components** | | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** | |
| **Marks** | | 40 | 60 | |
| **Total Marks** | | 100 | | |
| **Mapping Between COs and Pos** | | | | | | |
| **SN** | | **Course Outcome (CO)** | | | **Mapped Programme Outcome (PO)** | |
| 1 | |  | | |  | |
| 2 | |  | | |  | |
| 3 | |  | | |  | |
| 4 | |  | | |  | |

**CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | 3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 | 2 | 3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 | 3 |  |  | 3 |  |  |  |  |  |  |  |  | 2 |  |
| CO4 |  | 8 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  | 3 |  |  |  |  |  |  |  |  |  |  | 3 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Subject Code  **20CST321** | **Project Based Learning in Java** | L | T | P | C |
| Total Contact Hours : 30Hours | 0 | 2 | 0 | 2 |
| Common to all Specializations of CSE 3rd Year |
| Prerequisite: Knowing Programming Language Java | | | | |

**Course Objectives**

1.To understand the use of Java in a variety of technologies and on different platforms.

2. Understand the server side programming

3.To understand the basic concepts and fundamentals of platform independent object oriented language.

4. To demonstrate skills in writing programs using exception handling techniques and multithreading.

5. To understand streams and efficient user interface design techniques.

### Course Outcomes

1. To gain knowledge of the structure and use the Java programming language for various technologies

2. Use the syntax and semantics of java programming language and basic concepts of OOP.

3. Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.

4 . Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes.

5. Design event driven GUI and web related applications which mimic the real word scenarios.

### Unit-I

Java Fundamentals: Introduction to Java. Difference between C++ and Java. Keywords, Tokens, Data types. Use of public, private and protected.

OOPS using Java: Use of class and method in Java. Inheritance, Abstraction, Polymorphism, Encapsulation and data privacy. Difference between method overloading and method overriding.

Exception Handing: Introduction to Exceptions. Difference between error and exception. Use of try, catch and throw. Difference between throw and throws. Types of Exceptions, Exception handling in Java.

### Unit-II

Collection Framework: Use of Collections in Java. ArrayList, LinkedList, HashMap, TreeMap, HashSet in Java.. Multithreading in Java. Thread Synchronization. Thread Priority, Thread LifeCycle.

Wrapper Classes, I/O Streams and Annotations: Use of wrapper classes in Java- Integer, Character, Long, Boolean. Autoboxing and Unboxing. Byte stream, Character stream, Object serialization, cloning. System defined annotations, Custom annotations, application of annotations, Testing using JUnit.

JDBC: Database connectivity, Types of Drivers for connection, Connection Example. CRUD operations using Database, Configuring various types of drivers for Java Database Connectivity, MVC Model for project development, Sequence, Dual table , Date type management in Java.

### Unit-III

Servlets and JSP: Servlet Lifecycle, Generic Servlet, Http Servlet, Linking Servlet to HTML, HttpServlet Request and Response, Servlet with JDBC, Configuring project using servlet, Servlet Config and Servlet Mapping JSP declaration, JSP directives, JSP Scriptlets, JSP include tag, JSP page tag, JSTL.

XML and Web Services: Structure of XML, Elements of XML 1.0, 2.0, DTDs, XML parser, DOM parser, Web services using REST and HTTP, Creating web services for database access via remote servers

### Text books:

1. Herbert Schildt, Java : The Complete Reference, 9th Edition, Oracle Press.
2. Gary Cornell, Core Java Volume II Advanced Features, 8th Edition, Pearson Education.
3. Jim Keogh, J2ee : Complete Reference, 1st Edition, Tata McGraw Hill.

### Reference books:

1. James Gosling, Ken Arnold and David Holmes, Java Programming Language, 5th Edition, Pearson Education.
2. Gary Cornell, Core Java Volume I, 3rd Edition, Pearson Education.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | To understand the use of Java in a variety of technologies and on different platforms. | To gain knowledge of the structure and use the Java programming language for various technologies |
| 2 | Understand the server side programming | Use the syntax and semantics of java programming language and basic concepts of OOP. |
| 3 | To understand the basic concepts and fundamentals of platform independent object oriented language. | Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages |
| 4 | To demonstrate skills in writing programs using exception handling techniques and multithreading. | Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes. |
| 5 | To understand streams and efficient user interface design techniques. | Design event driven GUI and web related applications which mimic the real word scenarios. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **COs** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| **CO1** | 3 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| **CO2** | 3 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| **CO3** | 3 | – | – | – | – | – | – | – | – | – | – | – | – | – |
| **CO4** | 3 | 2 | 3 | – | – | – | – | – | – | – | – | – | – | 2 |
| **CO5** | 3 | 3 | 3 | 2 | – | – | – | – | – | – | – | – | – | 2 |
| **CO6** | 3 | 3 | 3 | 3 | – | – | – | – | – | – | – | – | – | 2 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST321** | **Project Based learning in JAVA** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SubjectCode**  **20CSP321** | **Project Based Learning in Java Lab** | **L** | **T** | **P** | **C** |
| Total Contact Hours : 60 Hours | **0** | **0** | **4** | **2** |
| Common to all Specializations of CSE  3rd Year |
| Prerequisite: Studied Programming Language | | | | |

**Course Objectives**

1. To generate analytical and conceptual ability related to fundamentals of Java.
2. To understand the concepts of Web application development.
3. To understand the concepts of Fundamentals of I/O , Database Connectivity
4. To Implement of the OOPS concepts using Eclipse Environment
5. To implement the concepts of Collections and able to access database through

**Course Outcomes**

1.Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.

2.Read and make elementary modifications to Java programs that solve real-world problems.

3.Designs will demonstrate the use of good object-oriented design principles including encapsulation and information hiding.

4.The implementation will demonstrate the use of a variety of basic control structures including selection and repetition; classes and objects in a tiered architecture (user interface, controller, and application logic layers); primitive and reference data types including composition; basic AWT components; file-based I/O; and one-dimensional arrays.

5.Test plans will include test cases demonstrating testing strategies.

**List of Experiments**

**UNIT-I**

* 1. Create a application to save the employee information usingarrays.
  2. Design and implement a simple inventory control system for a small video rentalstore.
  3. Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.

### UNIT-II

* 1. Create a program to set view of Keys from Java Hashtable.
  2. Create a program to show the usage of Sets of Collectioninterface.
  3. Write a Program to perform the basic operations like insert, delete, display and search inlist. List contains String object items where these operations are to beperformed.
  4. Create a menu based Java application with the following options.1.Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in afile. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application shouldexit.

UNIT-III

* 1. Create a palindrome creator application for making a longest possible palindrome out of given input string.
  2. Create a Servlet/ application with a facility to print any message on web browser.
  3. Create JSP application for addition, multiplication and division.

1. **Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | To generate analytical and conceptual ability related to fundamentals of Java. | Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs. |
| 2 | To understand the concepts of Web application development. | Read and make elementary modifications to Java programs that solve real-world problems. |
| 3 | To understand the concepts of Fundamentals of I/O , Database Connectivity | Designs will demonstrate the use of good object-oriented design principles including encapsulation and information hiding. |
| 4 | To Implement of the OOPS concepts using Eclipse Environment | The implementation will demonstrate the use of a variety of basic control structures including selection and repetition; classes and objects in a tiered architecture (user interface, controller, and application logic layers); primitive and reference data types including composition; basic AWT components; file-based I/O; and one-dimensional arrays. |
| 5 | To implement the concepts of Collections and able to access database through | * Test plans will include test cases demonstrating Testing strategies. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **COs** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| **CO1** | 3 | – | – | – | 2 | – | – | – | – | – | – | – | – | – |
| **CO2** | 3 | – | – | – | 2 | – | – | – | – | – | – | – | – | – |
| **CO3** | 3 | – | – | – | 2 | – | – | – | – | – | – | – | – | – |
| **CO4** | 3 | 2 | 2 | – | 2 | – | – | – | – | – | – | – | 2 | 2 |
| **CO5** | 3 | 2 | – | – | 2 | – | – | – | – | – | – | – | – | – |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code**  **21CSP-314** | **Competitive Coding-I** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
|  |  | 0 | 0 | 2 | 0 | 1 | 2 | EE |
|  | |  | | | | Course Code(s)  **20CSP-314** | | | |
| **PRE-REQUISITE** | | 21CSP-259 | | | |  | | | |
| **CO-REQUISITE** | | 21CST-313 ,21CST-315,21CST-316,21CST-319,21CSP-321 | | | |  | | | |
| **ANTI-REQUISITE** | | 21CSP-356 | | | |  | | | |

**a. Course Description**

During the course the student will learn everything needed to participate in real competitions. Along the way the students also also gain useful skills for which competitive programmers are so highly valued by employers: ability to write efficient, reliable, and compact code, manage your time well when it’s limited, apply basic algorithmic ideas to real problems, etc.

**b. Course Objectives**

* To give students the ability to write reliable codes.
* To provide skills to the students to write compact and efficient code in a quick manner
* **To provide logic building capability to the student.**

**c. Course Outcomes**

|  |  |
| --- | --- |
| CO1 | To Provide real time problems to students for coding |
| CO2 | To enhance the logic building capability of students |
| CO3 | To give students a quick understanding of new codes and techniques to be used |
| CO4 | Apply the knowledge on a wider set of problems to achieve better results |
| CO5 | Apply the appropriate logic to find the solution of problem in efficient time. |

**d. Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **Data Structures** | **Contact Hours:15** |
| **Arrays, Stacks & Queues** | <https://www.hackerrank.com/challenges/arrays-ds/problem>  <https://www.hackerrank.com/challenges/2d-array/problem>  <https://www.hackerrank.com/challenges/dynamic-array/problem>  <https://www.hackerrank.com/challenges/array-left-rotation/problem>  <https://www.hackerearth.com/fr/practice/data-structures/arrays/multi-dimensional/practice-problems/algorithm/add-alternate-elements-of-2-dimensional-array/>  <https://www.hackerrank.com/challenges/equal-stacks/problem>  <https://www.hackerrank.com/challenges/maximum-element/problem>  <https://www.hackerearth.com/fr/practice/data-structures/queues/basics-of-queues/practice-problems/algorithm/disk-tower-b7cc7a50/>  <https://www.hackerrank.com/challenges/queue-using-two-stacks/problem> | |
| **Linked List** | <https://www.hackerrank.com/challenges/print-the-elements-of-a-linked-list/problem>  <https://www.hackerrank.com/challenges/insert-a-node-at-the-tail-of-a-linked-list/problem>  <https://www.hackerrank.com/challenges/compare-two-linked-lists/problem>  <https://www.hackerrank.com/challenges/reverse-a-linked-list/problem>  <https://www.hackerrank.com/challenges/insert-a-node-into-a-sorted-doubly-linked-list/problem>  <https://www.hackerrank.com/challenges/reverse-a-doubly-linked-list/problem> | |
| **Graphs** | <https://www.hackerrank.com/challenges/insertionsort1/problem>  <https://www.hackerearth.com/practice/algorithms/graphs/breadth-first-search/practice-problems/algorithm/monk-and-the-islands/>  <https://www.hackerearth.com/practice/algorithms/graphs/depth-first-search/practice-problems/algorithm/anupam-graph/>  <https://www.hackerrank.com/challenges/kruskalmstrsub/problem>  <https://www.hackerrank.com/challenges/primsmstsub/problem>  <https://www.hackerrank.com/challenges/dijkstrashortreach/problem>  <https://www.hackerrank.com/challenges/minimum-mst-graph/problem>  <https://www.hackerrank.com/challenges/clique/problem>  <https://www.hackerrank.com/challenges/crab-graphs/problem>  <https://www.hackerrank.com/challenges/the-quickest-way-up/problem> | |
| **Trees** | <https://www.hackerrank.com/challenges/tree-preorder-traversal/problem>  <https://www.hackerrank.com/challenges/tree-postorder-traversal/problem>  <https://www.hackerrank.com/challenges/tree-inorder-traversal/problem>  <https://www.hackerrank.com/challenges/tree-height-of-a-binary-tree/problem>  <https://www.hackerrank.com/challenges/tree-level-order-traversal/problem>  <https://www.hackerrank.com/challenges/binary-search-tree-insertion/problem>  <https://www.hackerrank.com/challenges/binary-search-tree-lowest-common-ancestor/problem>  <https://www.hackerrank.com/challenges/self-balancing-tree/problem>  <https://www.hackerrank.com/contests/hourrank-19/challenges/maximal-tree-diameter/problem>  <https://www.hackerrank.com/contests/101hack33/challenges/longest-path> | |
| **Unit-2** | **Algorithms** | **Contact Hours:15** |
| **String Algorithms** | <https://www.hackerearth.com/practice/algorithms/string-algorithm/string-searching/practice-problems/algorithm/string-4-d1093b86/>  <https://www.hackerearth.com/practice/algorithms/string-algorithm/string-searching/practice-problems/algorithm/palindrome-string-22/>  <https://www.hackerearth.com/practice/algorithms/string-algorithm/basics-of-string-manipulation/practice-problems/algorithm/palindrome-124/>  <https://www.hackerearth.com/practice/algorithms/string-algorithm/basics-of-string-manipulation/practice-problems/algorithm/conversion-11/>  <https://www.hackerearth.com/practice/algorithms/string-algorithm/basics-of-string-manipulation/practice-problems/algorithm/print-first-occurence/>  <https://www.hackerrank.com/challenges/camelcase/problem>  <https://www.hackerrank.com/challenges/strong-password/problem> | |
| **Dynamic Programming** | <https://www.hackerrank.com/challenges/dynamic-programming-classics-the-longest-common-subsequence/problem>  <https://www.hackerrank.com/challenges/unbounded-knapsack/problem>  <https://www.hackerrank.com/challenges/longest-increasing-subsequent/problem>  <https://www.hackerrank.com/challenges/travel-around-the-world/problem>  <https://www.hackerrank.com/challenges/extremum-permutations/problem>  <https://www.hackerrank.com/challenges/coin-change/problem>  <https://www.hackerearth.com/practice/algorithms/dynamic-programming/introduction-to-dynamic-programming-1/practice-problems/algorithm/win-the-game/>  <https://www.hackerrank.com/challenges/sherlock-and-cost/problem> | |
| **Branch and Bound** | <https://www.hackerrank.com/contests/srin-aadc03/challenges/classic-01-knapsack>  <https://www.hackerrank.com/contests/bitsg-ai-lab-2/challenges/travelling-salesman-problem>  <https://www.hackerrank.com/challenges/assignment/problem> | |
| **Backtracking** | <https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/n-queensrecursion-tutorial/>  <https://www.hackerrank.com/challenges/subset-sum/problem>  <https://www.hackerrank.com/challenges/queens-on-board/problem>  <https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/biggest-forest-700592dd/>  <https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/simran-and-stairs/>  <https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/a-tryst-with-chess/>  <https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/hack-the-money/> | |

**e. Assessment Pattern - Internal and External**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 60 | 40 |
| **Total Marks** | 100 | |

**f. Internal Evaluation Component**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. No. | Type of Assessment | Weightage of actual conduct | Frequency of Task | Final Weightage in Internal  Assessment | Remarks |
| 1 | Conduct | 10 Marks per Practical | 1 per practical | 60 Marks per course |  |
| 2 | Report | 10 Marks per Practical | 1 per practical |  |
| 3 | Viva- Voce | 20 Marks per Course | 1 per Course |  |

**g. CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  | 3 | 3 |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  | 3 | 3 |  |  |  |  |  |  |  |  |  | 3 |  |
| CO4 |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  | 3 | 3 |  |  |  |  |  |  |  | 3 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST371/20CSP376** | **Data Mining** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours: 60 Hours | **3** | **0** | **0** | **0** | **3** |
| **Pre-requisites:** | 20CST316 | | | | | |
| **Co-requisites:** | 20CST311, 20CST313, 20CSP314, 20CST315, 20CST316, 20CST371, 20CST372, 20CST373, 20CST374, 20CSR318, 20CSP312 | | | | | |
| **Anti-Requisites:** | 20CST319 | | | | | |

**Course Objectives**

* To understand the data analysis techniques.
* To understand the concepts behind machine learning.

**Course Outcomes**

* Understand the various stages of data mining process and OLAP with its characteristics.
* Classify Supervised and Unsupervised Learning and understand Regression & Classification techniques.
* Analyse regression & ANOVA approaches.
* Analyse supervised and predictive like K -Nearest Neighbors, Regression and Classification Trees etc.
* Develop the concept of big data mining with its characteristics & challenges.

**Contents of the Syllabus**

**UNIT-I [15h]**

**Chapter-1 (Introduction to Data Mining)**

What is Data Mining, Data Mining Goals, Stages of Data Mining Process, Basic of Related technologies - Machine Learning, DBMS, OLAP, Statistics.

**Chapter-2 (Review of Statistical methods and Data)**

Types of data, Data quality, Data pre-processing.

**Chapter-3 (Machine learning and EDA)**

Supervised and Unsupervised Learning concepts, exploratory data analysis, Regression and Classification techniques, Clustering, Association.

**UNIT-II [15h]**

**Chapter -4 (Regression & ANOVA)**

Regression: Ordinary Least Squares, Ridge Regression, Lasso Regression, ANOVA (Analysis of Variance): Bias-Variance Dichotomy Model Validation Approaches

**Chapter-5 (Supervised and Predictive Analysis)**

K -Nearest Neighbors, Regression and Classification Trees, Support Vector Machines, Dimensionality Reduction

**UNIT-III [15h]**

**Chapter-6 (Unsupervised Learning)**

Clustering-K-means, Associative Rule and market basket analysis-Apriori, recommendation systems.

**Chapter-7 (Big data and Data Analysis)**

Introduction, Characteristics and challenges, Big data vs data science vs data analytics vs Big data analytics, Map reduce.

**ADVANCED TOPICS (BEYOND SYLLABUS)**

Hadoop, HDFS, Data Warehousing

**TEXT BOOKS**

1. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data. John Wiley & Sons, 2015.
2. Jaiwei Han, Micheline Kamber, “Data Mining Concepts and Techniques”, Elsevier, 2006.
3. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.

**REFERENCE BOOKS**

1. Cathy O’Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline. O’Reilly. 2014
2. Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, 2005.
3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann, 2005.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 |  |
| 2 | CO2 |  |
| 3 | CO3 |  |
| 4 | CO4 |  |
| 5 | CO5 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST371/**  **20CSP376** | **Data Mining** |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST371/20CSP376** | **Data Mining** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours: 60 Hours | **0** | **0** | **4** | **0** | **2** |
| **Pre-requisites:** | 20CST316 | | | | | |
| **Co-requisites:** | 20CST311, 20CST313, 20CSP314, 20CST315, 20CST316, 20CST371, 20CST372, 20CST373, 20CST374, 20CSR318, 20CSP312 | | | | | |
| **Anti-Requisites:** | 20CST319 | | | | | |

**Course Objectives**

* To understand the data analysis techniques.
* To understand the concepts behind machine learning.

**Course Outcomes**

* Understand the various stages of data mining process and OLAP with its characteristics.
* Classify Supervised and Unsupervised Learning and understand Regression & Classification techniques.
* Analyse regression & ANOVA approaches.
* Analyse supervised and predictive like K -Nearest Neighbors, Regression and Classification Trees etc.
* Develop the concept of big data mining with its characteristics & challenges.

**List of Experiments**

|  |
| --- |
| **UNIT-I** |
| **Experiment-1**  Demonstration of preprocessing on .arff file using student data .arff. |
| **Experiment-2**  To perform the statistical analysis of data. |
| **Experiment-3**  Demonstration of association rule mining using Apriory algorithm on supermarket data. |
| **Experiment-4**  Demonstration of FP Growth algorithm on supermarket data. |

|  |
| --- |
| **UNIT-II** |
| **Experiment-5**  To perform the classification by decision tree induction using WEKA tools. |
| **Experiment-6**  To perform classification using Bayesian classification algorithm using R. |
| **Experiment-7**  To perform the cluster analysis by k-means method using R. |

|  |
| --- |
| **UNIT-III** |
| **Experiment-8**  To perform the hierarchical clustering using R programming. |
| **Experiment-9**  Study of Regression Analysis using R programming. |
| **Experiment-10**  Outlier detection using R programming. |

**Text Book/Reference Book**

**TEXT BOOKS**

1. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data. John Wiley & Sons, 2015.
2. Jaiwei Han, Micheline Kamber, “Data Mining Concepts and Techniques”, Elsevier, 2006.
3. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.

**REFERENCE BOOKS**

1. Cathy O’Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline. O’Reilly. 2014.
2. Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, 2005.
3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann, 2005.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST372/20CSP377** | **Deep Learning** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours : 60 Hours | **3** | **0** | **0** | **0** | **3** |
| **Pre-requisites:** | 20CST316 | | | | | |
| **Co-requisites** | 20CST311, 20CST313, 20CSP314, 20CST315, 20CST316, 20CST371, 20CST372, 20CSR318, 20CST319, 20CSP312 | | | | | |
| **Anti-Requisites** | 20CST373, 20CST374 | | | | | |

**Course Objectives**

1. To impart fundamental knowledge on emerging fields of sciences like Machine Learning.
2. To impart knowledge on how to apply knowledge in application and research field.

**Course Outcomes**

1. Understand the role of Deep learning in Machine Learning Applications
2. Apply fundamental principles, theory and approaches for learning with deep neural networks
3. Understand key concepts, issues and practices, core algorithms and optimization when training and modeling with deep architectures
4. Analyze the limitation of Applying neural networks through various case studies and study its applications.
5. Develop the skills in using machine learning and deep learning software for solving practical problems.

**Contents of the Syllabus**

**UNIT-I [15h]**

**Chapter - 1 (Introduction)**

Machine Learning Algorithms, Training, Validation, Testing, Types of learning. Supervised, Unsupervised learning, Over-fitting, Under-fitting, Cross validation, Regularization.

**Chapter-2 (Neural Network)**

Artificial and biological neural networks, Neural Networks Characteristics, Introduction to Deep Learning, Challenges motivating deep learning, artificial neural net terminology, model of neuron, application areas.

**UNIT-II [15h]**

**Chapter -3 (Deep Networks Modern Practices)**

Hyper-parameters and validation sets, Types of Neural Networks, Multilayer Perceptron, Feed Forward Neural Networks, Backpropagation, Convolutional Neural Nets (CNN), Self - Organizing Maps (SOMs), Deep Belief Networks (DBNs).

**Chapter-4 (Optimization of Training Deep Models)**

Performance Metrics, Hyper-parameters Optimization, Ensemble Neural Network, Bagging, Boosting, Bootstrapping, Performance Comparison of Deep Learning Neural Network with standard state-of-the-art classifiers.

**UNIT-III [15h]**

**Chapter-5 (Deep Learning Research)**

Feature Selection, Linear Factor Models, large scale deep learning. Overview of Keras and Tensor Flow.

**Chapter-6 (Practical Methodology)**

Application of NN in pattern recognition, Computer Vision and decision making.

**ADVANCED TOPICS (BEYOND SYLLABUS)**

Machine Learning, Artificial Intelligence

**TEXT BOOKS**

1. Goodfellow, Bengio, and Courville’s, “Deep Learning” MIT press book.
2. Yegna Narayanan, “Artificial Neural Networks”. 8th Printing. PHI (2003)

**REFERENCE BOOKS**

1. Patterson Dan W, “Introduction to artificial Intelligence and Expert systems”, 3rd Ed., PHI
2. Jacek M Zaurada, “Introduction to artificial neural Networks Jaico Publishing Home, Fouth Impression.
3. Neural Networks and Deep Learning by Michael Nielsen
4. Deep Learning by Microsoft Research
5. Deep Learning Tutorial by LISA lab, University of Montreal
6. Simon Haykin, “Neural Networks” Pearson Education.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 |  |
| 2 | CO2 |  |
| 3 | CO3 |  |
| 4 | CO4 |  |
| 5 | CO5 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST372/**  **20CSP377** | **Deep Learning** |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST372/20CSP377** | **Deep Learning Lab** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours : 60 Hours | **0** | **0** | **4** | **0** | **2** |
| **Pre-requisites:** | 20CST316 | | | | | |
| **Co-requisites** | 20CST311, 20CST313, 20CSP314, 20CST315, 20CST316, 20CST371, 20CST372, 20CSR318, 20CST319, 20CSP312 | | | | | |
| **Anti-Requisites** | 20CST373, 20CST374 | | | | | |

**Course Objectives**

1. To impart fundamental knowledge on emerging fields of sciences like Machine Learning.
2. To impart knowledge on how to apply knowledge in application and research field.

**Course Outcomes**

1. Understand the role of Deep learning in Machine Learning Applications
2. Apply fundamental principles, theory and approaches for learning with deep neural networks
3. Understand key concepts, issues and practices, core algorithms and optimization when training and modeling with deep architectures
4. Analyze the limitation of Applying neural networks through various case studies and study its applications.
5. Develop the skills in using machine learning and deep learning software for solving practical problems.

**List of Experiments**

|  |
| --- |
| **UNIT-I** |
| **Experiment-1**  Implement and demonstrate the FIND-S algorithm for finding the most specific  hypothesis based on a given set of training data samples. Read the training data from a  .CSV file. |
| **Experiment-2**  For a given set of training data examples stored in a .CSV file, implement and  demonstrate the Candidate-Elimination algorithm to output a description of the set of all  hypotheses consistent with the training examples. |
| **Experiment-3**  Write a program to demonstrate the working of the decision tree based ID3 algorithm.  Use an appropriate data set for building the decision tree and apply this knowledge  To classify a new sample. |
| **Experiment-4**  Build an Artificial Neural Network by implementing the Back propagation algorithm  and test the same using appropriate data sets. |

|  |
| --- |
| **UNIT-II** |
| **Experiment-5**  Write a program to implement the naïve Bayesian classifier for a sample training data  set stored as a .CSV file. Compute the accuracy of the classifier, considering few test  data sets. |
| **Experiment-6**  Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier  model to perform this task. Built-in Java classes/API can be used to write the program.  Calculate the accuracy, precision, and recall for your data set. |
| **Experiment-7**  Write a program to construct a Bayesian network considering medical data. Use this  model to demonstrate the diagnosis of heart patients using standard Heart Disease Data  Set. You can use Java/Python ML library classes/API. |

|  |
| --- |
| **UNIT-III** |
| **Experiment-8**  Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set  for clustering using k-Means algorithm. Compare the results of these two algorithms  and comment on the quality of clustering. You can add Java/Python ML library  classes/API in the program. |
| **Experiment-9**  Write a program to implement k-Nearest Neighbour algorithm to classify the iris data  set. Print both correct and wrong predictions. Java/Python ML library classes can be  used for this problem. |
| **Experiment-10**  Implement the non-parametric Locally Weighted Regression algorithm in order to fit  data points. Select appropriate data set for your experiment and draw graphs. |

**Text Book/Reference Book**

**TEXT BOOKS**

1. Goodfellow, Bengio, and Courville’s, “Deep Learning” MIT press book.
2. Yegna Narayanan, “Artificial Neural Networks”. 8th Printing. PHI (2003)

**REFERENCE BOOKS**

1. Patterson Dan W, “Introduction to artificial Intelligence and Expert systems”, 3rd Ed., PHI
2. Jacek M Zaurada, “Introduction to artificial neural Networks Jaico Publishing Home, Fouth Impression.
3. Neural Networks and Deep Learning by Michael Nielsen
4. Deep Learning by Microsoft Research
5. Deep Learning Tutorial by LISA lab, University of Montreal
6. Simon Haykin, “Neural Networks” Pearson Education.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code 20CST315** | **SYSTEM PROGRAMMING** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites:** |  | | | | | |
| **Co-requisites** |  | | | | | |
| **Anti-Requisites** |  | | | | | |

**Course Objectives:**

* Design, write, and test moderately complicated low-level programs using a systems programming language.
* Proficiently use a preprocessor to implement code that is portable between different computing platforms.
* Implement routines that read and write structured binary files such as word processing documents, index systems, or serialized hierarchical data

**Course Outcomes:**

* To Understand different components of system software.
* To Understand the different phases and data structure used in assembly process by an assembler.
* To intermediate code generation in context of language designing.
* To Analyze the role of linkers and loaders in executing the programs
* To Recognize operating system functions such as memory management as pertaining to run time storage management.

**Contents of the Syllabus:**

**UNIT-I**

**Chapter-1 (Overview of System Software)**

**[15h]**

Introduction, Software, Software Hierarchy, Systems Programming, Machine Structure, Interfaces, Address Space, Computer Languages, Tools, Life Cycle of a Source Program, Different Views on the Meaning of a Program, System Software Development, Recent Trends in Software Development, Levels of System Software.

**Chapter-2 (Assemblers)**

Elements of Assembly Language Programming, Design of the Assembler, Assembler Design Criteria, Types of Assemblers, Two-Pass Assemblers, One-Pass Assemblers, Single pass Assembler for Intel x86 , Algorithm of Single Pass Assembler, Multi-Pass Assemblers, Advanced Assembly Process, Variants of Assemblers Design of two pass assembler

**Chapter-3 (Macros)**

Macro and Macro Processors Introduction, Macro Definition and Call, Macro Expansion, Nested Macro Calls, Advanced Macro Facilities, Design Of a Macro Preprocessor, Design of a Macro Assembler, Functions of a Macro Processor, Basic Tasks of a Macro Processor, Design Issues of Macro Processors, Features, Macro Processor Design Options, Two-Pass Macro Processors, One-Pass Macro Processors

**UNIT-II [15h]**

**Chapter -4(Compilers)**

Introduction to various translators, Various phases of compiler, Introduction to Grammars and finite automata, Bootstrapping for compilers, Lexical Analysis and syntax analysis, Intermediate Code Generation, Code optimization techniques, Code generation, Case study :LEXX and YACC, Design of a compiler in C++ as Prototype.

**Chapter -5(Scanning and Parsing)**

Scanning and Parsing Programming Language Grammars, Classification of Grammar, Ambiguity in Grammatic Specification, Scanning, Parsing, Top Down Parsing, Bottom up Parsing, Language Processor Development Tools, LEX, YACC

**Chapter -6(Debuggers)**

Introduction to various debugging techniques, Case Study: - Debugging in Turbo C++ IDE.

**UNIT-III [15h]**

**Chapter -7 (Linkers and Loaders)**

Introduction, Relocation of Linking Concept, Design of a Linker, SelfRelocating Programs, Linking in MSDOS, Linking of Overlay Structured Programs, Dynamic Linking, Loaders, Different Loading Schemes, Sequential and Direct Loaders, Compile-and-Go Loaders, General Loader Schemes, Absolute Loaders, Relocating Loaders, Practical Relocating Loaders, Linking Loaders, Relocating Linking Loaders, Linkers v/s Loaders

**Chapter -8 (Editors)**

Line editor, full screen editor and multi window editor, Case study MS-Word, DOS Editor and vi editor.

**Chapter – 9 (Operating System)**

Booting techniques and sub-routines, Design of kernel and various management for OS, Design of Shell and other utilities.

**ADVANCED TOPICS (BEYOND SYLLABUS)**

**Macro and Macro Processors, Scanning and Parsing**

## Text Books:

1. Donovan J.J., Systems Programming, New York, Mc-Graw Hill,1972.
2. Dhamdhere, D.M., Introduction to Systems Software, Tata Mc-Graw Hill1996.

## Reference Books:

* 1. Aho A.V. and J.D. Ullman Principles of compiler Design Addison Wesley/Narosa

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO2 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO3 |  | 2 | 3 | 2 |  |  |  |  | 1 | 2 |  | 3 | 3 |  |
| CO4 |  | 3 | 2 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO5 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST374/20CSP379** | **UI/UX DESIGN** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites:** |  | | | | | |
| **Co-requisites** |  | | | | | |
| **Anti-Requisites** |  | | | | | |

**Course Objectives:**

* Describe the web user Interface
* Describe the structure of user Interface and design process
* Organize the web systems and control.

**Course Outcomes: (5 Required)**

* Students are expected to describe the Characteristics of Graphics Interface and its Principles.
* Students are expected to design the standards and structures for Human computer interaction.
* Students are expected to explain the technical details of multimedia data representations.
* Student are expected Decompose a problem into function
* Students are expected Develop an application using the concepts of array, pointer, structure, and file management to solve engineering and/or scientific problems

**Contents of the Syllabus:**

**UNIT-I [15h]**

**Chapter-1(Introduction)**

Contents: Human–Computer Interface, Characteristics Of Graphics Interface, Direct Manipulation Graphical System, Web User Interface, Popularity, Characteristic & Principles.

**Chapter-2(Human Computer Interaction)**

# User Interface Design Process, Obstacles, Usability, Human Characteristics In Design, Human Interaction Speed, Business Functions, Requirement Analysis, Direct, Indirect Methods, Basic Business Functions, Design Standards, System Timings, Human Consideration In Screen Design, Structures Of Menus.

Functions Of Menus–Contents Of Menu, Formatting, Phrasing The Menu, Selecting Menu Choice–Navigating Menus, Graphical Menus.

**UNIT-II [15h]**

**Chapter -3(Windows)**

Contents: Characteristics, Components, Presentation Styles, Types, Managements–Organizations, Operations, Web Systems, Device, Based Controls Characteristics–Screen, Based Controls, Operate Control, Text Boxes, Selection Control–Combination Control, Custom Control, Presentation Control.

**UNIT-III [15h]**

**Chapter -4 (Multimedia)**

Text For Web Pages, Effective Feedback, Guidance & Assistance–Internationalization, Accessibility, Icons, Image, Multimedia, Coloring.

**Chapter -5 (Window Layout test)**

Contents: Prototypes, Kinds Of Tests, Retest, Information Search, Visualization, Hypermedia, WWW, Software Tools..

**TEXT BOOKS**

1. Wilbent. O. Galitz ,“The Essential Guide To User Interface Design”, John Wiley& Sons, 2001.
2. Ben Sheiderman, “Design The User Interface”, Pearson Education, 1998.**T3.** Li & Drew, “ Fundamentals of Multimedia” , Pearson Education, 2009.

**REFERENCE BOOKS**

1. Alan Cooper, “The Essential Of User Interface Design”, Wiley-Dream Tech Ltd.,2002.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST373/20CSP378** | **WEB TECHNOLOGIES** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites:** |  | | | | | |
| **Co-requisites** |  | | | | | |
| **Anti-Requisites** |  | | | | | |

**Course Objectives:**

* This Subject is useful for Making own Web page and how to host own web site on internet. Along with that Students will also learn about the protocols involv in internet technology.

**Course Outcomes: (5 Required)**

* History and development of the World Wide Web and associated technologies.
* The client-server architecture of the World Wide Web and its communication protocol HTTP/HTTPS.
* Formats and languages used in modern web-pages: HTML, XHTML, CSS, XML, XSLT, JavaScript, DOM.
* Programming web pages with JavaScript/DOM (client)
* Good design, universal design, multi-platform web applications.

**Contents of the Syllabus:**

**UNIT-I [15h]**

**Chapter-1(Introduction)**

**Introduction to WWW:** Protocols and programs, secure connections, application and development tools, the web browser, What is server, choices, setting up UNIX and Linux web servers, Logging users, dynamic IP

**Web Design**: Web site design principles, planning the site andnavigation.

**Chapter-2(Introduction to HTML)**

**Introduction to HTML:** The development process, Html tags and simple HTML forms, web site structure **Introduction to XHTML** : XML, Move to XHTML, Meta tags, Character entities, frames and frame sets,insidebrowser.

**UNIT-II [15h]**

**Chapter -3(Style Sheets)**

Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, bordersand boxes, margins, padding lists, positioning using CSS, CSS2.

**Chapter -4(Java Script)**

**Javascript**: Client side scripting, What is Javascript, How to develop Javascript, simple Javascript, variables, functions, conditions, loops and repetition

**Advance script**, Javascript and objects, Javascript own objects, the DOM and web browser environments, forms and validations

**DHTML** : Combining HTML, CSS and Javascript, events and buttons, controlling yourbrowser, **Ajax:** Introduction, advantages & disadvantages ,Purpose of it ,ajax based web application, alternatives ofajax.

**UNIT-III [15h]**

**Chapter -5 (XML)**

**XML** : Introduction to XML, uses of XML, simple XML, XML key components, DTD and Schemas, Well formed, using XML with application.XML, XSL andXSLT.

Introduction to XSL, XML transformed simple example, XSL elements, transforming with XSLT.

**Chapter -6 (PHP)**

Hash Table, Hash Functions, Collision Resolution Strategies, Hash Table Implementation.

**TEXT BOOKS/REFERENCE BOOKS**

1. Steven Holzner, ”HTML Black Book”, Dremtechpress.
2. Web Technologies, Black Book, DreamtechPress
3. Web Applications : Concepts and Real World Design, Knuckles,Wiley-India
4. Internet and World Wide Web How to program, P.J. Deitel& H.M. DeitelPearson..

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-211** | **Data Structures** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code** | **TITLE OF THE COURSE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
| **1** | **Engineering Encapsulation/Technical Training** | **0** | **0** | **2** | **0** | **1** | **2** | **EE** |
| **Subject Code**  **20CSP-356** | |  | | | | **Course Code(s)**  **20CSP-356** | | | |
| **PRE-REQUISITE** | | **20CSP-356** | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

**a. Course Description**

Competitive programming is the course in which students will learn how to apply algorithms in order to solve complex problems. The goal of this course is to teach students how to apply familiar algorithms to non-intuitive problems.

**b. Course Objectives**

Competitive programming is the course in which students will learn how to apply algorithms in order to solve complex problems.

The goal of this course is to teach students how to apply familiar algorithms to non-intuitive problems.

**c. Course Outcomes**

|  |  |
| --- | --- |
| CO1 | Describe how algorithmic problems are solved |
| CO2 | Recognize the time and memory complexity of an algorithm or a structure |
| CO3 | Explain the concrete algorithms and data structures |
| CO4 | Analyze the given problem and recognize sub-problems |
| CO5 | Apply the knowledge on a wider set of problems to achieve better time complexity. |

|  |  |  |
| --- | --- | --- |
| **Topic Name** | **Subdomain** | **Task** |
| **Data Structures** | **Arrays,**  **Stacks & Queues** | https://www.hackerrank.com/challenges/arrays-ds/problem |
| https://www.hackerrank.com/challenges/2d-array/problem |
| https://www.hackerrank.com/challenges/dynamic-array/problem |
| https://www.hackerrank.com/challenges/array-left-rotation/problem |
| https://www.hackerrank.com/challenges/crush/problem |
| https://www.hackerrank.com/challenges/equal-stacks/problem |
| https://www.hackerrank.com/challenges/maximum-element/problem |
| https://www.hackerrank.com/challenges/truck-tour/problem |
| https://www.hackerrank.com/challenges/queue-using-two-stacks/problem |
| **Linked List** | https://www.hackerrank.com/challenges/print-the-elements-of-a-linked-list/problem |
| https://www.hackerrank.com/challenges/insert-a-node-at-the-tail-of-a-linked-list/problem |
| https://www.hackerrank.com/challenges/compare-two-linked-lists/problem |
| https://www.hackerrank.com/challenges/reverse-a-linked-list/problem |
| https://www.hackerrank.com/challenges/insert-a-node-into-a-sorted-doubly-linked-list/problem |
| https://www.hackerrank.com/challenges/reverse-a-doubly-linked-list/problem |
| **Sorting & Searching** | https://www.hackerrank.com/challenges/insertionsort1/problem |
| https://www.hackerrank.com/challenges/insertionsort2/problem |
| https://www.hackerrank.com/challenges/quicksort1/problem |
| https://www.hackerearth.com/practice/algorithms/sorting/bubble-sort/practice-problems/algorithm/min-max-difference/ |
| https://www.hackerearth.com/practice/algorithms/sorting/selection-sort/practice-problems/algorithm/old-keypad-in-a-foreign-land-24/ |
| https://www.hackerearth.com/practice/algorithms/searching/linear-search/practice-problems/ |
| https://www.hackerearth.com/practice/algorithms/searching/binary-search/practice-problems/ |
| https://www.hackerrank.com/challenges/countingsort1/problem |
| https://www.hackerrank.com/challenges/countingsort2/problem |
|  | **Graphs** | https://www.hackerrank.com/challenges/bfsshortreach/problem |
| https://www.hackerearth.com/practice/algorithms/graphs/breadth-first-search/practice-problems/algorithm/monk-and-the-islands/ |
| https://www.hackerearth.com/practice/algorithms/graphs/depth-first-search/practice-problems/algorithm/anupam-graph/ |
| https://www.hackerrank.com/challenges/kruskalmstrsub/problem |
| https://www.hackerrank.com/challenges/primsmstsub/problem |
| https://www.hackerrank.com/challenges/dijkstrashortreach/problem |
| https://www.hackerrank.com/challenges/minimum-mst-graph/problem |
| https://www.hackerrank.com/challenges/clique/problem |
| https://www.hackerrank.com/challenges/crab-graphs/problem |
| https://www.hackerrank.com/challenges/the-quickest-way-up/problem |
|  | **Trees** | https://www.hackerrank.com/challenges/tree-preorder-traversal/problem |
| https://www.hackerrank.com/challenges/tree-postorder-traversal/problem |
| https://www.hackerrank.com/challenges/tree-inorder-traversal/problem |
| https://www.hackerrank.com/challenges/tree-height-of-a-binary-tree/problem |
| https://www.hackerrank.com/challenges/tree-level-order-traversal/problem |
| https://www.hackerrank.com/challenges/binary-search-tree-insertion/problem |
| https://www.hackerrank.com/challenges/binary-search-tree-lowest-common-ancestor/problem |
| https://www.hackerrank.com/challenges/self-balancing-tree/problem |
| https://www.hackerrank.com/contests/hourrank-19/challenges/maximal-tree-diameter/problem |
| https://www.hackerrank.com/contests/101hack33/challenges/longest-path |
| **Algorithms** | **String Algorithms** | https://www.hackerearth.com/practice/algorithms/string-algorithm/string-searching/practice-problems/algorithm/string-4-d1093b86/ |
| https://www.hackerearth.com/practice/algorithms/string-algorithm/string-searching/practice-problems/algorithm/palindrome-string-22/ |
| https://www.hackerearth.com/practice/algorithms/string-algorithm/basics-of-string-manipulation/practice-problems/algorithm/palindrome-124/ |
| https://www.hackerearth.com/practice/algorithms/string-algorithm/basics-of-string-manipulation/practice-problems/algorithm/conversion-11/ |
| https://www.hackerearth.com/practice/algorithms/string-algorithm/basics-of-string-manipulation/practice-problems/algorithm/print-first-occurence/ |
| https://www.hackerrank.com/challenges/camelcase/problem |
| https://www.hackerrank.com/challenges/strong-password/problem |
| **Dynamic Programming** | https://www.hackerrank.com/challenges/dynamic-programming-classics-the-longest-common-subsequence/problem |
| https://www.hackerrank.com/challenges/unbounded-knapsack/problem |
| https://www.hackerrank.com/challenges/longest-increasing-subsequent/problem |
| https://www.hackerrank.com/challenges/travel-around-the-world/problem |
| https://www.hackerrank.com/challenges/extremum-permutations/problem |
| https://www.hackerrank.com/challenges/coin-change/problem |
| https://www.hackerearth.com/practice/algorithms/dynamic-programming/introduction-to-dynamic-programming-1/practice-problems/algorithm/win-the-game/ |
| https://www.hackerrank.com/challenges/sherlock-and-cost/problem |
| **Branch and Bound** | https://www.hackerrank.com/contests/srin-aadc03/challenges/classic-01-knapsack |
| https://www.hackerrank.com/contests/bitsg-ai-lab-2/challenges/travelling-salesman-problem |
| https://www.hackerrank.com/challenges/assignment/problem |
| **Backtracking** | https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/n-queensrecursion-tutorial/ |
| https://www.hackerrank.com/challenges/subset-sum/problem |
| https://www.hackerrank.com/challenges/queens-on-board/problem |
| https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/biggest-forest-700592dd/ |
| https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/simran-and-stairs/ |
| https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/a-tryst-with-chess/ |
| https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practice-problems/algorithm/hack-the-money/ |

**d. Assessment Pattern - Internal and External**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 60 | 40 |
| **Total Marks** | 100 | |

**e. Internal Evaluation Component**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. No. | Type of Assessment | Weightage of actual conduct | Frequency of Task | Final Weightage in Internal  Assessment | Remarks |
| 1 | Conduct | 10 Marks per Practical | 1 per practical | 60 Marks per course |  |
| 2 | Report | 10 Marks per Practical | 1 per practical |  |
| 3 | Viva- Voce | 20 Marks per Course | 1 per Course |  |

**f. CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO2 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO3 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO4 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO5 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code** | **TITLE** | **L** | **T** | **P** | **C** | **CH** | **Course Type\*** |
| 3 | **WEB AND MOBILE SECURITY (Professional Elective-I)** | 3 | 0 | 0 | 3 | 3 | DE |
| **20CST-333** | |  | | | | **20CST-333** | | |
| **PRE-REQUISITE** | | 20CST-254 | | | |  | | |
| **CO-REQUISITE** | | 20CST-313,20CSP-314,20CST-315,20CST-316,20CST-319,20CSP-321 | | | |  | | |
| **ANTI-REQUISITE** | | 20CST-355 | | | |  | | |

**Course Objectives**

This Subject is useful for Making own Web page and how to host own web site on internet. Along with that Students will also learn about the protocols involve in internet technology.

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | Use network-based tools for network analysis |
| CO2 | Use techniques for Network scanning |
| CO3 | Identify network vulnerability |
| CO4 | Use tools to simulate intrusion detection system |
| CO5 | To understand and install a firewall |

**Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **Introduction to Soft Computing** | **Contact Hours: 15** |
| Ch-1.1 | Web Fundamentals – HTML, HTTP, Client-side scripting, Server-side scripting. | |
| Ch-1.2 | Web server architecture - Windows & Linux, IIS and LAMP servers, Network topologies and DMZ. | |
| Ch-1.3 | Mobile Security Fundamentals- Introduction to Mobile Security, Building Blocks – Basic security and cryptographic techniques. | |
| **Unit-2** | **Artificial Neural Network and Associative Memory Networks** | **Contact Hours: 15** |
| Ch-2.1 | Web applications: Introduction to web applications, Web application hacking, Overview of browsers, extensions, and platforms.  Mobile applications and Models: Mobile Malware and App Security, Android Security Model , IOS Security Model , Security Model of the Windows Phone | |
| Ch-2.2 | Web Security-Attacks, detection evasion techniques, and countermeasures for the most popular web platforms, including IIS, Apache, PHP, and ASP.NET  Attacks and countermeasures for common web authentication mechanisms, including password-based, multifactor (e.g., CAPTCHA), and online authentication services like Windows Live ID. | |
| Ch-2.3 | Mobile Security-Security of GSM Networks , Security of UMTS Networks, LTE Security, WiFi and Bluetooth Security , SIM/UICC Security. | |
| **Unit-3** | **Genetic Algorithm and related case studies** | **Contact Hours: 15** |
| Ch-3.1 | Advanced session analysis, hijacking, and fixation techniques, cross-site scripting, SQL  injection, classic categories of malicious input, Overlong input (like buffer overflows), canonicalization attacks (like the infamous dot-dot-slash), and meta characters (including angle brackets, quotes, single quote, double dashes, percent, asterisk, underscore, newline, ampersand, pipe, and semicolon), beginner-to-advanced SQL injection tools and techniques, stealth-encoding techniques and input validation/ output-encoding countermeasures. | |
| Ch-3.2 | Emerging Trends in Mobile Security-Mobile Geo-location and Mobile Web Security, Security of Mobile VoIP Communications. | |
| Ch-3.3 | Web services vulnerabilities discovery and exploited through techniques including WSDL  disclosure, input injection, external entity injection, and XPath injection. Web application management attacks against remote server management, web content management/authoring, admin misconfigurations, and developer-driven mistakes. Web browser exploits. | |

**Textbooks / Reference Books**

* Hacking Exposed Web Applications, 3rd edition, JOEL SCAMBRAY, VINCENT LIU, CALEB SIMA
* The Web Application Hacker's Handbook Discovering and Exploiting Security Flaws By Dafydd Stuttard, Marcus Pinto
* Rich Bowen, Ken Coar, “Apache Cookbook”, O’Reilly
* Open Web Application Security Project. A Guide to Building Secure Web Applications and Web Services. http://www.owasp.org/index.php/Category:OWASP\_Guide\_Project

**Mode of Evaluation: The performance of students is evaluated as follows:**

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 40 | 60 |
| **Total Marks** | 100 | |

**CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 |  |  |  | - | - | - | - | - | - | - | - | - | - |
| CO2 | 3 |  | 3 |  | - | - | - | - | - | - | - | - | 3 | 2 |
| CO3 |  | 3 |  |  | - | - | - | - | - | - | - | - | 3 | 2 |
| CO4 |  |  | 3 |  | - | - | - | - | - | - | - | - | - | - |
| CO5 |  | 3 |  | 2 | - | - | - | - | - | - | - | - | 3 | 2 |

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

**CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 |  |  |  | - | - | - | - | - | - | - | - | - | - |
| CO2 | 3 |  | 3 |  | - | - | - | - | - | - | - | - | 3 | 2 |
| CO3 |  | 3 |  |  | - | - | - | - | - | - | - | - | 3 | 2 |
| CO4 |  |  | 3 |  | - | - | - | - | - | - | - | - | - | - |
| CO5 |  | 3 |  | 2 | - | - | - | - | - | - | - | - | 3 | 2 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Program Code-CS201** | **Course Title** | **L** | **T** | **P** | **C** | **CH** | **Course Type** |
| **5** | **Course Code(s)**  **20CST-355** | **Mobile Application Development with Lab** | 2 | 0 | 2 | 3 | 4 | CR |
| **PRE-REQUISITE** | | 21CSH-319- Project Based learning in java with Lab | | | | | | |
| **CO-REQUISITE** | | - | | | | | | |
| **ANTI-REQUISITE** | | - | | | | | | |

**Course Objectives**

* Install and configure Android application development tools.
* Design and develop user Interfaces for the Android platform.
* Save state information across important operating system events.
* Apply Java programming concepts to Android application development.

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | Design and develop User Interfaces for the Android platform. |
| CO2 | Ability to apply general programming knowledge in the field of developing mobile applications. |
| CO3 | Understanding of the specific requirements, possibilities and challenges when developing for a mobile context. |
| CO4 | Understanding of the interactions between user interface and underlying application infrastructure. |
| CO5 | At the end of this course student will Apply essential Android Programming concepts. |

**Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **Introduction to Android** | **Contact Hours: 15** |
| **Chapter 1.1** | **Introduction:**  Cost of Mobile Application Development, Importance of Mobile Strategies, Challenges, Myths, Third-Party Frameworks, Mobile Web Presence, ApplicationsFactors in Developing Mobile Applications :Mobile Software Engineering ,Frameworks and Tools, Generic UI Development ,Android User | |
| **Chapter 1.2** | **Introduction to Mobility:**Mobility Landscape, Mobile Platforms**,**Mobile apps development, Overview of Android Platform, Setting up the mobile apps development environment with emulator.Telephony :a. Deciding Scope of an App b. Wireless Connectivity and Mobile Apps  c. Android Telephony. | |
| **Chapter 1.3** | **Building block of Mobile apps:**App user Interface Designing, Layout, User Interface elements, Draw-able, Menu, Activity states and lifecycle, Interaction among activities.  **App functionality based user interface**:Threads, Asynchronous task, Services-states and lifecycle, Notifications, Broadcast receivers, Telephony and SMS API**.** | |
| Experiment No 1.1 | Setting Up the Development Environment | |
| Experiment No 1.2 | Create "Hello World" Application | |
| Experiment No 1.3 | Create Application by Using Widgets | |
| **Unit-2** | **Data Handling** | **Contact Hours: 15** |
| **Chapter 2.1** | **Naïve Data Handling:**On Device File I/O, Shared preferences, Mobile Databases such as SQLite and enterprise data access. | |
| **Chapter 2.2** | **Sprucing up Mobile Apps:**Graphics and animation-custom views, canvas, animation API multimedia-audio/video playback and record, location aware.  **Testing Mobile apps**: Debugging Apps, White and Black Box Testing and test automation of apps. | |
| **Chapter 2.3** | **Creating Consumable Web Services for Mobile Devices:**What is a Web Service, Web Services Languages (Formats), Creating an Example Web Service, Debugging Web Services | |
| Experiment No 2.1 | Creating the Application by using Tex Edit control | |
| Experiment No 2.2 | Creating the Application Choosing Options CheckBox | |
| Experiment No 2.3 | Creating the Application Choosing Options RadioButton | |
| Experiment No. 2.4 | Creating the Application Choosing Options RadioGroup | |
| **Unit-3** | **User Interface Design** | **Contact Hours:15** |
| **Chapter 3.1** | **Mobile User Interface Design:**Effective Use of Screen Real Estate, Understanding Mobile Information Design, Understanding Mobile Application Users, Understanding Mobile Platforms, Using the Tools of Mobile Interface Design. | |
| **Chapter 3.2** | **Mobile Websites:**Choosing a Mobile Web Option, Adaptive Mobile Websites, Dedicated Mobile Websites Mobile Web Apps with HTML5  **Android:**Android as Competition to itself, Connecting to the Google Play, Android Development Practices, Building an App in Android. | |
| **Chapter 3.3** | **Operating Systems iOS:**IOS Project, Debugging iOS Apps, Objective-C Basics, Building the Derby App in IOS  **Windows Phone 7:**Windows Phone 7 Project, Building an App in Windows Phone 7, Distribution. | |
| Experiment No 3.1 | Create Application by Using Building Blocks for Android Application (Design by using Linear Layout) | |
| Experiment No 3.2 | Create Application by Using Building Blocks for Android Application (Design by using Relative Layout) | |
| Experiment No 3.3 | Create Application by Using Building Blocks for Android Application (Design by using Absolute Layout) | |
| Experiment No 3.4 | Create Application by Using Building Menus and Storing Data | |
| Experiment No 3.5 | Design the Application for Menus and Action Bar | |

**Text Books:**

1.     Jeff Mcwherter, Scott Gowell, Professional Mobile Application Development, Wrox Publisher (2012), 1st Ed.

**Reference Books:**

1.     Lauren Darcy, Shane Conder, Sams Teach Yourself Android Application Development in 24 Hrs, 1st ed.

2.      Himanshu Dwivedi, Chris Clark, David Thiel, Mobile Application Security, Tata McGraw Hill (2010), 1st Edition.

**Mode of Evaluation: The performance of students is evaluated as follows:**

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 40 | 60 |
| **Total Marks** | 100 | |

**CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 |  |  |  | - | - | - | - | - | - | - | - | - | - |
| CO2 | 3 |  | 3 |  | - | - | - | - | - | - | - | - | 3 | 2 |
| CO3 |  | 3 |  |  | - | - | - | - | - | - | - | - | 3 | 2 |
| CO4 |  |  | 3 |  | - | - | - | - | - | - | - | - | - | - |
| CO5 |  | 3 |  | 2 | - | - | - | - | - | - | - | - | 3 | 2 |

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST354** | **Introduction to Information Security** | **L** | **T** | **P** | **C** |
| Total Contact Hours : 45Hours | **3** | **0** | **0** | **3** |
| Common to all Specializations of CSE is 4thYear |
| Prerequisite: Studied computer networks | | | | |

Course Objectives

1. To familiarize the students with the basic concepts of services, attacks with its models and concepts of encryption.
2. To conceptualize digital signature and different encryption algorithm. To state the various authentication protocols and their requirements.
3. To elucidate an applications of security and their effects on security standards.
4. To comprehend IP security and their methods.
5. To familiarize the student this basic encryption and decryptions

Course Outcomes

|  |  |
| --- | --- |
| **CO1** | Discuss the basics of network security and cryptography. |
| **CO2** | Explain the various standards Symmetric Encryption algorithms used to provide confidentiality. |
| **CO3** | Explain the various standards Asymmetric Encryption algorithms to achieve authentication. |
| **CO4** | Demonstrate encryption techniques to secure data in transit across network. |
| **CO5** | Explore the knowledge of key exchange protocols. |
| **CO6** | Examine the effects on digitized security. |

### UNIT-I

**Introduction & Number Theory:** Services, Mechanisms and attacks-the OSI security architecture-Network security model-Classical Encryption techniques (Symmetric cipher model, substitution techniques, transposition techniques, steganography).FINITE FIELDS AND NUMBER THEORY: Groups, Rings, Fields-Modular arithmetic-Euclid‟s algorithm-Finite fields- Polynomial Arithmetic –Prime numbers- Fermat‟s and Euler‟s theorem-Testing for primality -The Chinese remainder theorem- Discrete alogarithms.

**Block Ciphers :**Data Encryption Standard-Block cipher principles-block cipher modes of operation- Advanced Encryption Standard (AES)-Triple DES-Blowfish-RC5 algorithm.

**UNIT II**

**Public key cryptography:** Principles of public key cryptosystems-The RSA algorithm-Key management – Diffie Hellman Key exchange

**Hash Functions And Digital Signatures:** Authentication requirement – Authentication function – MAC – Hash function – Security of hash function and MAC – MD5–SHA512–HMAC – CMAC – Digital signature and authentication protocols – DSS – EI Gamal – Schnorr.

**UNIT III**

**Security Practice & System Security:** Authentication applications – Kerberos – Authentication services – Internet Firewalls for Trusted System: Roles of Firewalls – Firewall related terminology- Types of Firewalls – Firewall designs –Intruder – Intrusion detection system – Virus and related threats.

**E-mail Security:** Security Services for E-mail-attacks possible through E-mail – establishing keys privacy- authentication of the source-Message Integrity-Non-repudiation-Pretty Good Privacy-S/MIME.

**IPSecurity and Web Security:** Overview of IPsec – IP and IPv6-Authentication Header-Encapsulation Security Payload (ESP)- SSL Architecture and its layers.

### Text Books:

1. William Stallings, Cryptography and Network Security, 6th Edition, Pearson Education, March 2013.
2. Charlie Kaufman, Radia Perlman and Mike Speciner, “Network Security*”*, Prentice Hall of India, 2002.

### Reference Books:

1. Behrouz A. Ferouzan, “Cryptography & Network Security”, Tata McGrawHill.
2. Man Young Rhee, “Internet Security: Cryptographic Principles”, “Algorithms and Protocols”, WileyPublications.
3. Charles Pfleeger, “Security in Computing”, 4th Edition, Prentice Hall ofIndia.
4. Ulysess Black, “Internet Security Protocols”, Pearson EducationAsia.
5. Charlie Kaufman and Radia Perlman, Mike Speciner, “Network Security, Second Edition, Private Communication in Public World”,PHI.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | To familiarize the students with the basic concepts of services, attacks with its models and concepts of encryption. | Discuss the basics of network security and cryptography. |
| 2 | **To conceptualize digital signature and different encryption algorithm.** | Explain the various standards Symmetric Encryption algorithms used to provide confidentiality. |
| 3 | To elucidate an applications of security and their effects on security standards. | Explain the various standards Asymmetric Encryption algorithms to achieve authentication. |
| 4 | To comprehend IP security and their methods. | Demonstrate encryption techniques to secure data in transit across network. |
| 5 | To familiarize the student this basic encryption and decryptions | Explore the knowledge of key exchange protocols. |
|  |  | Examine the effects on digitized security. |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Mapping Between COs and POs** | | | | | | | | | | | | | |
| **COs** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | | **PSO2** |
| **CO1** | 1 | – | – | 2 | – | – | – | – | – | – | – | – | – | | – |
| **CO2** | 1 | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | | 2 |
| **CO3** | 1 | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | | 2 |
| **CO4** | 2 | 2 | 1 | 1 | – | – | – | – | – | – | – | – | – | | 2 |
| **CO5** | 2 | 2 | 2 | 1 | – | – | – | – | – | – | – | – | – | | 3 |
| **CO6** | 3 | 3 | 3 | 2 | – | – | – | – | – | – | – | – | – | | 3 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST354** | **Introduction to Information Security** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Subject Code (20CSP354)** | **Information Security Lab** | **L** | **T** | **P** | **C** |
| Total Contact Hours : 10 Hours | **0** | **0** | **2** | **0** |
| CSE 3rd Year |
| Prerequisite: Studied Encryption Techniques | | | | |

**Course Objectives**

1.To generate encrypted text using different techniques.

2.To understand the cipher-text in order to provide more security to data.

3. To understand the concepts of Digital Signature and its implementation.

4. Design and develop a security architecture for an organization.

5 Design operational and strategic cyber security strategies and policies.

Course Outcomes

1. Analyze and evaluate the cyber security needs of an organization.

2 . Determine and analyze software vulnerabilities and security solutions to reduce the risk of exploitation.

3. Measure the performance and troubleshoot cyber security systems.

4. Implement cyber security solutions and use of cyber security, information assurance, and cyber/computer forensics software/tools.

5. Comprehend and execute risk management processes, risk treatment methods, and key risk and performance indicators

**List of Experiments**

**UNIT-I**

1. Perform the practical of CaesarCipher.
2. Perform the practical of Substitutioncipher.
3. Perform the practical of Hill Cipher.
4. Perform the practical of Playfairalgorithm

### UNIT-II

1. Perform the practical of Diffie hellman key exchangealgorithm.
2. Perform the practical of RSAalgorithm.
3. Perform the practical ofMD5

### UNIT-III

1. Perform the practical ofSHA1.
2. Perform the practical to generate digital certificate.
3. Perform the practical of digital signature ontext.

CO1:Understand the conceptual foundation of information security awareness.

CO2:Analysis the risk events, treatment plans, assessment

CO3:Detail evaluation of information classification, roles and responsibilities

CO4:Examining the access controls, monitoring, management and review process

CO5:Study the physical and logical perimeters of information assets and its security

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **COs** | | **Information Security and Cryptography** | | | | | | | | | | | | | |
| **CO1** | | Discuss the basics of network security and cryptography. | | | | | | | | | | | | | |
| **CO2** | | Explain the various standards Symmetric Encryption algorithms used to provide confidentiality. | | | | | | | | | | | | | |
| **CO3** | | Explain the various standards Asymmetric Encryption algorithms to achieve authentication. | | | | | | | | | | | | | |
| **CO4** | | Demonstrate encryption techniques to secure data in transit across network. | | | | | | | | | | | | | |
| **CO5** | | Explore the knowledge of key exchange protocols. | | | | | | | | | | | | | |
| **CO6** | | Examine the effects on digitized security. | | | | | | | | | | | | | |
| **COs** | **PO1** | | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| **CO1** | 1 | | – | – | 2 | – | – | – | – | – | – | – | – | – | – |
| **CO2** | 1 | | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | 2 |
| **CO3** | 1 | | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | 2 |
| **CO4** | 2 | | 2 | 1 | 1 | – | – | – | – | – | – | – | – | – | 2 |
| **CO5** | 2 | | 2 | 2 | 1 | – | – | – | – | – | – | – | – | – | 3 |
| **CO6** | 3 | | 3 | 3 | 2 | – | – | – | – | – | – | – | – | – | 3 |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST354** | **Introduction to Information Security** | **L** | **T** | **P** | **C** |
| Total Contact Hours : 45Hours | **3** | **0** | **0** | **3** |
| Common to all Specializations of CSE is 4thYear |
| Prerequisite: Studied computer networks | | | | |

Course Objectives

1. To familiarize the students with the basic concepts of services, attacks with its models and concepts of encryption.
2. To conceptualize digital signature and different encryption algorithm. To state the various authentication protocols and their requirements.
3. To elucidate an applications of security and their effects on security standards.
4. To comprehend IP security and their methods.
5. To familiarize the student this basic encryption and decryptions

Course Outcomes

|  |  |
| --- | --- |
| **CO1** | Discuss the basics of network security and cryptography. |
| **CO2** | Explain the various standards Symmetric Encryption algorithms used to provide confidentiality. |
| **CO3** | Explain the various standards Asymmetric Encryption algorithms to achieve authentication. |
| **CO4** | Demonstrate encryption techniques to secure data in transit across network. |
| **CO5** | Explore the knowledge of key exchange protocols. |
| **CO6** | Examine the effects on digitized security. |

### UNIT-I

**Introduction & Number Theory:** Services, Mechanisms and attacks-the OSI security architecture-Network security model-Classical Encryption techniques (Symmetric cipher model, substitution techniques, transposition techniques, steganography).FINITE FIELDS AND NUMBER THEORY: Groups, Rings, Fields-Modular arithmetic-Euclid‟s algorithm-Finite fields- Polynomial Arithmetic –Prime numbers- Fermat‟s and Euler‟s theorem-Testing for primality -The Chinese remainder theorem- Discrete alogarithms.

**Block Ciphers :**Data Encryption Standard-Block cipher principles-block cipher modes of operation- Advanced Encryption Standard (AES)-Triple DES-Blowfish-RC5 algorithm.

**UNIT II**

**Public key cryptography:** Principles of public key cryptosystems-The RSA algorithm-Key management – Diffie Hellman Key exchange

**Hash Functions And Digital Signatures:** Authentication requirement – Authentication function – MAC – Hash function – Security of hash function and MAC – MD5–SHA512–HMAC – CMAC – Digital signature and authentication protocols – DSS – EI Gamal – Schnorr.

**UNIT III**

**Security Practice & System Security:** Authentication applications – Kerberos – Authentication services – Internet Firewalls for Trusted System: Roles of Firewalls – Firewall related terminology- Types of Firewalls – Firewall designs –Intruder – Intrusion detection system – Virus and related threats.

**E-mail Security:** Security Services for E-mail-attacks possible through E-mail – establishing keys privacy- authentication of the source-Message Integrity-Non-repudiation-Pretty Good Privacy-S/MIME.

**IPSecurity and Web Security:** Overview of IPsec – IP and IPv6-Authentication Header-Encapsulation Security Payload (ESP)- SSL Architecture and its layers.

### Text Books:

1. William Stallings, Cryptography and Network Security, 6th Edition, Pearson Education, March 2013.
2. Charlie Kaufman, Radia Perlman and Mike Speciner, “Network Security*”*, Prentice Hall of India, 2002.

### Reference Books:

1. Behrouz A. Ferouzan, “Cryptography & Network Security”, Tata McGrawHill.
2. Man Young Rhee, “Internet Security: Cryptographic Principles”, “Algorithms and Protocols”, WileyPublications.
3. Charles Pfleeger, “Security in Computing”, 4th Edition, Prentice Hall ofIndia.
4. Ulysess Black, “Internet Security Protocols”, Pearson EducationAsia.
5. Charlie Kaufman and Radia Perlman, Mike Speciner, “Network Security, Second Edition, Private Communication in Public World”,PHI.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | To familiarize the students with the basic concepts of services, attacks with its models and concepts of encryption. | Discuss the basics of network security and cryptography. |
| 2 | **To conceptualize digital signature and different encryption algorithm.** | Explain the various standards Symmetric Encryption algorithms used to provide confidentiality. |
| 3 | To elucidate an applications of security and their effects on security standards. | Explain the various standards Asymmetric Encryption algorithms to achieve authentication. |
| 4 | To comprehend IP security and their methods. | Demonstrate encryption techniques to secure data in transit across network. |
| 5 | To familiarize the student this basic encryption and decryptions | Explore the knowledge of key exchange protocols. |
|  |  | Examine the effects on digitized security. |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Mapping Between COs and POs** | | | | | | | | | | | | | |
| **COs** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | | **PSO2** |
| **CO1** | 1 | – | – | 2 | – | – | – | – | – | – | – | – | – | | – |
| **CO2** | 1 | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | | 2 |
| **CO3** | 1 | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | | 2 |
| **CO4** | 2 | 2 | 1 | 1 | – | – | – | – | – | – | – | – | – | | 2 |
| **CO5** | 2 | 2 | 2 | 1 | – | – | – | – | – | – | – | – | – | | 3 |
| **CO6** | 3 | 3 | 3 | 2 | – | – | – | – | – | – | – | – | – | | 3 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST354** | **Introduction to Information Security** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CSP354** | **Information Security Lab** | **L** | **T** | **P** | **C** |
| Total Contact Hours : 10 Hours | **0** | **0** | **2** | **0** |
| CSE 3rd Year |
| Prerequisite: Studied Encryption Techniques | | | | |

**Course Objectives**

1.To generate encrypted text using different techniques.

2.To understand the cipher-text in order to provide more security to data.

3. To understand the concepts of Digital Signature and its implementation.

4. Design and develop a security architecture for an organization.

5 Design operational and strategic cyber security strategies and policies.

Course Outcomes

1. Analyze and evaluate the cyber security needs of an organization.

2 . Determine and analyze software vulnerabilities and security solutions to reduce the risk of exploitation.

3. Measure the performance and troubleshoot cyber security systems.

4. Implement cyber security solutions and use of cyber security, information assurance, and cyber/computer forensics software/tools.

5. Comprehend and execute risk management processes, risk treatment methods, and key risk and performance indicators

**List of Experiments**

**UNIT-I**

1. Perform the practical of CaesarCipher.
2. Perform the practical of Substitutioncipher.
3. Perform the practical of Hill Cipher.
4. Perform the practical of Playfairalgorithm

### UNIT-II

1. Perform the practical of Diffie hellman key exchangealgorithm.
2. Perform the practical of RSAalgorithm.
3. Perform the practical ofMD5

### UNIT-III

1. Perform the practical ofSHA1.
2. Perform the practical to generate digital certificate.
3. Perform the practical of digital signature ontext.

CO1:Understand the conceptual foundation of information security awareness.

CO2:Analysis the risk events, treatment plans, assessment

CO3:Detail evaluation of information classification, roles and responsibilities

CO4:Examining the access controls, monitoring, management and review process

CO5:Study the physical and logical perimeters of information assets and its security

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **COs** | | **Information Security and Cryptography** | | | | | | | | | | | | | |
| **CO1** | | Discuss the basics of network security and cryptography. | | | | | | | | | | | | | |
| **CO2** | | Explain the various standards Symmetric Encryption algorithms used to provide confidentiality. | | | | | | | | | | | | | |
| **CO3** | | Explain the various standards Asymmetric Encryption algorithms to achieve authentication. | | | | | | | | | | | | | |
| **CO4** | | Demonstrate encryption techniques to secure data in transit across network. | | | | | | | | | | | | | |
| **CO5** | | Explore the knowledge of key exchange protocols. | | | | | | | | | | | | | |
| **CO6** | | Examine the effects on digitized security. | | | | | | | | | | | | | |
| **COs** | **PO1** | | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| **CO1** | 1 | | – | – | 2 | – | – | – | – | – | – | – | – | – | – |
| **CO2** | 1 | | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | 2 |
| **CO3** | 1 | | 3 | 2 | 1 | – | – | – | – | – | – | – | – | – | 2 |
| **CO4** | 2 | | 2 | 1 | 1 | – | – | – | – | – | – | – | – | – | 2 |
| **CO5** | 2 | | 2 | 2 | 1 | – | – | – | – | – | – | – | – | – | 3 |
| **CO6** | 3 | | 3 | 3 | 2 | – | – | – | – | – | – | – | – | – | 3 |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code** | **TITLE OF THE COURSE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
|  | **Competitive Coding – II** | **0** | **0** | **2** | **0** | **1** | **2** | **EE** |
| **20CSP351** | |  | | | | **Course Code(s)**  **20CSP-351** | | | |
| **PRE-REQUISITE** | | **21CSP-314** | | | |  | | | |
| **CO-REQUISITE** | | **21CST-352,21CST-353,21CST-354,21CST-355,21CST-357,21CST-371,21CSP-356** | | | |  | | | |
| **ANTI-REQUISITE** | | **21CST-475** | | | |  | | | |

**a. Course Description**

Competitive programming is the course in which students will learn how to apply algorithms in order to solve complex problems. The goal of this course is to teach students how to apply familiar algorithms to non-intuitive problems.

**b. Course Objectives**

Competitive programming is the course in which students will learn how to apply algorithms in order to solve complex problems.

The goal of this course is to teach students how to apply familiar algorithms to non-intuitive problems.

**c. Course Outcomes**

|  |  |
| --- | --- |
| CO1 | Describe how algorithmic problems are solved |
| CO2 | Recognize the time and memory complexity of an algorithm or a structure |
| CO3 | Explain the concrete algorithms and data structures |
| CO4 | Analyze the given problem and recognize sub-problems |
| CO5 | Apply the knowledge on a wider set of problems to achieve better time complexity. |

**d. Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **Data Structures** | **Contact Hours:** |
| **Arrays,**  **Stacks, Queues and linked list** | <https://leetcode.com/problems/3sum/>  <https://leetcode.com/problems/jump-game-ii/>  <https://leetcode.com/problems/valid-parentheses/>  <https://leetcode.com/problems/simplify-path/>  <https://leetcode.com/problems/implement-queue-using-stacks/>  <https://leetcode.com/problems/queue-reconstruction-by-height/>  <https://leetcode.com/problems/merge-two-sorted-lists/>  <https://leetcode.com/problems/remove-duplicates-from-sorted-list-ii/>  <https://leetcode.com/problems/reorder-list/> | |
| **Divide and conquer** | <https://leetcode.com/problems/count-and-say/>  <https://leetcode.com/problems/1-bit-and-2-bit-characters/>  <https://leetcode.com/problems/jewels-and-stones/>  <https://leetcode.com/problems/snakes-and-ladders/>  <https://leetcode.com/problems/water-and-jug-problem/>  https://leetcode.com/problems/find-and-replace-in-string/ | |
| **Heap** | <https://leetcode.com/problems/kth-largest-element-in-a-stream/>  <https://leetcode.com/problems/last-stone-weight/>  <https://leetcode.com/problems/race-car/>  <https://leetcode.com/problems/cheapest-flights-within-k-stops/>  <https://leetcode.com/problems/k-closest-points-to-origin/>  <https://leetcode.com/problems/network-delay-time/>  <https://leetcode.com/problems/distant-barcodes/>  <https://leetcode.com/problems/furthest-building-you-can-reach/>  https://leetcode.com/problems/swim-in-rising-water/ | |
| **Unit-2** | **(Name of the Unit)** | **Contact Hours:** |
| **Greedy** | <https://leetcode.com/problems/candy/>  <https://leetcode.com/problems/best-time-to-buy-and-sell-stock-ii/>  <https://leetcode.com/problems/remove-duplicate-letters/>  <https://leetcode.com/problems/can-place-flowers/>  <https://leetcode.com/problems/assign-cookies/>  <https://leetcode.com/problems/best-time-to-buy-and-sell-stock-with-transaction-fee/>  <https://leetcode.com/problems/lemonade-change/>  <https://leetcode.com/problems/boats-to-save-people/>  <https://leetcode.com/problems/minimum-add-to-make-parentheses-valid/>  https://leetcode.com/problems/three-equal-parts/ | |
| **Trees** | <https://leetcode.com/problems/binary-tree-inorder-traversal/>  <https://leetcode.com/problems/same-tree/>  <https://leetcode.com/problems/symmetric-tree/>  <https://leetcode.com/problems/balanced-binary-tree/>  <https://leetcode.com/problems/path-sum/>  <https://leetcode.com/problems/count-complete-tree-nodes/>  <https://leetcode.com/problems/sum-of-left-leaves/>  <https://leetcode.com/problems/delete-node-in-a-bst/>  <https://leetcode.com/problems/diameter-of-binary-tree/>  https://leetcode.com/problems/binary-tree-tilt/ | |
| **Backtracking** | <https://leetcode.com/problems/binary-watch/>  <https://leetcode.com/problems/stickers-to-spell-word/>  <https://leetcode.com/problems/all-paths-from-source-to-target/>  <https://leetcode.com/problems/word-ladder-ii/>  <https://leetcode.com/problems/subsets/>  <https://leetcode.com/problems/combinations/>  <https://leetcode.com/problems/palindrome-partitioning/> | |
| **Graph** | <https://leetcode.com/problems/is-graph-bipartite/>  <https://leetcode.com/problems/gray-code/>  <https://leetcode.com/problems/k-th-symbol-in-grammar/>  <https://leetcode.com/problems/group-the-people-given-the-group-size-they-belong-to/>  <https://leetcode.com/problems/the-skyline-problem/>  <https://leetcode.com/problems/find-the-difference/>  <https://leetcode.com/problems/predict-the-winner/>  https://leetcode.com/problems/construct-the-rectangle/ | |
| **Dynamic Programming** | <https://leetcode.com/problems/best-time-to-buy-and-sell-stock/>  <https://leetcode.com/problems/decode-ways/>  <https://leetcode.com/problems/scramble-string/>  <https://leetcode.com/problems/climbing-stairs/>  <https://leetcode.com/problems/unique-paths/>  <https://leetcode.com/problems/maximum-subarray/>  <https://leetcode.com/problems/longest-palindromic-substring/>  <https://leetcode.com/problems/house-robber-ii/>  <https://leetcode.com/problems/range-sum-query-immutable/>  https://leetcode.com/problems/word-break/ | |

**e. Assessment Pattern - Internal and External**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 60 | 40 |
| **Total Marks** | 100 | |

**f. Internal Evaluation Component**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sr. No. | Type of Assessment | Weightage of actual conduct | Frequency of Task | Final Weightage in Internal  Assessment | Remarks |
| 1 | Conduct | 10 Marks per Practical | 1 per practical | 60 Marks per course |  |
| 2 | Report | 10 Marks per Practical | 1 per practical |  |
| 3 | Viva- Voce | 20 Marks per Course | 1 per Course |  |

**g. CO-PO Mapping**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO2 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO3 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO4 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |
| CO5 |  | 3 | 3 | 3 |  |  |  |  | 1 | 2 |  | 2 | 3 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST357/20CSP358** | **Internet of Things** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours: 60 Hours | **3** | **0** | **0** | **0** | **3** |
| **Pre-requisites:** | 20CST355 | | | | | |
| **Co-requisites** | 20CST352, 20CST353, 20CST354, 20CST355, 20CST357, 20CST332, 20CST333, 20CST334 | | | | | |
| **Anti-Requisites** | 20CST331 | | | | | |

**Course Objectives**

* To study Origins, Drivers and Applications of Internet of Things.
* To study Internet of Things Communications Models.
* To learn what issues are raised by the Internet of Things.

**Course Outcomes**

* Analyze the basic terminologies associated with IOT and use it.
* Justify the applications of internet of things and correlate them.
* Compare different objects and communication strategies and also able to see the issues raised by communication strategies in IOT.
* Examine the protocols required for communication and packet size required for each application.
* Illustrate security issues with IOT like security, privacy, communication standard and some other legal issues.

**Contents of the Syllabus**

**UNIT-I [15h]**

**Chapter-1 (Introduction)**

What is the Internet of Things, Components of IOT, Applications, Different Definitions, Similar Concepts, Sensing, Actuation, Smart Objects, Smart applications.

**Chapter-2 (IOT Applications for Industry)**

Value Creation and Challenges. IoT Today, IoT as a Network of Networks, Why Is IoT Important, IoT: Critical for Human Progression, Challenges and Barriers to IoT.

**UNIT-II [15h]**

**Chapter-3 (Internet of Things Communication Models)**

Device – to – Device Communications, Device – to – Cloud Communications, Device – to - Gateway Model, Back - End Data - Sharing Model.

**Chapter-4 (Issues Raised by Internet of Things)**

Security Issues, The IoT Security Challenge, A Spectrum of Security Considerations, Unique Security Challenges of IoT Devices and Privacy.

Considerations: Internet of Things Privacy Background, Unique Privacy Aspects of Internet of Things, Interoperability.

**UNIT-III [15h]**

**Chapter-5 (Standard Issues)**

IoT Interoperability / Standards Background, Key Considerations and Challenges in IoT Interoperability / Standards, Regulatory, Legal, and Rights Issues: Data Protection and Cross border Data Flows, IoT Data Discrimination, IoT Devices as Aids to Law Enforcement and Public Safety, IoT Device Liability.

**Chapter-6 (Proliferation of IOT Devices)**

Used in Legal Actions, Regulatory, Legal, and Rights Issues Summary, Emerging Economy and Development Issues: Ensuring IoT Opportunities are Global, Economic and Development Opportunities.

**Chapter-7 (Case Study)**

Case study on smart homes using internet of things.

**ADVANCED TOPICS (BEYOND SYLLABUS)**

Mobile Cloud Services, IOT and Cloud Security, Smart Cloud and IOT

**TEXT BOOKS**

1. Ovidiu Vermesan, Peter Friess, “Internet of Things-Converging Technologies for Smart Environments & integrated Ecosystem”, River Publications Netherlands.
2. Internet Society, “An overview of Internet of Things”.

**REFERENCE BOOKS**

1. Pfister, Cuno, “Getting started with the Internet of Things: connecting sensors and microcontrollers to the cloud", O'Reilly Media, Inc.”, 2011.
2. Greenfield, Adam,” Everyware: The dawning age of ubiquitous computing”, New Riders, 2010.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 |  |
| 2 | CO2 |  |
| 3 | CO3 |  |
| 4 | CO4 |  |
| 5 | CO5 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST357/**  **20CSP358** | **Internet of Things** |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST357/20CSP358** | **Internet of Things** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours: 60 Hours | **0** | **0** | **4** | **0** | **2** |
| **Pre-requisites:** | 20CST355 | | | | | |
| **Co-requisites:** | 20CST352, 20CST353, 20CST354, 20CST355, 20CST357, 20CST332, 20CST333, 20CST334 | | | | | |
| **Anti-Requisites:** | 20CST331 | | | | | |

**Course Objectives**

* To study Origins, Drivers and Applications of Internet of Things.
* To study Internet of Things Communications Models.
* To learn what issues are raised by the Internet of Things.

**Course Outcomes**

* Analyze the basic terminologies associated with IOT and use it.
* Justify the applications of internet of things and correlate them.
* Compare different objects and communication strategies and also able to see the issues raised by communication strategies in IOT.
* Examine the protocols required for communication and packet size required for each application.
* Illustrate security issues with IOT like security, privacy, communication standard and some other legal issues.

**List of Experiments**

|  |
| --- |
| **UNIT-I** |
| **Experiment-1**  Demonstration of   1. Arduino-MATLAB Interface 2. Long-Distance Serial Link Between Two Arduino Devices |
| **Experiment-2**  Study of basic script - based programming in MATLAB. |
| **Experiment-3**  To study of Simulink and different toolboxes in MATLAB. |
| **Experiment-4**  Explain the working of IoT on the MATLAB Platform. |

|  |
| --- |
| **UNIT-II** |
| **Experiment-5**  To deploy LIFA (LabVIEW interface for Arduino) and to study the response using LabVIEW and Arduino. |
| **Experiment-6**  To measure the distance of an object using SONAR principle by ultrasonic proximity sensor. |
| **Experiment-7**  To study the operation of digital humidity sensor and calculate the accuracy of the device. |

|  |
| --- |
| **UNIT-III** |
| **Experiment-8**  To study the following VI loops   1. Do While loop 2. For loop 3. Case Structure |
| **Experiment-9**  To study the sampling and quantization of analog sensor outputs. |
| **Experiment-10**  To understand servo control system. |

**Text Book/Reference Book**

**TEXT BOOKS**

1. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data. John Wiley & Sons, 2015.
2. Jaiwei Han, Micheline Kamber, “Data Mining Concepts and Techniques”, Elsevier, 2006.
3. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.

**REFERENCE BOOKS**

1. Cathy O’Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline. O’Reilly. 2014.
2. Introduction to Data Mining by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, 2005.
3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann, 2005.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST334/20CSP339** | **MULTIMEDIA TECHNOLOGIES** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites:** |  | | | | | |
| **Co-requisites** |  | | | | | |
| **Anti-Requisites** |  | | | | | |

**Course Objectives:**

* The aim of the syllabus is to provide orientation as regard to uses of Multimedia.
* This course will explain the technologies underlying digital images, videos and audio contents, including various compression techniques and standards, and the issues to deliver multimedia content over the Internet.

**Course Outcomes: (5 Required)**

* To identify the essential features of graphics/image data types, file formats, and colour models in images and video.
* To explain the technical details of multimedia data representations.
* To perform a comparative analysis of the major methods and algorithms for multimedia data compression.
* To explain the technical details of popular multimedia compression standards.
* To perform different operations on videos.

**Contents of the Syllabus:**

**UNIT-I [15h]**

**Chapter-1(Introduction)**

Contents: What is multimedia, Components of multimedia, Web and Internet multimedia applications, Transition from conventional media to digital media.

**Chapter-2(**Computer Fonts and Hypertext**)**

# Contents: Usage of text in Multimedia, Families and faces of fonts, outline fonts, bitmap fonts

International character sets and hypertext, Digital fonts techniques.

**UNIT-II [15h]**

**Chapter -3(**Audio fundamentals and representations**)**

Digitization of sound, frequency and bandwidth, decibel system, data rate, audio file format, Sound synthesis, MIDI, wavetable, Compression and transmission of audio on Internet, Adding sound to your multimedia project, Audio software and hardware.

**Chapter -4(**Image fundamentals and representations**)**

Contents: ColourScience ,Colour, Colour Models, Colour palettes, Dithering, 2D Graphics, ImageCompression and File Formats :GIF, JPEG, JPEG 2000, PNG, TIFF, EXIF, PS, PDF, BasicImage Processing [ Can Use Photoshop ], Use of image editing software, White balancecorrection, Dynamic range correction, Gamma correction, Photo Retouching.

**UNIT-III [15h]**

**Chapter -5 (**Video and Animation**)**

# Video Basics , How Video Works, Broadcast Video Standards, Analog video, Digital video,Video Recording and Tape formats, Shooting and Editing Video (Use Adobe Premier forediting), Video Compression and File Formats. Video compression based on motioncompensation, MPEG-1, MPEG-2, MPEG-4, MPEG-7, MPEG-21, Animation: Cell.

Animation, Computer Animation, Morphing.

**Chapter -6 (**Multimedia Authoring**)**

Multimedia Authoring Basics, Some Authoring Tools, Macromedia Director & Flash.

**TEXT BOOKS**

1. Tay Vaughan, “Multimedia making it work”, Tata McGraw-Hill, 2008.
2. Rajneesh Aggarwal & B. B Tiwari, “ Multimedia Systems”, Excel Publication, New Delhi, 2007.
3. Li & Drew, “ Fundamentals of Multimedia” , Pearson Education, 2009..

**REFERENCE BOOKS**

1. Parekh Ranjan, “Principles of Multimedia”, Tata McGraw-Hill, 2007
2. AnirbanMukhopadhyay and Arup Chattopadhyay, “Introduction to Computer Graphics and Multimedia”, Second Edition, Vikas Publishing House.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-211** | **Data Structures** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST352** | **Network Operating System** | **L** | **T** | **P** | **S** | **C** |
| Total Contact Hours: 45 Hours | **3** | **0** | **0** | **0** | **3** |
| **Pre-requisites:** | 20CST355 | | | | | |
| **Co-requisites:** | 20CST352, 20CST353, 20CST354, 20CST355, 20CST357, 20CST332, 20CST333, 20CST334 | | | | | |
| **Anti-Requisites:** | 20CST331 | | | | | |

**Course Objectives**

* To become familiar with the architecture of RPC, DNS and DHCP.
* To learn the basic features and functionalities of network monitoring tools, window server 2008 and active directories.

**Course Outcomes**

* Explain and recall the TCP/IP and OSI reference models, the architecture of DNS, DHCP and RPC.
* Analyse and apply the configurations of Network Monitoring Tool.
* Recognize the basic structure of operating systems and classify roles and responsibilities of an operating System.
* Investigate and demonstrate the features of Windows server 2008.
* Examine and illustrate the various features of Active directories.

**Contents of the Syllabus**

**UNIT-I [15h]**

**Chapter-1 (Introduction to Computer Networks)**

Introduction of Computer Networks: Reference Model: OSI Reference Model, TCP/IP reference model, Core protocols of TCP/IP, IP routing, IP routers, Routing Tables.

Introduction to DNS: DNS Architecture, Understanding the DNS Domain Namespace, DNS domain name hierarchy, DNS and Internet domains, DNS delegation.

Introduction to DHCP: Introduction to DHCP, DHCP Architecture, Benefits to DHCP, DHCP Terminologies and DHCP Server Responsibility.

Introduction to RPC: Introduction to RPC, RPC Dependencies and Interaction, RPC Architecture, RPC Component and RPC Processes and Interaction.

Introduction to Microsoft Network Monitoring Tool: Introduction to Netmon tool, Installation and configuration of Netmon Tool.

**UNIT-II [15h]**

**Chapter-2 (Introduction to OS and Window Servers)**

Introduction to operating System: Introduction to operating system, Types of operating System: Mainframe, Desktop, Multiprocessor, Distributed, Clustered, Multiprogramming, Real time, Embedded and Time Sharing.

Operating System Components: Process Management Component, Memory Management component, I/O Management component, File Management component, Protection System, and Networking management component. Comparison of Microsoft OS (Desktop, Server and Client), Hardware Requirement for OS (Desktop, Server and client), Workgroups and Domains.

Installing windows server 2008: Plan for a server role and installing windows server core, configure server core, Add and configure server roles, add backup feature and migrate roles from previous versions of windows server.

Configuring Windows Server 2008: Windows server registry, Control Panel, Delegate administration, Add and removes feature in window server, initial configuration tasks, server manager console, server manager wizards and windows power shell.

**UNIT-III [15h]**

**Chapter-3 (Active Directory)**

Hyper-V: Introducing Hyper-V, Virtual Machines.

Introducing to Active Directory: role of an AD DS Server, features in AD DS, Common Terminologies and Active Directory Concepts, Active Directory Schema, Active Directory Objects, Active Directory Concepts, Active Directory Data structure and storage architecture, Active Directory Structure and storage components, DNS support for active directory, active directory DNS support components.

Install Active Directory Domains Services in Windows 2008 R2: Installing a new forest by using the graphical user interface (GUI), Understanding active directory domain. Servicex functional Levels.

Active Directory Administration: Active Directory Users and Computers. Managing organizational units using Active Directory Users and Computers, Managing Trusts using Active directory domains and trusts and managing forest trusts using active directory domains and trusts snap-in.

**ADVANCED TOPICS (BEYOND SYLLABUS)**

Computer Networking, Network Security

**TEXT BOOKS**

1. “Data communication & Networking”, Frozen Tata McGraw Hill Publication.
2. “Data & computer communications”, Stalling PHI New Delhi.
3. Operating Systems Design and Implementation (Prentice Hall Software Series), Andrew S Tanenbaum, Albert S Woodhull.

**REFERENCE BOOKS**

1. Capture Network traffic---http://support.microsoft.com/kb/812953
2. About Network Monitor-[--http://support.microsoft.com/kb/294818](http://support.microsoft.com/kb/294818)
3. Traces--<http://support.microsoft.com/kb/169292>

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 |  |
| 2 | CO2 |  |
| 3 | CO3 |  |
| 4 | CO4 |  |
| 5 | CO5 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST352** | **Network Operating System** |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **CSY-360** | **Software Project Management** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites:** | 21CST-254 | | | | | |
| **Co-requisites** | **21CST-352,21CST-353,21CST-354,21CST-355,21CST-357,21CST-371,21CSP-356** | | | | | |
| **Anti-Requisites** | 21CST-463 | | | | | |

**Course Objectives:**

* Introduces the concepts and methods required for the construction of large software systems.
* Gain knowledge on the principles and techniques of software project management
* Provides an understanding of both theoretical and methodological issues involved in software engineering project management.

**Course Outcomes: (5 Required)**

* To implement modeling, communication, construction and deployment practices in software development.
* To estimate project costs and perform cost-benefit evaluation among projects.
* To apply schedule and cost control techniques for project monitoring.
* Students will learn how to choose estimation techniques from different methods according to project scope, size, and nature.
* Students will know the new techniques of software project management and their benefits.

**Contents of the Syllabus:**

**UNIT-I [15h]**

**Chapter-1(SOFTWARE PROJECT EVALUATION AND PROJECT PLANNING)**

Importance of Software Project Management – Activities Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation – Strategic program Management – Stepwise Project Planning-Software process and Process Models – Choice of Process models - mental delivery – Rapid Application development – Agile methods – Extreme Programming – SCRUM – Managing interactive processes

**UNIT-II [15h]**

**Chapter -2(**SOFTWARE PROJECT LIFE CYCLE AND EFFORT ESTIMATION**)**

Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points - COCOMO II A Parametric Productivity Model - Staffing Pattern- Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Forward Pass & Backward Pass techniques – Critical path (CRM) method– Risk identification – Assessment – Monitoring – PERT technique – Monte Carlo simulation –Resource Allocation – Creation of critical patterns – Cost schedules.

**UNIT-III [15h]**

**Chapter -3 (**SOFTWARE PROJECT MANAGEMENT AND CONTROL**)**

Framework for Management and control – Collection of data Project termination – Visualizing progress – Cost monitoring – Earned Value Analysis- Project tracking – Change control-Software Configuration Management – Managing contracts – Contract Management - Staffing -Managing people – Organizational behavior – Best methods of staff selection – Motivation – The Oldham-Hackman job characteristic model – Ethical and Programmed concerns – Working in teams – Decision making – Team structures – Virtual teams – Communications genres – Communication plans.

**TEXT BOOKS**

1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Fifth Edition, Tata McGraw Hill, New Delhi, 2012.

**REFERENCE BOOKS**

1. Robert K. Wysocki “Effective Software Project Management” – Wiley Publication,2011.
2. Walker Royce: “Software Project Management”- Addison-Wesley, 1998.
3. Gopalaswamy Ramesh, “Managing Global Software Projects” – McGraw Hill Education (India), Fourteenth Reprint 2013.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **CSY-360** | **Software Project Management** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code** | **TITLE OF THE COURSE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type** |
| 5 | **Machine learning** | 3 | 0 | 2 | 0 | 4 | 5 | CR |
| **20CSP-317** | |  | | | | Course Code(s)  **20CSP-317** | | | |
| **PRE-REQUISITE** | |  | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

**Course Objectives**

* To formulate machine learning problems corresponding to different applications.
* To understand a range of machine learning algorithms along with their strengths and weaknesses.
* To understand the basic theory underlying machine learning.
* To apply machine learning algorithms to solve problems of moderate complexity

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | **Implement** winner take it all and delta learning law using Python |
| CO2 | **Analyze**the implementation results of various perceptron learning laws. |
| CO3 | **Apply** multilayer perceptron. |
| CO4 | **Develop** and use C4.5 decision tree. |
| CO5 | **Design**& demonstrate K means Clustering. |

**Syllabus**

**List of Experiments**

**UNIT-I**

1. Program to implement perceptron learning law
2. Program to implement delta learning law
3. Program to implement winner take it all learning law
4. Program to implement single layer perceptron

**UNIT-II**

1. Program to implement multilayer perceptron
2. Program to solve XOR problem using multilayer perceptron
3. Program to implement C4.5 decision tree

**UNIT-III**

1. Program to implement K-means clustering algorithm
2. Program to cluster some sample data set into disjoint clusters using K-means
3. Program to for Text Classification for the give sentence.

**Text Books:**

1. Rich E., *Artificial Intelligence*, Tata McGrawHills.
2. George F. Luger, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, Pearson Education Asia.
3. EthemAlpaydın, Introduction to Machine Learning (Adaptive Computation and Machine Learning), MIT Press, 2004.
4. Mitchell. T, Machine Learning, McGraw Hill, 1997.

**Reference Material:**

1. D.W. Patterson, *Introduction to AI and Expert Systems*, PHI.
2. N.J. Nilsson, *Principles of Artificial Intelligence*, Kaufmann,1980
3. Saroj Kaushik, *Logic and Prolog Programming*, NewAgeInternational Publications.
4. PH.Winston, *Artificial Intelligence*, Addison Wesley.
5. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
6. Ryszard S.  Michalski,  Jaime G.  Carbonell,  Tom M.  Mitchell,  Machine Learning  : An Artificial Intelligence Approach, Tioga Publishing Company, 1983.

**Assessment Pattern- internal and External**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Practical** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **External Viva** |
| **Marks** | 60 | 40 |
| **Total Marks** | 100 | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 | PO3,5 |
| 2 | CO2 | Po4 |
| 3 | CO3 | PO2 |
| 4 | CO4 | PO3 |
| 5 | CO5 | PO1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CSP-317** | **Machine Learning Lab** | **CO5** | **Co3** | **CO1,4** | **CO2** | **CO1** |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | **Course Code** | **TITLE OF THE COURSE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type** |
| 5 | **Machine learning** | 3 | 0 | 2 | 0 | 4 | 5 | CR |
| **20CST-316** | |  | | | | Course Code(s)  **20CST-316** | | | |
| **PRE-REQUISITE** | |  | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

**Course Objectives**

* To formulate machine learning problems corresponding to different applications.
* To understand a range of machine learning algorithms along with their strengths and weaknesses.
* To understand the basic theory underlying machine learning.
* To apply machine learning algorithms to solve problems of moderate complexity.

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | **Analyze** the preliminaries of Machine learning |
| CO2 | **Illustrate** different techniques like task-oriented studies, cognitive studies |
| CO3 | **Evaluate** various ML Algorithms |
| CO4 | **Choose & Apply** appropriate machine learning algorithms to solve various problems |
| CO5 | **Justify** contemporary techniques in machine learning |

**Syllabus**

**Preliminaries** - Introduction - Learning Input-Output Functions - Learning and Bias - Sample applications - Boolean Functions - Representation - Classes of Boolean Functions - Introduction to Neural Networks.

**Using Version Spaces for Learning** - Version Spaces and Mistake Bounds - Version Graphs -Learning as Search of a Version Space - The Candidate Elimination Method – Neural Networks - Threshold Logic Units - Linear Machines - Networks of TLUs – Training Feed forward Networks by Back propagation - Synergies Between Neural Network and Knowledge-Based Methods - Statistical Learning - Using Statistical Decision Theory -Learning Belief Networks - Nearest-Neighbor Methods.

**UNIT-II [15h]**

**Decision Trees** - Definitions - Supervised Learning of Univariate Decision Trees – NetworksEquivalent to Decision Trees – Over fitting and Evaluation - The Problem of Replicated Subtrees - The problem of Missing Attributes - Comparisons - Inductive Logic Programming - Notations and Definitions - A Generic ILP Algorithm - Inducing Recursive Programs - Choosing Literals to Add - Relationship Between ILP and Decision Tree Induction - Computational Learning Theory - Notation and Assumptions for PAC Learning Theory - PAC Learning - The Vapnik-Chervonenkis Dimension - VC Dimension and PAC Learning.

**UNIT-III [16h]**

**Unsupervised Learning** - Clustering Methods - Hierarchical Clustering Methods - Temporal-Difference Learning - Temporal Patterns and Prediction Problems - Supervised and Temporal-Difference Methods - Incremental computation of the (delta w)i - An experiment with TD Methods - Theoretical Results - Intra-Sequence Weight Updating - Delayed. **Reinforcement Learning** - The General Problem - Temporal Discounting and Optimal Policies - Q-Learning - Discussion, Limitations, and Extensions of Q-Learning - Explanation-Based Learning - Deductive Learning - Domain Theories - Evaluable Predicates - More General Proofs - Utility of EBL – Applications.

**Text Books:**

1. EthemAlpaydın, Introduction to Machine Learning (Adaptive Computation and Machine Learning), MIT Press, 2004.
2. Mitchell. T, Machine Learning, McGraw Hill, 1997.

**Reference Material:**

1. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
2. Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell, Machine Learning : An Artificial Intelligence Approach, Tioga Publishing Company, 1983.

**Mode of Evaluation:**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 40 | 60 |
| **Total Marks** | 100 | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 | PO1 |
| 2 | CO2 | PO1 |
| 3 | CO3 | PO2 AND PO3 |
| 4 | CO4 | PO2 |
| 5 | CO5 | PO1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-316** | **Machine Learning** | **CO1,2,5** | **Co3,4** | **Co3** |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | 20CSP-336 | **TITLE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
| 1 | STATISTICAL INFERENCE USING R LAB | 0 | 2 | 0 | 0 | 2 | 2 | <UC/PE/PC/MNG> |
|  | |  | | | | Course Code(s) | | | |
| **PRE-REQUISITE** | | Basic Knowledge of Statistics | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

\*Course Type

University Core, Program Core, Program Specialization, Program Elective, University Open Elective, Domain Aptitude, MOOCs, Project/Research Project, Thesis/Dissertation, Seminar, Mandatory Non-Graded, Self-Study MNG, Summer/Institutional/Industrial Training

**Course Objectives**

* The course will enable students to learn a new programming paradigm.
* The course will help students to develop data analytic skills.
* The course will encourage students to develop predictive analysis.

**Course Outcomes**

|  |  |  |
| --- | --- | --- |
| CO1 | **Develop** the solution using R |  |
| CO2 | **Analyze** data using Statistical Analysis |  |
| CO3 | **Interpret and visualize** the Data Sets |  |
| CO4 | **Implement** predictive analysis on data |  |
| CO5 | **Apply**  statistical analysis on data |  |

**Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** |  | Contact Hours:15 |
| Ex:1 | Setting R in Local Environment: Installation from Comprehensive R Archieve Network(CRAN), R packages and libraries. | |
| Ex:2 | To print messages in R command prompt, also demonstrating the use of single line and multi line comments | |
| Ex:3 | To create variable using different data types. | |
| Ex:4 | Create a dataframe of 15 tuples and apply at least 10 manipulations. (use data.frame() function) | |
| **Unit-2** |  | Contact Hours:15 |
| Ex:5 | To implement statistical in built functions in R(at least 10). | |
| Ex:6 | To import data and manipulate it using different data files like csv,xlsx,txt. | |
| Ex:7 | To plot data using pie chart, bar chart, box plot, histogram, linegraph and scatterplot. | |
| **Unit-3** |  | Contact Hours:15 |
| Ex:8 | To Find central tendency and variance (using housing data) in R. | |
| Ex:9 | Import a data file and implement Regression Analysis using R. Also apply ANOVA. | |
| Ex:10 | Create a data of 50 numbers which are normally as well as binomially distributed. | |

**Textbooks / Reference Books**

* [Discovering Statistics Using R](https://geni.us/yAkp) by Hadley
* R for Data Science: Import, Tidy, Transform, Visualize, and Model Data by Andy field

**Assessment Pattern- internal and External**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Practical** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **External Viva** |
| **Marks** | 60 | 40 |
| **Total Marks** | 100 | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 | PO3,5 |
| 2 | CO2 | PO2,4 |
| 3 | CO3 | PO2 |
| 4 | CO4 | PO3 |
| 5 | CO5 | PO1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CSP-336** | **STATISTICAL Inference using R** | **CO5** | **CO2,3** | **CO1,4** | **CO2** | **CO1** |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

**Annexure -3**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | 20CSP-337 | **TITLE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
| 1 | SOFT COMPUTING LAB | 0 | 2 | 0 | 0 | 2 | 2 | <UC/PE/PC/MNG> |
|  | |  | | | | Course Code(s) | | | |
| **PRE-REQUISITE** | | Knowledge of basic computer science principles and skills, at a level sufficient to write a reasonably non-trivial computer program. | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

**Course Objectives**

* To formulate machine learning problems corresponding to different applications.
* To understand a range of machine learning algorithms along with their strengths and weaknesses.
* To understand the basic theory underlying machine learning.
* To apply machine learning algorithms to solve problems of moderate complexity.

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | **Develop** the solution using python and its latest libraries |
| CO2 | **Generalize** the solution for the problems through implementation. |
| CO3 | **Experiment** different concepts related to fuzzy systems |
| CO4 | **Analyse** between different fuzzy properties through its application |
| CO5 | I**mplemen**t different Artificial Neural Networks. |

1. **Syllabus**

|  |  |  |
| --- | --- | --- |
| Unit-1 |  | Contact Hours: |
| Ex:1 | To perform Union, Intersection and Complement operations. | |
| Ex:2 | To implement De-Morgan’s Law. | |
| Ex:3 | Generate ANDNOT function using McCulloch-Pitts neural net. | |
| Unit-2 |  | Contact Hours: |
| Ex:4 | Generate XOR function using McCulloch-Pitts neural net. | |
| Ex:5 | Hebb Net to classify two dimensional input patterns in bipolar with given targets. | |
| Ex:6 | Perceptron net for an AND function with bipolar inputs and targets. | |
| Ex:7 | To calculate the weights for given patterns using hetero-associative neural net. | |
| Unit-3 |  | |
| Ex:8 | To store vector in an auto-associative net. Find weight matrix & test the net with input | |
| Ex:9 | To store the vector, find the weight matrix with no self-connection. Test this using a discrete Hopfield net. | |
| Ex:10 | To plot various membership functions. | |

**Textbooks / Reference Books**

**Text Books**:

1. S.N. Shivanandam, Principle of soft computing, Wiley. ISBN13: 9788126527410 (2011)
2. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, “Neuro-Fuzzy and Soft Computing”, PrenticeHall of India, 2003.
3. George J. Klir and Bo Yuan, “Fuzzy Sets and Fuzzy Logic-Theory and Applications”, Prentice Hall, 1995.

**Reference Material**

1. James A. Freeman and David M. Skapura, “Neural Networks Algorithms, Applications, and Programming Techniques”, Pearson Edn., 2003.
2. Mitchell Melanie, “An Introduction to Genetic Algorithm”, Prentice Hall, 1998.
3. David E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison Wesley, 1997.

**Assessment Pattern- internal and External**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Practical** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **External Viva** |
| **Marks** | 60 | 40 |
| **Total Marks** | 100 | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 | PO3,5 |
| 2 | CO2 | Po4 |
| 3 | CO3 | PO2 |
| 4 | CO4 | PO3 |
| 5 | CO5 | PO1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CSP-337** | **Soft Computing Lab** | **CO5** | **Co3** | **CO1,4** | **CO2** | **CO1** |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SNo** | 20CST-331 | **TITLE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
|  | STATISTICAL INFERENCE USING R | 0 | 2 | 0 | 0 | 2 | 2 | <UC/PE/PC/MNG> |
|  | |  | | | | Course Code(s) | | | |
| **PRE-REQUISITE** | | Fundamental knowledge of other programming languages | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

\*Course Type

University Core, Program Core, Program Specialization, Program Elective, University Open Elective, Domain Aptitude, MOOCs, Project/Research Project, Thesis/Dissertation, Seminar, Mandatory Non-Graded, Self-Study MNG, Summer/Institutional/Industrial Training

**Course Objectives**

* The course provides an overview of modern statistical data analysis.
* The Course includes Programming with data and that will be an integral part of the course.
* The topic related to Inferential computation will include parameter estimation, hypothesis testing for proportions, means and medians, goodness of fit tests, and tests for independence.

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | Apply the basics of R programming |
| CO2 | Understand Advance R concepts |
| CO3 | To do predictive and statistical analysis on data |
| CO4 | Students will learn to create and present graphs |
| CO5 | Students will learn the scope of inferential conclusions for numerous scenarios (experiments, observational studies, etc.). |

**Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **Introduction to R** | Contact Hours:15 |
| Ch-1.1 | Introduction to R programming, evolution of R, features of R, Understanding the importance of R in context with inferential statistics, Comparative analysis of R and Python | |
| Ch-1.2 | Basic R syntax, Foundational R programming concepts: R data types, Manipulating R variables(variable assignment, searching and deleting) | |
| Ch-1.3 | R operators(arithmetic, relational, logical, assignment, miscellaneous operators),R Decision making statements, R-loops | |
| **Unit-2** | **Advancing R concepts** | Contact Hours:15 |
| Ch-2.1 | Creating and Using functions in R, Exploring various inbuilt statistical R functions | |
| Ch-2.2 | Manipulating R strings, arrays, vector, lists, data frames, Exploring various R data interfaces, data visualization using charts and graphs | |
| Ch-2.3 | Estimates and margins of errors of populations, parameters, estimates and standard errors in order to make predictions about data Basics of Bayesian statistics and predictive modelling | |
| **Unit-3** | **Statistical concepts using R** | Contact Hours:15 |
| Ch-3.1 | Regression analysis using R(linear regression, multiple regression, logistic regression) | |
| Ch-3.2 | Normal and Binomial distribution using R | |
| Ch-3.3 | ANOVA, Hypothesis Testing using R | |

**Textbooks / Reference Books**

* [Discovering Statistics Using R](https://geni.us/yAkp) by Hadley
* R for Data Science: Import, Tidy, Transform, Visualize, and Model Data by Andy field

**Mode of Evaluation:**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 40 | 60 |
| **Total Marks** | 100 | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 | PO1 |
| 2 | CO2 | PO1 |
| 3 | CO3 | PO2 AND PO3 |
| 4 | CO4 | PO2 |
| 5 | CO5 | PO1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-331** | **STATISTICAL Inference using R** | **CO1,2,5** | **Co3,4** | **Co3** |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SN** | 20CST-332 | **TITLE** | **L** | **T** | **P** | **S** | **C** | **CH** | **Course Type\*** |
| 1 | SOFT COMPUTING | 0 | 2 | 0 | 0 | 2 | 2 | <UC/PE/PC/MNG> |
|  | |  | | | | Course Code(s) | | | |
| **PRE-REQUISITE** | | Knowledge of basic concepts of Fuzzy logic and genetic algorithms | | | |  | | | |
| **CO-REQUISITE** | |  | | | |  | | | |
| **ANTI-REQUISITE** | |  | | | |  | | | |

\*Course Type

University Core, Program Core, Program Specialization, Program Elective, University Open Elective, Domain Aptitude, MOOCs, Project/Research Project, Thesis/Dissertation, Seminar, Mandatory Non-Graded, Self-Study MNG, Summer/Institutional/Industrial Training

**Course Objectives**

* To understand soft computing concepts , techniques and Fuzzy System
* To study the concepts of neural networks
* To understand GA and Particle Swarm Optimization Technique

**Course Outcomes**

|  |  |
| --- | --- |
| CO1 | **Classify** the fundamental terminologies and concepts of ANN. |
| CO2 | **Distinguish** between supervised and unsupervised learning network |
| CO3 | **Analyze** fuzzy logic along with applications. |
| CO4 | **Compare & Contrast** traditional algorithms with genetic algorithms and recall various concepts. |
| CO5 | **Understand** the concepts related to Genetic Algorithms |

**Syllabus**

|  |  |  |
| --- | --- | --- |
| **Unit-1** | **Introduction to Soft Computing** | **Contact Hours:15** |
| Ch-1.1 | Introduction to Soft Computing – Soft Computing vs. Hard Computing – Characteristics of soft Computing. | |
| Ch-1.2 | Introduction to Fuzzy Logic – Fuzzy Sets and Membership functions – Operations on Fuzzy sets – Fuzzy Relations, Rules, | |
| Ch-1.3 | Propositions Implications and Inferences – Defuzzification Techniques: Lambda-cut method, Weighted average method, Maxima method, Centroid method. | |
| **Unit-2** | **Artificial Neural Network and Associative Memory Networks** | **Contact Hours:15** |
| Ch-2.1 | Artificial Neural Network Introduction – Fundamental concept – Biological neuron and its working-Evolution of Neural Networks – Basic Models of Artificial Neural Networks – Important Terminologies of ANNs –Activation Functions | |
| Ch-2.2 | Architecture: Single layer feed forward architecture, Multilayer feed forward architecture – McCulloch-Pitts Neuron Supervised Learning vs. Unsupervised Learning–– Supervised Learning Networks: Back-Propagation Network – Unsupervised Learning Networks: Hebb Network. | |
| Ch-2.3 | Associative Memory Networks: Training Algorithms for Pattern Association – Autoassociative Memory Network – Heteroassociative Memory Network – Hopfield Networks – Iterative Autoassociative Memory Networks – Temporal Associative Memory Network. | |
| **Unit-3** | **Genetic Algorithm and related case studies** | **Contact Hours:15** |
| Ch-3.1 | Genetic Algorithm: Introduction –Terminologies in GAs – Traditional Algorithm vs. Genetic Algorithm – Simple GA – General Genetic Algorithm – Basic GA framework | |
| Ch-3.2 | different GA architectures – GA Operators: Encoding, Crossover, Selection, Mutation. Particle Swarm optimization. | |
| Ch-3.3 | **Case Study**: To implement FIS Editor. Use Fuzzy toolbox to model tip value that is given after a dinner based on quality answers. | |

**Textbooks / Reference Books**

**Text Books**:

1. S.N. Shivanandam, Principle of soft computing, Wiley. ISBN13: 9788126527410 (2011)
2. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, “Neuro-Fuzzy and Soft Computing”, PrenticeHall of India, 2003.
3. George J. Klir and Bo Yuan, “Fuzzy Sets and Fuzzy Logic-Theory and Applications”, Prentice Hall, 1995.

**Reference Material**

1. James A. Freeman and David M. Skapura, “Neural Networks Algorithms, Applications, and Programming Techniques”, Pearson Edn., 2003.
2. Mitchell Melanie, “An Introduction to Genetic Algorithm”, Prentice Hall, 1998.
3. David E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison Wesley, 1997.

**Mode of Evaluation:**

The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | 40 | 60 |
| **Total Marks** | 100 | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 | CO1 | PO1 |
| 2 | CO2 | PO1 |
| 3 | CO3 | PO2 AND PO3 |
| 4 | CO4 | PO2 |
| 5 | CO5 | PO1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-332** | **Soft Computing** | **CO1,2,5** | **Co3,4** | **Co3** |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CSP-312** | **Design & Analysis of Algorithms Lab** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **0** | **0** | **2** | **0** | **1** |
|  | | | | | |
| **Pre-requisites** | **Basics of Computers and C/C++** | | | | | |
| **Co-requisites** | **--** | | | | | |
| **Anti-Requisites** | **--** | | | | | |

**Course Objectives:**

* To understand meaning and characteristics of algorithms.
* To study different algorithm design techniques.
* To implement different algorithm design techniques for solving engineering and related problems.

**Course Outcomes:**

* Apply the knowledge of algorithm design techniques to solve the problems of searching, sorting and graph algorithms.
* Design the algorithm using advanced techniques for solving complex problems with Real life Examples.
* Develop the solution of a real time problem using various tools like flowchart, algorithms, programs, etc.
* Utilize the modern engineering tools for algorithm techniques to implementation algorithms for complex engineering problems like divide and conquer, greedy approach, etc.
* Develop algorithms to solve real-time problems like finding shortest path and will able to see function on multi-disciplinary teams through mini projects based on various problems.

**List of Experiments**

* + 1. **List of Practical’s (Graded)**

**UNIT-I**

1. Code and analyze to compute the greatest common divisor (GCD) of two numbers
2. Code implement power function in O(logn) time complexity
3. Code to find frequency of elements in a given array in O(n) time complexity.
4. (i) Code to Insert and Delete an element at the beginning and at end in Doubly and Circular Linked List.

(ii) Code to push & pop and check Isempty, Isfull and Return top element in stacks using templates.

**UNIT-II**

1. Code and analyze to find an optimal solution to matrix chain multiplication using dynamic programming.
2. To implement subset-sum problem using Dynamic Programming
3. Code to implement 0-1 Knapsack using Dynamic Programming

**UNIT-III**

1. Code and analyze to do a depth-first search (DFS) on an undirected graph. Implementing an application of DFS such as (i) to find the topological sort of a directed acyclic graph, OR (ii) to find a path from source to goal in a maze.
2. Code and analyze to find shortest paths in a graph with positive edge weights using Dijkstra’s algorithm.
3. Code and analyze to find all occurrences of a pattern P in a given string S.
   * 1. **List of Practical’s (Non Graded)**
4. Code to Insert and Delete an element at the beginning and end in Singly Linked List.
5. Code for enqueue, dequeue, Isfull and Isempty operation in queues using templates.
6. Code and analyze to sort an array of integers using Quick sort.
7. Code and analyze to sort an array of integers using Merge sort
8. Code to find the height of a Binary tree with Analysis
9. Code to find the ignored successor in an Binary Search Trees with complexity Analysis.
10. To implement maximum and minimum problem using divide and conquer strategy.
11. To implement binary search using divide and conquer strategy.
12. To implement LCS Problem using Dynamic Programming.
13. Code and analyze to do a breadth-first search (BFS) on an undirected graph. Implementing an application of BFS such as (i) to find connected components of an undirected graph, OR (ii) to check whether a given graph is bipartite.
14. Code and analyze to find shortest paths in a graph with arbitrary edge weights using Bellman-Ford algorithm.
15. To implement KMP (Knuth-Morris-Pratt) algorithm.
16. To implement Prim’s algorithm for minimum spanning tree.
17. To implement Kruskal’s algorithm for minimum spanning tree.

**TEXT BOOKS**

1. Cormen, Leiserson, Rivest, Stein, “*Introduction to Algorithms*”, Prentice Hall of India, 3rd edition 2012. problem, Graph coloring.
2. Horowitz, Sahni and Rajasekaran, “*Fundamentals of ComputerAlgorithms”*, University Press (India), 2nd edition.

**REFERENCE BOOKS**

1. Tanenbaum, Augenstein, &Langsam, “*Data Structures using C and C++*”, Prentice Hall of India.
2. Brassard, Bratley, “*Fundamentals of Algorithms*”, Prentice Hall of India.
3. Knuth “*The Art of Computer Programming, Volume 1: Fundamental Algorithms*” (Addison-Wesley, Third Edition).
4. Lipschutz, S., “*Data Structures, Schaum's Outline Series*”, Tata McGraw Hill.
5. Kruse, “*Data Structures & Program Design*”, Prentice Hall of India.
6. Aho, Haperoft and Ullman, ”*The Design and analysis of Computer Algorithms*”, Pearson

Education India.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST-311** | **Design and Analysis of Algorithms** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites** | Studied Data Structures, C/C++ in Previous semesters | | | | | |
| **Co-requisites** | **--** | | | | | |
| **Anti-Requisites** | **--** | | | | | |

**Course Objectives:**

* To understand meaning and characteristics of algorithms
* To study different algorithm design techniques.
* To implement different algorithm design techniques for solving engineering and related problems.

**Course Outcomes:**

* Apply the knowledge of efficiency evaluation of algorithm with respect to time and space complexity of algorithms.
* Describe the various algorithm development approaches to solve the problems like divide and conquer, graph based, tree based, etc.
* Evaluate the complexity of the algorithms to evaluate the efficiency and effectiveness as greedy strategy, dynamic programming strategy and will able to gain knowledge about backtracking, branch and bound and string matching techniques to deal with some hard problems.
* Analyze the various classes for complex problems like P, NP, and NP-Complete and Correlate existing algorithms to improve efficiency.
* Analyze the various techniques for algorithm design and apply the knowledge to solve complex engineering problems.

**Contents of the Syllabus**

**UNIT-1 [15h]**

**Chapter-1 (Algorithms and Program Performance)**

Designing and analysing algorithms, Time and Space complexity, Average and worst case Analysis, Asymptotic notations, recurrence equations and their solution: substitution method, recursion-tree method, master method.

**Chapter-2 (Review of Data Structures)**

Arrays, Stacks, Queues, Pointers, Linked Lists (One –way, Two-way and circular Two-way), Hashing, Trees (BST, B Tree, balanced trees (AVL, Red black trees)), Heaps, Graphs

**Chapter-3 (Sorting algorithm)**

Sorting in linear time: counting sort, radix sort, bucket sort

**UNIT-II [15h]**

**Chapter-4 (Divide and conquer & Greedy algorithms)**

Divide and conquer: The General method, Binary search, Finding maximum and minimum of a sequence of numbers, 2 way Merge sort, Quick sort, Selection sort, Strassen’s matrix multiplication.

Greedy algorithms: The general method, Fractional Knapsack problem, Minimum cost spanning tree: Prim’s Algorithm, Kruskal Algorithm; Huffman coding, Optimal merge patterns.

**Chapter-5 (Dynamic programming)**

The general method, 0/1 knapsack, Subset Sum problem, Change making problem, optimal binary search tree, Matrix-chain Multiplication, Longest common Subsequence Problem, Travelling salesman problem. Comparison of Divide & Conquer and Dynamic Programming techniques.

**Chapter-6 (Backtracking & Branch and Bound)**

Backtracking: The general method, N-queen’s problem, sum-of-subsets, Hamiltonian cycles.

Branch and Bound: Branch and Bound method, 0/1 Knapsack problem, Travelling salesperson problem.

**UNIT-III [15h]**

**Chapter-7 (Graph Algorithms)**

Representation of Graphs, Depth First Search, Breadth First search, Topological sort, Single source shortest path: Dijkstra Algorithm & Bellman Ford Algorithm. All-pair shortest paths: Floyd Warshall Algorithm, Minimum Spanning Tree: Sollin’s algorithm.

**Chapter-8 (Computational complexity)**

Basic concepts, P and NP-classes, proof of NP-hard and NP-completeness.

**Chapter-9 (Miscellaneous topics)**

Euclid Algorithm for GCD of 2 numbers, modulo arithmetic, Chinese remainder theorem, string manipulation/matching algorithms: Rabin Karp algorithm, KMP (Knuth-Morris-Pratt) algorithm, Boyer-Moore algorithm; Convex Hull.

**TEXT BOOKS**

1. Cormen, Leiserson, Rivest, Stein, “*Introduction to Algorithms*”, Prentice Hall of India, 3rd edition 2012. problem, Graph coloring.
2. Horowitz, Sahni and Rajasekaran, “*Fundamentals of ComputerAlgorithms”*, University Press (India), 2nd edition.

**REFERENCE BOOKS**

1. Tanenbaum, Augenstein, &Langsam, “*Data Structures using C and C++*”, Prentice Hall of India.
2. Brassard, Bratley, “*Fundamentals of Algorithms*”, Prentice Hall of India.
3. Knuth “*The Art of Computer Programming, Volume 1: Fundamental Algorithms*” (Addison-Wesley, Third Edition).
4. Lipschutz, S., “*Data Structures, Schaum's Outline Series*”, Tata McGraw Hill.
5. Kruse, “*Data Structures & Program Design*”, Prentice Hall of India.
6. Aho, Haperoft and Ullman, ”*The Design and analysis of Computer Algorithms*”, Pearson

Education India.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CST-353** | **Theory of Computation** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **3** | **0** | **0** | **0** | **3** |
|  | | | | | |
| **Pre-requisites** | Knowing importance of system programs | | | | | |
| **Co-requisites** | **--** | | | | | |
| **Anti-Requisites** | **--** | | | | | |

**Course Objectives:**

* To understand the concept of formal languages and their relation with finite automata.
* To study and design different finite automata.
* To study context free grammars and ambiguity related issues.
* To gain familiarization with Push- Down Automata and Turing Machines.
* To explore relationship between different classes of formal languages.

**Course Outcomes:**

* Gain knowledge of formal languages and classify basic operations on them.
* Illustrate Finite Automata and differentiate DFA and NFA with the help of examples
* Explain and support the properties of Regular sets using pumping lemma and theorems.
* Analyze finite automata with output and compare and contrast CFG, Regular grammar, CNF, GNF.
* Explain Chomsky hierarchy and be familiar with the concept of Turing Machine, Pushdown Automata and justify with examples deterministic and non- deterministic Turing machine

**Contents of the Syllabus**

# UNIT –I

**Introduction:** Basic Terminology: Alphabet, Formal Language and operations on formal languages, Examples of formal languages.

**Finite automata:** Concept of Basic Machines, Properties and Limitations of Finite State Machines, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Equivalence of DFA and NDFA, Non-Deterministic Finite automata with Λ-Transitions.

**Regular expression**: Regular Languages and Regular Expressions, Kleen’s Theorem. Arden’s Method.

# UNIT –II

**Properties of Regular sets:** The Pumping Lemma for Regular sets, Application of the Pumping Lemma, Closure Properties of Regular Sets, Myhill- Nerode Theorem and Minimization of Finite Automata, Minimization Algorithm.

**Finite Automata with output:** Moore and Mealy Machines. Equivalence of Moore and Mealy Machines.

**Context Free Grammars:** Examples and Definitions, Derivation trees and ambiguity, An Unambiguous CFG for Algebraic Expressions. Regular Grammar, Simplified forms and Normal forms: Removal of useless symbols and unit production, Removal of Λ-moves, Chomsky Normal Form (CNF), Griebach Normal Form (GNF).

# UNIT –III

**Context sensitive Language**: Context sensitive Language and grammar, Relation between languages of classes

**Pushdown Automata**: Introduction and Definition of Push-Down Automaton, Applications of Push Down Automata.

**Turing Machines**: Definitions and Examples, Deterministic and Non- Deterministic Turing Machines, Unsolvable Problems: A Non recursive Language and an Unsolvable Problem, PCP Problem and MPCP Problem.

**More General Languages and Grammars:** Recursively Enumerable and Recursive Languages, Unrestricted grammars, Chomsky hierarchies of grammars.

# Text Books:

1. Martin J.C., “Introduction *to Languages and Theory of Computation*”, Tata McGraw-Hill Publishing Company Limited, 3rd Edition.
2. Hopcroft J.E. and Ullman J.D., “Introduction *to Automata Theory Languages and Computation*”, Narosa Publications.

# Reference Books:

1. Sipser,” *Theory of Computation*, Cengage Learning.
2. Daniel I.A. Cohen, “*Introduction to computer Theory*”, John Wiley.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **40** | **60** |
| **Total Marks** | **100** | |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CSP339** | **MULTIMEDIA TECHNOLOGIES** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **0** | **0** | **2** | **0** | **1** |
|  | | | | | |
| **Pre-requisites:** | 21CST-254 | | | | | |
| **Co-requisites** | 21CST-313,21CSP-314,21CST-315,21CST-316,21CST-319,21CSP-321 | | | | | |
| **Anti-Requisites** | 21CST-472 | | | | | |

**Course Objectives:**

* The aim of the syllabus is to provide orientation as regard to uses of Multimedia.
* This course will explain the technologies underlying digital images, videos and audio contents, including various compression techniques and standards, and the issues to deliver multimedia content over the Internet.

**Course Outcomes: (5 Required)**

* To identify the essential features of graphics/image data types, file formats, and colour models in images and video.
* To explain the technical details of multimedia data representations.
* To perform a comparative analysis of the major methods and algorithms for multimedia data compression.
* To explain the technical details of popular multimedia compression standards.
* To perform different operations on videos.

**Contents of the Syllabus:**

**UNIT-I [15h]**

Experiment 1: Write a program to justify a text entered by the user on both left and right hand side.forexample the test “ An architect may have a graphics program to draw an entire building but be interested in only ground floor”,can be justified in 30 columns.An architect may have a graphics programs draw an entire building but interested in ground floor.

Experiment 2: Study the notes of a piano and stimulate them using the keyboard and store them in file

Experiment 3: Write a program to read a paragraph and store it to a file name suggested by the author

Experiment 4: Devise a routine to produce the animation effect of a square transforming to a triangle and then to a circle.

**UNIT-II [15h]**

Experiment 5: Write a program to show a bitmap image on your computer screen.

Experiment 6: Create a web page for a clothing company which contains all the details of that company and at least five links to other web pages.

Experiment 7: Write a program by which we can split mpeg video into smaller pieces for the purpose of sending it over the web or by small capacity floppy diskettes and then joining them at the destination.

**UNIT-III**

Experiment 8: Write a program to simulate the game of pool table

# Experiment 9: Write a program to simulate the game mine sweeper

# Experiment 10: Write a program to play “wave” or “midi” format sound files

**TEXT BOOKS**

1. Tay Vaughan, “Multimedia making it work”, Tata McGraw-Hill, 2008.
2. Rajneesh Aggarwal & B. B Tiwari, “ Multimedia Systems”, Excel Publication, New Delhi, 2007.
3. Li & Drew, “ Fundamentals of Multimedia” , Pearson Education, 2009..

**REFERENCE BOOKS**

1. Parekh Ranjan, “Principles of Multimedia”, Tata McGraw-Hill, 2007
2. AnirbanMukhopadhyay and Arup Chattopadhyay, “Introduction to Computer Graphics and Multimedia”, Second Edition, Vikas Publishing House.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Lab** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **60** | **40** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-339** | **MULTIMEDIA TECHNOLOGIES** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CSP378** | **WEB TECHNOLOGIES lab** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **0** | **0** | **2** | **0** | **1** |
|  | | | | | |
| **Pre-requisites:** | 21CST-254 | | | | | |
| **Co-requisites** | **21CST-352,21CST-353,21CST-354,21CST-355,21CST-357,21CST-371,21CSP-356** | | | | | |
| **Anti-Requisites** | 21CST-474 | | | | | |

**Course Objectives:**

* This Subject is useful for Making own Web page and how to host own web site on internet. Along with that Students will also learn about the protocols involv in internet technology.

**Course Outcomes: (5 Required)**

* History and development of the World Wide Web and associated technologies.
* The client-server architecture of the World Wide Web and its communication protocol HTTP/HTTPS.
* Formats and languages used in modern web-pages: HTML, XHTML, CSS, XML, XSLT, JavaScript, DOM.
* Programming web pages with JavaScript/DOM (client)
* Good design, universal design, multi-platform web applications.

**Contents of the Syllabus:**

**UNIT-I [15h]**

Experiment 1: Design the following static web pages required for an online book store web site.

1) HOME PAGE: The static home page must contain three frames.

2) LOGIN PAGE

3) CATOLOGUE PAGE: The catalogue page should contain the details of all the books available in the web site in a table.

4) REGISTRATION PAGE

Experiment 2: **Write *JavaScript* to validate the following fields of the Registration page.**

**1. First Name** (Name should contains alphabets and the length should not be less than 6 characters).

**2. Password** (Password should not be less than 6 characters length).

**3. E-mail id** (should not contain any invalid and must follow the standard pattern name@domain.com)

**4. Mobile Number** (Phone number should contain 10 digits only).

**5. Last Name and Address** (should not be Empty).

Experiment 3: **Develop and demonstrate the usage of inline, internal and external style sheet using CSS**

Experiment 4: **Develop and demonstrate JavaScript with POP-UP boxes and functions for the following problems:**

**a)** Input: Click on Display Date button using onclick( ) function

Output: Display **date** in the textbox

**b)** Input: A number n obtained using **prompt**

Output: **Factorial** of n number using **alert**

**c)** Input: A number n obtained using **prompt**

Output: A **multiplication table** of numbers from 1 to 10 of n using

alert

**d)** Input: A number n obtained using **prompt** and add another number

using **confirm**

Output: **Sum** of the entire n numbers using **alert**

**UNIT-II**

Experiment 5: **Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next in the list. Add CSS to customize the properties of the font of the capital (color,bold and font size).**

Experiment 6: **Write an HTML page including any required JavaScript that takes a number from text field in the range of 0 to 999 and shows it in words. It should not accept four and above digits, alphabets and special characters.**

Experiment 7: **Develop and demonstrate PHP Script for the following problems:**

**a) Write a PHP Script to find out the Sum of the Individual Digits.**

**b) Write a PHP Script to check whether the given number is Palindrome or not**

**UNIT-III**

Experiment 8: **Create an XML document that contains 10 users information. Write a Java Program, which takes User Id as input and returns the user details by taking the user information from XML document using DOM parser or SAX parser.**

Experiment 9: **Implement the following web applications using (a) PHP**

(b) Servlets

(c) JSP

Experiment 10: **Implement the web applications with Database using**

**(a) PHP, (b) Servlets and (c) JSP.**

**TEXT BOOKS/REFERENCE BOOKS**

1. Steven Holzner, ”HTML Black Book”, Dremtechpress.
2. Web Technologies, Black Book, DreamtechPress
3. Web Applications : Concepts and Real World Design, Knuckles,Wiley-India
4. Internet and World Wide Web How to program, P.J. Deitel& H.M. DeitelPearson..

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **60** | **40** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and POs** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-378** | **WEB TECHNOLOGIES lab** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Subject Code**  **20CSP379** | **UI/UX DESIGN lab** | **L** | **T** | **P** | **S** | **C** |
| **Total Contact Hours : 45 Hours** | **0** | **0** | **2** | **0** | **1** |
|  | | | | | |
| **Pre-requisites:** | **21CST-258** | | | | | |
| **Co-requisites** | **21CST-352,21CST-353,21CST-354,21CST-355,21CST-357,21CST-371,21CSP-356** | | | | | |
| **Anti-Requisites** | **21CST-473** | | | | | |

**Course Objectives:**

* Describe the web user Interface
* Describe the structure of user Interface and design process
* Organize the web systems and control.

**Course Outcomes: (5 Required)**

* Students are expected to describe the Characteristics of Graphics Interface and its Principles.
* Students are expected to design the standards and structures for Human computer interaction.
* Students are expected to explain the technical details of multimedia data representations.
* Student are expected Decompose a problem into function
* Students are expected Develop an application using the concepts of array, pointer, structure, and file management to solve engineering and/or scientific problems

**Contents of the Syllabus:**

**UNIT-I**

Experiment 1: (a) Write a console application that obtains four int values from the user and displays the product.

(b) If you have two integers stored in variables var1 and var2, what Boolean test can you perform to see if one or the other (but not both) is greater than 10?

(c) Write an application that includes the logic from Exercise 1, obtains two numbers from the user, and displays them, but rejects any input where both numbers are greater than 10 and asks for two new numbers.

(d) Write a console application that places double quotation marks around each word in a string .

(e) Write an application that uses two command-line arguments to place values into a string and an integer variable, respectively. Then display these values.

(f) Write an application that receives the following information from a set of students: Student Id: Student Name: Course Name: Date of Birth: The application should also display the information of all the students once the data is Entered. Implement this using an Array of Structures.

(g) Write programs using conditional statements and loops:

I) Generate Fibonacci series.

II) Test for prime numbers.

III) Generate prime numbers.

IV) Reverse a number and find sum of digits of a number.

V) Test for vowels.

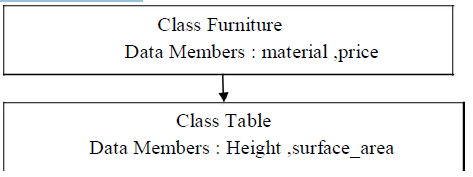
VI) Use of foreach loop with arrays.

Experiment 2: (a) Write a program to declare a class „staff‟ having data members as name and post.accept this data 5for 5 staffs and display names of staff who are HOD.

(b) Write a program to declare class „Distance‟ have data members dist1,dist2 ,dist3. Initialize the two data members using constructor and store their addition in third data member using function and display addition.

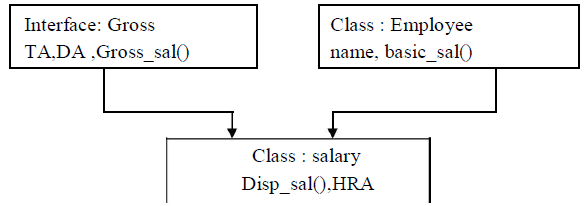
(c) Write a program using function overloading to swap two integer numbers and swap two float numbers.

(d) Write a program to implement single inheritance from following figure. Accept and display data for one table.

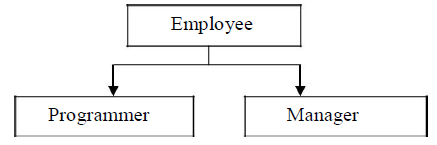


(e) Define a class „salary‟ which will contain member variable Basic, TA, DA, HRA. Write a program using Constructor with default values for DA and HRA and calculate the salary of employee.

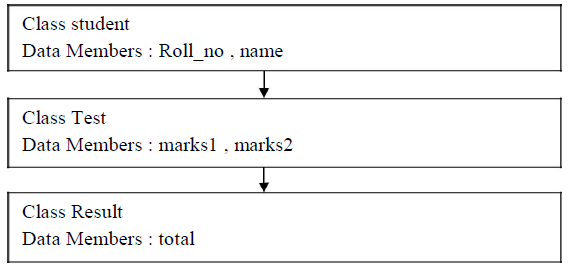
(f) Program to implement the following multiple inheritance using interface.



(g) Write a program for above class hierarchy for the Employee where the base class is Employee and derived class and Programmer and Manager. Here make display function virtual which is common for all and which will display information of Programmer and Manager interactively.



(h) Write a program to implement multilevel inheritance from following figure. Accept and display data for one student.



(i) Write a program to create a delegate called TrafficDel and a class called TrafficSignal with the following delegate methods. Public static void Yellow() { Console.WriteLine(“Yellow Light Signal To Get Ready”); } Public static void Green() { Console.WriteLine(“Green Light Signal To Go”); } Public static void Red() { Console.WriteLine(“Red Light Signal To Stop”); } Also include a method IdentifySignal() to initialize an array of delegate with the above methods and a method show() to invoke members of the above array.

(j) Write a program to accept a number from the user and throw an exception if the number is not an even number.

Experiment 3:(a) Create an application that allows the user to enter a number in the textbox named „getnum‟. Check whether the number in the textbox „getnum‟ is palindrome or not. Print the message accordingly in the label control named lbldisplay when the user clicks on the button „check‟.

(b) Create an application which will ask the user to input his name and a message, display the two items concatenated in a label, and change the format of the label using radio buttons and check boxes for selection , the user can make the label text bold ,underlined or italic and change its color . include buttons to display the message in the label, clear the text boxes and label and exit.

(c) List of employees is available in listbox. Write an application to add selected or all records from listbox (assume multi-line property of textbox is true).

(d) “How is the book ASP.NET with c# by Vipul Prakashan?” Give the user three choice : i)Good ii)Satisfactory iii)Bad. Provide a VOTE button. After user votes, present the result in percentage using labels next to the choices.

(e) Create a project that calculates the total of fat, carbohydrate and protein. Allow the user to enter into text boxes. The grams of fat, grams of carbohydrate and grams of protein. Each gram of fat is 9 calories and protein or carbohydrate is 4 calories. Display the total calories of the current food item in a label. Use to other labels to display and accumulated some of calories and the count of items entered. The form food have 3 text boxes for the user to enter the grams for each category include label next to each text box indicating what the user is enter.

Experiment 4: (a) Set the label border color of rollno to red using css.

(b) Set the font-Arial , font style-bond , font size-18px of different controls(ie. Label, textbox, button) using css.

(c) Design the same webpages for BMS, BAF, BscIT students and apply same background color for all the pages using css.

(d) Change the font family and color of all heading of above webpage using css.

(e) Use pseudo classes and display link, visited link and active link of contact us differently.

**UNIT-II**

Experiment 5: (a) Programs using ASP.NET Server controls.

Create the application that accepts name, password ,age , email id, and user id. Allthe information entry is compulsory. Password should be reconfirmed. Age should be within 21 to 30. Email id should be valid. User id should have at least a capital letter and digit as well as length should be between 7 and 20 characters.

(b) Programs using ASP.NET Server controls.

Create a website for a bank and include types of navigation.

Experiment 6: (a) Database programs with ASP.NET and ADO.NET.

Create a Web App to display all the Empname and Deptid of the employee from the database using SQL source control and bind it to GridView . Database fields are(DeptId, DeptName, EmpName, Salary).

(b) Database programs with ASP.NET and ADO.NET

Create a Login Module which adds Username and Password in the database. Username in the database should be a primary key.

(c) Database programs with ASP.NET and ADO.NET

Create a web application to insert 3 records inside the SQL database table having following fields( DeptId, DeptName, EmpName, Salary). Update the salary for any one employee and increment it to 15% of the present salary. Perform delete operation on 1 row of the database table.

Experiment 7: (a) Programs using Language Integrated query. Create the table with the given fields. FIELD NAME DATA TYPE EmpNo number EmpName varchar EmpSal number EmpJob varchar EmpDeptNo number.

For the given table design a web page to display the employee information from table to grid control. Use LINQ TO ADO.NET.

(b) Programs using Language Integrated query. Create the table with the given fields. FIELD NAME DATA TYPE SRollno int SName string SAddress string SFees int.

For the given table design a web page to display the employee information from table to grid control. Use LINQ TO XML.

(c) Programs using Language Integrated query. Create the table with the given fields . FIELD NAME DATA TYPE PID string PName string PPrice int PWeight int

For the given table design a web page to display the employee information from table to grid control. Use LINQ TO Objects.

**UNIT-III**

Experiment 8: (A) For the web page created for the display OF Employee data change the authentication mode to Windows.

(B) For the webpage created for the display of Student data change the authorization mode so that only users who have logged in as VSIT will have the authority to aces the page

Experiment 9: (a) Create a web page to display the news from the news table(id, news\_dtl). Use AJAX.

(b) In the above website also display the feedback on the browser as “work is in progress”.

(c) Create a web page to display the cricket score from the table event(id, name, score). Refresh the website automatically after every 30 seconds.

Experiment 10: (a) Create a web page to give different color effects for paragraph tags, heading tags and complete web page using JQuery.

(b) Create a web page to display animation using JQuery.

(c) Create a web page to display hide, show, slidedown, slideup and Toggle effects for paragraph tags, using JQuery.

**TEXT BOOKS**

1. Wilbent. O. Galitz ,“The Essential Guide To User Interface Design”, John Wiley& Sons, 2001.
2. Ben Sheiderman, “Design The User Interface”, Pearson Education, 1998.**T3.** Li & Drew, “ Fundamentals of Multimedia” , Pearson Education, 2009.

**REFERENCE BOOKS**

1. Alan Cooper, “The Essential Of User Interface Design”, Wiley-Dream Tech Ltd.,2002.

# Mode of Evaluation: The performance of students is evaluated as follows:

|  |  |  |
| --- | --- | --- |
|  | **Theory** | |
| **Components** | **Continuous Internal Assessment (CAE)** | **Semester End Examination (SEE)** |
| **Marks** | **60** | **40** |
| **Total Marks** | **100** | |

**Relationship between the Course Outcomes (COs) and Program Outcomes (POs)**

|  |  |  |
| --- | --- | --- |
| **Mapping Between COs and Pos** | | |
| **SN** | **Course Outcome (CO)** | **Mapped Programme Outcome (PO)** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Engineering Knowledge** | **Problem analysis** | **Design/development of solutions** | **Conduct investigations of complex** | **Modern tool usage** | **The engineer and society** | **Environment and sustainability** | **Ethics** | **Individual or team work** | **Communication** | **Project management and finance** | **Life-long Learning** |
| Course Code | Course Name | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **1**  **0** | **1**  **1** | **12** |
| **20CST-379** | **UI/UX DESIGN lab** |  |  |  |  |  |  |  |  |  |  |  |  |

1 = addressed to small extent

2 = addressed significantly

3 = major part of course