**w.e. f. June 2019**

**Bachelor of Technology (B.Tech) in Computer Science & Engineering: 5th Semester Scheme**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr. No.** | **Course**  **No.** | **Course Category** | **Course Title** | **Teaching Schedule** | | | **Credits** | **Examination Marks** | | | |
|  | **L** | **T** | **P** | **Internal** | **Theory** | **Practical** | **Total** |
| 1. | MA-05 / HUM-11 / HUM-12 | General | Multivariate Analysis, Linear Algebra and Special Functions / Economics for Engineers / Law for Engineers / \*Swayam | 3 | 1 | - | 4.0 | 40 | 60 | - | 100 |
| 2. | MA-08 / SAP-02/ HUM-13 | General | Probability and Statistics  / SAP ABAP / Marketing Management /\* Swayam | 2 | 1 | - | 3.0 | 40 | 60 | - | 100 |
| 3. | CS-22 | Specialization | Web Technologies | 3 | - | - | 3.0 | 40 | 60 | - | 100 |
| 4. | CS-23 | Specialization | Analysis and Design of Algorithms | 3 | 1 | - | 4.0 | 40 | 60 | - | 100 |
| 5. | CS-24 | Specialization | Unix & Linux Shell Programming | 3 | - | - | 3.0 | 40 | 60 | - | 100 |
| 6. | CS-25 | Specialization | Business Intelligence & its Applications | 3 | - | - | 3.0 | 40 | 60 | - | 100 |
| 7. | CS-26 | Specialization | Analysis and Design of Algorithms Lab | - | - | 2 | 2.0 | 60 | - | 40 | 100 |
| 8. | CS-27 | Specialization | Unix & Linux Shell Programming Lab | - | - | 2 | 2.0 | 60 | - | 40 | 100 |
| 9. | CS-28 | Specialization | Web Technologies Lab | - | - | 2 | 2.0 | 60 | - | 40 | 100 |
| 10. | CS-29 | Core | Project-I | - | - | 4 | 4.0 | 60 | - | 40 | 100 |
| **Total** | | | | **17** | **3** | **10** | **30** | **480** | **360** | **160** | **1000** |

**\*One Course -3 credit through Swayam**

\*Swayam means subject that cover under self study/learning mode via online study material

**B. Tech. (5thSem) Computer Science &Engineering**

**CS-22 (Web Technologies)**

**L T P Continuous evaluation 40**

**3 - - End semester exam 60**

**Total marks 100**

**Credits 3.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To analyze a web page and identify its elements and attributes. |
|  | To learn the Hypertext Markup Language (HTML) and basics of Java script. |
|  | To learn the Extensible Markup Language (XTML). |
|  | To learn about PHP. |

**Unit-1: Information Architecture**

The role of Information Architect, Collaboration and communication, Organizing information, organizational challenges, Organizing web sites and Intranets, Creating cohesive organization systems, designing navigation systems, types of navigation systems, Integrated navigation elements, designing elegant navigation systems, Searching systems, Searching your web site, designing the search interface, Indexing the right stuff, To search or not to search grouping content, conceptual design, High level Architecture Blueprint, Architectural Page Mockups, Design Sketches, Website development and its phases, Introduction to user interface.

**Unit-2: HTML & Fundamentals of JavaScript**

HTML- Basic concepts, Good web design, web content types, text formatting tags, div and span tag, Images and Anchors, working with HTML5 forms, Html5 elements, Html5 multimedia, Style sheets, text formatting using CSS, Embedded CSS, Inline CSS and external CSS, Background and Color Gradients in CSS, Fonts and text Styles, Creating Boxes and Columns using CSS, Displaying, Positioning and floating an element, List Styles, Table Layouts.

JavaScript Fundamentals- Introduction to JavaScript, JavaScript functions, Events, Arrays, time and date functions.

**Unit-3: XML**

Relationship between HTML, SGML and XML, Basic XML, Valid documents, ways to use XML, XML for data files, embedding XML into HTML documents, Converting XML to HTML for Display, Displaying XML using CSS and XSL, rewriting HTML as XML, the future of XML.

**Unit-4: PHP**

Introduction, variables and operators, Writing PHP Code, Control Statements, Functions, Arrays, Strings, and time and date functions, Introduction to forms, Handling form data with get and post method, Sessions- creating session and handling sessions data. Databases- Creating Databases, SQL statements, insert update, delete and retrieving table data.

Files- Reading and writing files using PHP.

**Course Outcomes:**

* Able to understand, analyze and apply the role of languages like HTML, DHTML, CSS, XML, JavaScript, VBScript, ASP, PHP and protocols in the workings of the web and web applications
* Able to analyze a web project and identify its elements and attributes in comparison to traditional projects.
* Able to understand, analyze and create web pages using HTML, DHTML and Cascading Styles sheets.
* Able to understand, analyze and build web applications using PHP.
* Able to understand, analyze and create XML documents and XML Schema.

**Instructions for paper setter:** All Questions are compulsory. The Question paper is divided in to four sections A, B, C and D. Section A is compulsory and comprises of 12 questions of one mark each, 3 from each unit. The questions shall be asked in such a manner that there are no direct answers including one word answer, fill in the blanks or multiple choice questions. Section B comprises of 4 questions of 2 marks each, one from each unit. Section C Comprises of 4 questions of 4 marks each, one from each unit. Section D Comprises of 4 questions of 6 marks each, one from each unit. There is no overall choice, however internal choice may be provided in section C and D, if paper setter so desires.

**Text Books/Reference Books:**

1. Thomas A Powell, “HTML The Complete Reference”, Tata McGraw Hill Publications.

2. Yong, “XML step by step”, PHI.

3. HTML5, CSS3, JavaScript - HTML 5 Black Book: Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP and jQuery-by Kogent Learning Solutions Inc.

4. Matt Doyle, “Php- Beginning Php 5.3”, Wrox Publishers.

**B. Tech. (5thSem) Computer Science &Engineering**

**CS-23 (Analysis & Design of Algorithms)**

**L T P Continuous evaluation 40**

**3 1 - End semester exam 60**

**Total marks 100**

**Credits 4.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | Analyze the asymptotic performance of algorithms. |
|  | Apply important algorithmic design paradigms and methods of analysis. |
|  | To study about greedy, dynamic, backtracking and branch and bound techniques. |
|  | To study various shortest path algorithms. |
|  | To understand the mathematical criterion for deciding whether an algorithm is efficient, and know many practically important problems that does not admit any efficient algorithms. |

**Unit:-1**

Introduction- Algorithm, Analyzing algorithms, asymptotic notation, recurrence relations, sorting in linear time.

Problem solving techniques- Introduction to problem solving, Computational problem & its classification, Logic & its types.

Divide & Conquer strategy- Methodology, Merge sort, Quick sort.

**Unit:-2**

Greedy algorithms- Elements, activity-selection problem, Huffman codes, task scheduling problem, Knapsack problems, Minimum spanning trees using Kruskal and Prim's algorithm, Single source shortest paths (Dijkstra's algorithm).

Dynamic programming- Elements, Matrix-chain multiplication, Longest common subsequence, All pairs shortest paths (shortest paths and matrix multiplication, Floyd-Warshall algorithm).

**Unit:-3**

Back Tracking- Overview, 8-Queen problem, Knapsack problem. Branch and Bound- LC Searching Bounding, FIFO Branch and Bound, LC Branch and Bound Application (0/1 knapsack Problem, Traveling Salesman Problem). Problem Classes- Introduction to P, NP, NP-hard and NP-complete problems.

**Unit:-4**

Directed Acyclic Graph- Topological sort, Strongly connected component, Single source shortest paths for directed acyclic graphs, Single source shortest path (Bellman-Ford algorithm), Difference constraints and shortest paths.

Flow Network- Ford-Fulkerson method, Maximum bipartite matching.

**Course Outcomes:**

* Able to calculate time complexity of any given algorithm.
* Able to solve real life problems with the help of various problem solving techniques such as Divide Conquer, Greedy & Dynamic programming.
* Able to identify class of particular problem (whether solvable or not).
* Apply and implement graph algorithms to solve real life problems.

**Instructions for paper setter:** All Questions are compulsory. The Question paper is divided in to four sections A, B, C and D. Section A is compulsory and comprises of 12 questions of one mark each, 3 from each unit. The questions shall be asked in such a manner that there are no direct answers including one word answer, fill in the blanks or multiple choice questions. Section B comprises of 4 questions of 2 marks each, one from each unit. Section C Comprises of 4 questions of 4 marks each, one from each unit. Section D Comprises of 4 questions of 6 marks each, one from each unit. There is no overall choice, however internal choice may be provided in section C and D, if paper setter so desires.

**Text Books/Reference Books:**

1. Cormen, Leiserson and Rivest, “Introduction to Algorithms”, 2nd edition, PHI.

2. Horowitz, Ellis and Sahni, Sartaj, “Fundamentals of Computer Algorithms”, Galgotia Publications.

3. Aho, Hopcroft and Ullman, “The Design and Analysis of Computer Algorithms”, Addison Wesley.

4. R. B. Patel, “Expert Data Structures with C”, Khanna Publications, 2nd Edition 2004.

**B. Tech. (5thSem) Computer Science &Engineering**

**CS-24 (Unix & Linux Shell Programming)**

**L T P Continuous evaluation 40**

**3 - - End semester exam 60**

**Total marks 100**

**Credits 3.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To study the basics of Unix OS and shell programming. |
|  | To learn about Unix file system. |
|  | To study about the installation and configuration of a Linux OS. |
|  | To study about how to manage user accounts, file system on Linux platform. |

**Unit-1: UNIX Startup & Shell Programming**

UNIX Startup- User accounts, Accessing, starting and shutting processes, Logging in and Logging out, Command line, Simple commands. Shell Programming- Unix file system, Unix files, i-nodes and structure and file system related commands, Shell as command processor, Shell variables, Creating command substitution, Scripts, Functions, Conditionals, Loops, Customizing environment

**Unit-2:** **Regular Expressions and Filters**

Introducing regular expressions patterns, Syntax, Character classes, Quantifiers, Introduction to egrep, sed, Programming with awk and perl.

**Unit-3: The C Environment**

The C compiler, vi editor, Compiler options, Managing projects, Memory management, Use of makefiles, Dependency calculations, Memory management (Dynamic and static memory), Building and using static and dynamic libraries, Using ldd, soname, Dynamic loader, Debugging with gdb.

**Unit-4: Processes in Linux**

Processes, Starting and stopping processes, Initialization processes, rc and init files, Job control (at, batch, cron, time), Network files, Security, Privileges, Authentication, Password administration, Archiving, Signals and signal handlers, Linux I/O system.

**Course Outcomes:**

* Able to understand line and screen text editors with regular expressions.
* Explain Unix file system including advanced file processing and practice pipelining and IO redirecting.
* Able to write complex shell scripts and cooperating processes.
* Able to understand the installation and configuration of a Linux system.
* Able to manage user accounts, file systems, networking and system logs on Linux platform.
* Understand the integration of Linux with other operating environments.

**Instructions for paper setter:** All Questions are compulsory. The Question paper is divided in to four sections A, B, C and D. Section A is compulsory and comprises of 12 questions of one mark each, 3 from each unit. The questions shall be asked in such a manner that there are no direct answers including one word answer, fill in the blanks or multiple choice questions. Section B comprises of 4 questions of 2 marks each, one from each unit. Section C Comprises of 4 questions of 4 marks each, one from each unit. Section D Comprises of 4 questions of 6 marks each, one from each unit. There is no overall choice, however internal choice may be provided in section C and D, if paper setter so desires.

**Text Books/Reference Books:**

1. John Goerzen, “Linux Programming Bible”, IDG Books, New Delhi, 2000.

2. Sumitabha Das, “Your Unix - The Ultimate Guide”, TMH, 2000.

3. Mathew, “Professional Linux Programming”, Vol. l & 2, Wrox-Shroff, 2001.

4. Welsh & Kaufmann, “Running Linux”, O’Reiley & Associates, 2000.

**B. Tech. (5thSem) Computer Science &Engineering**

**CS-25 (Business Intelligence and its applications)**

**L T P Continuous evaluation 40**

**3 - - End semester exam 60**

**Total marks 100**

**Credits 3.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To acquire the knowledge about the terminology, classifications, methods, trends used in Business Intelligence. |
|  | To learn about the Extraction Transformation Loading used for data integration. |
|  | To learn about various modeling scheme used for multi-dimensional data modeling. |
|  | To learn about tools used for enterprise reporting. |

**Unit-1: Introduction to Business Intelligence**

Introduction to OLTP and OLAP, BI definition & Concepts, Business Applications of BI, BI Framework, Role of Data Warehousing in BI, BI infrastructure Components (BI Process, BI Technology, BI Roles & Responsibilities).

**Unit-2:** **Basics of Data Integration (Extraction Transformation Loading)**

Concepts of data integration need and advantages of using data integration, Introduction to common data integration approaches, Introduction to ETL using SSIS, Introduction to data quality, Data profiling concepts and applications.

**Unit-3: Introduction to Multi-Dimensional Data Modeling**

Introduction to data and dimension modeling, multidimensional data model, ER Modeling vs. multi dimensional modeling, Concepts of dimensions, Facts, Cubes, Attribute, Hierarchies, Star and snowflake schema, Introduction to business metrics and KPIs, Creating cubes using SSAS.

**Unit-4: Basics of Enterprise Reporting**

Introduction to enterprise reporting, concepts of dashboards, Balanced scorecards, Introduction to SSRS Architecture, Enterprise reporting using SSRS.

**Course Outcomes:**

* Appraise and apply evidence practice (EBP) to formulate effective solutions to deal with contemporary performance problems and issues associated with the delivery of business information systems.
* Create a consultant report that critically evaluates important design principles and operations involving business intelligence and that offers effective recommendations aimed at enhancing business outcomes.
* Devise a framework to assess company/industry performance and to apply it to produce a performance report of a nominated entity.
* Evaluate the importance and implementation of learning theory to construct and apply practices that facilitate aspects of personal and institutional change.
* Demonstrate competence in oral, written, and visual communication in business reports and presentations.

**Instructions for paper setter:** All Questions are compulsory. The Question paper is divided in to four sections A, B, C and D. Section A is compulsory and comprises of 12 questions of one mark each, 3 from each unit. The questions shall be asked in such a manner that there are no direct answers including one word answer, fill in the blanks or multiple choice questions. Section B comprises of 4 questions of 2 marks each, one from each unit. Section C Comprises of 4 questions of 4 marks each, one from each unit. Section D Comprises of 4 questions of 6 marks each, one from each unit. There is no overall choice, however internal choice may be provided in section C and D, if paper setter so desires.

**Text Books/Reference Books:**

1. R. N. Prasad, Seema Acharya, “Fundamentals of Business Analytics”.

2. David Loshin, “Business Intelligence”.

3. Mike Biere, “Business Intelligence for the enterprise”.

4. Larissa Terpeluk Moss, Shaku Atre, “Business Intelligence roadmap”.

5. Cindi Howson, “Successful Business Intelligence: Secrets to making Killer BI Applications”.

6. Brain, Larson, “Delivering business intelligence with Microsoft SQL server 2008”.

7. Lynn Langit , “Foundation of SQL Server 2005 Business Intelligence”.

8. Stephen Few, “Information dashboard design”.

**B. Tech. (5thSem) Computer Science &Engineering**

**SAP-02 (SAP-ABAP)**

**L T P Continuous evaluation 40**

**2 1 - End semester exam 60**

**Total marks 100**

**Credits 3.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To build an understanding of the fundamental concepts of ERP systems, their architecture, and working of different modules in ERP. |
|  | To understand the concepts of ABAP Objects and how to use them in Workbench tools to develop their own business applications. |
|  | To enhance the Standard SAP Software to meet specific customer requirements. |
|  | To understand the key concepts in the Object Oriented Programming is to integrate it into SAP ABAP to implement the various applications. |

**Unit:-1**

Introduction- SAP systems, SAP portfolio overview, Navigation basics.

System core- AS ABAP, AS JAVA architecture and various processes, Communication and Integration technologies. Introduction to data dictionary, database tables, views, domain, data types, type group, search help, indexes, text table.

**Unit:-2**

Introduction to ABAP programming- Data types, types of programs, create packages, internal tables, various operations on internal tables, Joins, Control break statements, Data selection statements, modularization techniques (function modules, subroutines, includes).

**Unit:-3**

Classical ABAP reports, Selection screen, Introduction to screen programming: simple screen elements, screen error handling, subscreens, tabstrip controls, ALV (classical ALV reports, function module ALV).

**Unit:-4**

Introduction to object oriented programming and its syntax, inheritance and casting, interfaces and casting, object oriented events, object oriented repository objects, class based exceptions, design patterns, program calls and memory management, introduction to Webdynpro, enhancement of dictionary elements.

**Course Outcomes:**

* Understand the ABAP data types, terminology, and dictionary concepts, including structures, tables, buffers, indexes etc.
* Write programs by implementing ABAP programming basics, covering formatting, string/data operations, syntax check, performance trace and more
* Use modularization techniques covering macros, includes, subroutines, and function modules
* Create programs using module pools, file handling, and scripting.
* Work with advanced features say BADI, ALE, IDOC, RFC, ALV, LSMW, and more.

**Note: Online multiple choice examinations will be taken at the end of semester.**

**Text Books/Reference Books:**

* Kogent Learning Solutions Inc., SAP ABAP/4 Covers SAP ECC 6.0 Black Book, Dreamtech Press (2009).
* ABAP Workbench fundamentals TAW10 Part-1 and Part-II ,SAP.
* ABAP Workbench Concepts TAW12 Part-1 and Part-II ,SAP

**B. Tech. (5th Sem) Computer Science & Engineering**

**CS-26 (Analysis & Design of Algorithms Lab)**

**L T P Continuous evaluation 60**

**- - 2 End semester exam 40**

**Total marks 100**

**Credits 2.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To acquire knowledge about how to design new algorithms, prove them correct, and analyze their asymptotic behavior. |
|  | To apply classical sorting algorithms using programming language. |
|  | To apply classical searching algorithms using programming language. |
|  | To apply optimization and graph algorithms. |

**List of Practical**

Experimental work will be based upon the course Analysis & Design of Algorithms(CS-23).

**Course Outcomes:**

|  |
| --- |
| * Able to identify the problem given and design the algorithm using various algorithm design techniques. * Able to implement various algorithms in a high level language. * Able to analyze the performance of various algorithms. * Able to compare the performance of different algorithms for same problem. |
|  |
|  |
|  |

**B. Tech. (5th Sem) Computer Science & Engineering**

**CS-27 (Unix & Linux Shell Programming Lab)**

**L T P Continuous evaluation 60**

**- - 2 End semester exam 40**

**Total marks 100**

**Credits 2.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To learn how to write shell script programs to solve problems. |
|  | To learn C programming in Unix editor environment |
|  | To implement some standard Linux utilities such as ls.cpetc using system calls. |
|  | To develop network based applications. |

**List of Practical**

Experimental work will be based upon the course Unix & Linux Shell Programming(CS-24).

**Course Outcomes:**

|  |
| --- |
| * Able to identify the basic Unix general purpose commands. * Able to apply and change the ownership and file permissions using advance Unix commands. * Able to implement shell scripts and sed. * Able to apply networking Unix commands. |

**B. Tech. (5th Sem) Computer Science & Engineering**

**CS-28 (Web Technologies Lab)**

**L T P Continuous evaluation 60**

**- - 2 End semester exam 40**

**Total marks 100**

**Credits 2.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To analyze a web page and identify its elements and attributes. |
|  | To create web pages using XHTML and Cascading Style Sheets. |
|  | To build dynamic web pages using JavaScript. |
|  | To create XMLdocuments and Schemas. |

**List of Practical**

Experimental work will be based upon the course Web Technologies(CS-22).

**Course Outcomes:**

|  |
| --- |
| * Able t o develop a static webpage by the use of HTML. * Able to develop a dynamic webpage by the use of JavaScript and XHTML. * Able to write a well formed / valid XML document. * Able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table. |

**B. Tech. (5th Sem) Computer Science & Engineering**

**CS-29 (Project-I)**

**L T P Continuous evaluation 60**

**- - 4 End semester exam 40**

**Total marks 100**

**Credits 4.0**

**Course Objectives:**

|  |  |
| --- | --- |
|  | To learn about various phases of software development life cycle. |
|  | To learn about how to provide software solution for real life problems. |
|  | To learn about coding and testing of solutions. |
|  | To learn about report writing concepts. |

The students are required to develop a project during semester and final evaluation will be entirely based upon his/her project work.

**Course Outcomes:**

|  |
| --- |
| * Able to identify software solution corresponding to real life problems. * Able to code software solution. * Able to test software solution. * Able to write reports. |