# B.Sc., Computer Science

**Allied**

| Subject Code | Subject Name | Category | L | T | P | S | Credits | Inst. Hours | Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | CIA | External | Total |
| **23BCEA1** | **Digital Logic Fundamentals** | Allied | 3 | - | - | - | 3 | 3 | 25 | 75 | 100 |
| **Learning Objective** | | | | | | | | | | | |
| LO1 | It aims to train the student to the basic concepts of Digital Computer Fundamentals | | | | | | | | | | |
| LO2 | To impart the in-depth knowledge of logic gates, Boolean algebra, combinational circuits and sequential circuits. | | | | | | | | | | |
| | **Contents** | | | | | | | | | | |
| **UNIT I** | Number Systems and Codes: Number System – Base Conversion – Binary Codes – Code Conversion. Digital Logic: Logic Gates – Truth Tables – Universal Gates. | | | | | | | | | | |
| **UNIT II** | Boolean Algebra: Laws and Theorems – SOP, POS Methods– Simplification of Boolean Functions – Using Theorems, K-Map, Prime – Implicate Method – Binary Arithmetic: Binary Addition – Subtraction – Various Representations of Binary Numbers –Arithmetic Building Blocks–Adder– Subtractor. | | | | | | | | | | |
| **UNIT III** | Combinational Logic: Multiplexers – Demultiplexers – Decoders – Encoders – Code Converters – Parity Generators and Checkers. | | | | | | | | | | |
| **UNIT IV** | Sequential Logic: RS, JK, D, and T Flip-Flops – Master-Slave Flip-Flops. Registers: Shift Registers – Types of Shift Registers. | | | | | | | | | | |
| **UNIT V** | Counters: Asynchronous and Synchronous Counters - Ripple, Mod, Up-Down Counters– Ring Counters. Memory: Basic Terms and Ideas – Typesof ROMs – Types of RAMs. | | | | | | | | | | |
| **Course Outcomes** | | | | | | | | | | | |
| CO1 | Identifythelogicgatesandtheirfunctionality. | | | | | | | | | | |
| CO2 | Performnumberconversionsfromonesystemtoanothersystem | | | | | | | | | | |
| CO3 | Understandthe functionsofcombinationalcircuits | | | | | | | | | | |
| CO4 | Performnumberconversions. | | | | | | | | | | |
| CO5 | PerformCounterdesign andlearnitsoperations. | | | | | | | | | | |
| **Text Book** | | | | | | | | | | | |
| 1 | D.P.LeachandA.P.Malvino,*DigitalPrinciplesandApplications*–TMH – FifthEdition – 2002. | | | | | | | | | | |
| **Reference Books** | | | | | | | | | | | |
| 1. | V.RajaramanandT.Radhakrishnan,*DigitalComputerDesign*,Prentice Hallof India, 2001 | | | | | | | | | | |
| 2. | M.MorisMano,*DigitalLogicandComputerDesign*,PHI,2001. | | | | | | | | | | |
| 3. | T.C.Bartee,*Digital Computer Fundamentals*, 6$^{th}$Edition,Tata McGraw Hill,1991. | | | | | | | | | | |
| | | | | | | | | | | | |

| CC | | Allied | L | T | P | C | H/W |
|---|---|---|---|---|---|---|---|
| **Coursecode:** | **23BCEAP1** | **DIGITAL ELECTRONICS LAB** | **2** | **-** | **-** | **2** | **2** |

| | |
|---|---|
| **Objectives** | ● To Understand the Digital Electronics Practically<br>● To know how to solve gates and other functions. |

1. AND, OR and NOT Gate using TruthTable
2. Universality of NAND& NORgates.
3. VerificationofBooleanlawsusingNANDgates(Associative,Commutative&Distributive Laws)
4. VerificationofBooleanlawsusingNORgates(Associative,Commutative&DistributiveLaws)
5. Sum of Products using NAND gates and Product of Sums using NORGates.
6. 4-bitbinaryparalleladderandSubtractorIC7483
7. CounterusingIC7473
8. Study of RS, D,T and JK Flip-Flops with IC's.
9. Study of Encoder &Decoder.
10. Study of Multiplexer &De-Multiplexer.
11. Half and Full Adder using Simple & NAND Gates.
12. Half and Full Subtractor using Simple &NAND Gates.

| | |
|---|---|
| **Outcomes** | ● Studentswereabletosolvesimplegatefunctions.<br>● StudentswereabletosolveandDesigncircuitsusingIC. |

| Subject Code | Subject Name | Category | L | T | P | S | Credits | Inst. Hours | Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | CIA | External | Total |
| 23BCEA2 | Resource Management Techniques | Allied | 3 | - | - | - | 3 | 3 | 25 | 75 | 100 |

| Course Objective | |
|---|---|
| CO 1 | Describe the fundamental concepts of operations research and linear programming concepts. |
| CO 2 | Understand the mathematical formulation and optimality test. |
| CO 3 | Describe the concept of transhipment problem and assignment problem. |
| CO 4 | Classify the sequencing problems. |
| CO 5 | Demonstrate the use of network scheduling by PERT/CPM. |

| | Details | No. of Hours |
|---|---|---|
| UNIT I | **Basics of Operations Research:** Introduction – Scope of Operations Research – Phases of Operations Research -**Linear Programming:** Introduction – Formulation of LP Problems – **Graphical Method:** Procedure for Solving LPP by Graphical Method. | 6 |
| UNIT II | **Transportation Problem:** Introduction – Mathematical Formulation – Definitions – Optimal Solution – North-West Corner Rule – Least Cost or Matrix Minima Method – Vogel's Approximation Method – Optimality Test – MODI Method. | 6 |
| UNIT III | **Transhipment and Assignment Problems:** Introduction – Transhipment Problem – Assignment Problem – Hungarian Method Procedure – Unbalanced Assignment Problem- Maximization in Assignment Problem. | 6 |
| UNIT IV | **Sequencing Problems:** Introduction – Definition – Terminology and Notations – Principal Assumptions – Type I: Problems with n Jobs through Two Machines – Type II: Processing n Jobs through Three Machines A, B, C – Type III: Problems with n Jobs and k Machines – Type IV: Problems with 2 Jobs through k Machines. | 6 |
| UNIT V | **Network Scheduling by PERT/CPM:** Introduction - Basic Terms - Common Errors - Rules of Network Construction - Numbering the Events (Fulkerson's Rule) - Time Analysis – Critical Path Method (CPM). | 6 |
| | **Total** | **30** |

| | Course Outcomes | Programme Outcome |
|---|---|---|
| **CO** | Upon completion of the course the students would be Able to: | |
| CO 1 | Remember the fundamental concepts of operations research and linear programming concepts. | PO1, PO6 |
| CO 2 | Understand the mathematical formulation and optimality test. | PO2 |
| CO 3 | Apply the concept of transhipment problem and assignment problem | PO4, PO7 |
| CO 4 | Analyze the sequencing problems. | PO6 |
| CO 5 | Understand the use of network scheduling by PERT/CPM. | PO7, PO8 |
| **Text Book** | | |
| 1 | S.D. Sharma, Operations Research (Theory, Method & Applications) - Kedar Nath Ram Nath & Co – 1997. | |
| **Reference Books** | | |
| 1. | Hamdy A. Taha, Operations Research- An Introduction, Pearson Education, 10th Edition, 2019. | |
| 2 | Frederick S. Hillier, Gerald J. Lieberman et al., Introduction to operations Research, 11th Edition, TATA McGraw Hill, 2021 | |
| **Web Resources** | | |
| 1. | https://www.mooc-list.com/tags/operations-research | |

| **CO/PSO** | **PSO1** | **PSO2** | **PSO3** | **PSO4** | **PSO5** | **PSO6** |
|---|---|---|---|---|---|---|
| **CO1** | **3** | **2** | **1** | **-** | **-** | **1** |
| **CO2** | **2** | **2** | **2** | **1** | **-** | **-** |
| **CO3** | **3** | **1** | **1** | **-** | **1** | **-** |
| **CO4** | **1** | **2** | **1** | **2** | **2** | **1** |
| **CO5** | **3** | **2** | **1** | **2** | **3** | **2** |
| **Weightage of course contributed to each PSO** | 12 | 9 | 6 | 5 | 6 | 4 |

**S-Strong-3 M-Medium-2L-Low-1**

| Code | Subject Name | Category | L | T | P | S | Credits | Inst. Hours | Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | CIA | External | Total |
| **23BCEAP2** | **Resource Management Techniques Lab (Using C/C++/Python)** | **Allied Lab** | - | - | 2 | - | 2 | 2 | 25 | 75 | 100 |

| **Course Objective** | |
|---|---|
| CO1 | Describe the linear programming model. |
| CO2 | Understand the basic function of drawing the feasible region. |
| CO3 | Describe the concept of north west corner rule. |
| CO4 | Classify the Vogel's approximation rule and assignment problem. |
| CO5 | Demonstrate the job sequencing problem and network scheduling by PERT/CPM. |

| **S. No** | **List of Lab Programs** | **No. of Hours** |
|---|---|---|
| 1 | Write a program to formulate the Linear Programming Model | **30** |
| 2 | Write a Program to represent the feasible region graphically | |
| 3 | Write a program to Implement the North-West Corner Rule | |
| 4 | Write a program to implement the Vogel's Approximation method | |
| 5 | Write a program to implement the assignment problem | |
| 6 | Write a program to implement the Hungarian Method | |
| 7 | Write a program to implement Job sequencing Problem | |
| 8 | Write a program to implement the Network Scheduling by PERT/CPM | |

| **Course Outcomes** | | **Programme Outcome** |
|---|---|---|
| CO | Upon completion of the course the students would be able to: | |
| CO1 | Remember the linear programming model. | PO1, PO6 |
| CO 2 | Understand the programming basic function of drawing the feasible region | PO2 |
| CO 3 | Apply the programming concept of north west corner rule | PO4, PO7 |
| CO 4 | Analyze the Vogel's approximation rule and assignment problem. | PO6 |
| CO 5 | Know the job sequencing problem and network scheduling by PERT/CPM. | PO7, PO8 |

| **Text Book** | |
|---|---|
| 1 | S.D. Sharma, Operations Research (Theory, Method & Applications) - Kedar Nath Ram Nath & Co – 1997. |

| **Reference Books** | |
|---|---|
| 1. | Hamdy A. Taha, Operations Research- An Introduction, Pearson Education, 10th Edition, 2019. |
| 2. | Frederick S. Hillier, Gerald J. Lieberman et al., Introduction to operations Research, 11th Edition, TATA McGraw Hill, 2021 |

| **Web Resources** | |
|---|---|
| 1. | https://www.mooc-list.com/tags/operations-research |

**Mapping with Programme Outcomes:**

| **CO/PSO** | **PSO1** | **PSO2** | **PSO3** | **PSO4** | **PSO5** | **PSO6** |
|---|---|---|---|---|---|---|
| **CO1** | **3** | **2** | **1** | **-** | **-** | **1** |
| **CO2** | **2** | **2** | **2** | **1** | **-** | **-** |
| **CO3** | **3** | **1** | **1** | **-** | **1** | **-** |
| **CO4** | **1** | **2** | **1** | **2** | **2** | **1** |
| **CO5** | **3** | **2** | **1** | **2** | **3** | **2** |
| **Weightage of course contributed to each PSO** | 12 | 9 | 6 | 5 | 6 | 4 |

**Strong-3 M-Medium-2 L-Low-1**

| Subject Code | Subject Name | Category | L | T | P | S | Credits | Inst. | Marks | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | CIA | External | Total |
| **23BCEA3** | Markup and Scripting Languages | **Allied** | | T | - | - | 3 | 3 | 25 | 75 | 100 |
| **Learning Objective** | | | | | | | | | | | |
| LO1 | Learn scripting language to validate web page forms | | | | | | | | | | |
| LO2 | Learn the basics of HTML, DHTML, XML, CSS, JavaScript, AJAX | | | | | | | | | | |
| | **Contents** | | | | | | | | | | |
| **UNITI** | **HTML:** HTML-Introduction-tag basics- page structure-adding comments working with texts, paragraphsandlinebreaks.Emphasizingtest-headingandhorizontalrules-list-fontsize,face andcolor-alignment-links-tables-frames | | | | | | | | | | |
| **UNITII** | **Forms&ImagesUsingHtml:** Graphics: Introduction-How to work efficiently with images in web pages, image maps, GIF animation, adding multimedia, data collection with HTML forms textbox, password, list box, combo box, text area, tools for building web page front page. | | | | | | | | | | |
| **UNITIII** | **XML & DHTML:** Cascading style sheet (CSS)-what isCSS-Whywe use CSS-addingCSSto your web pages-Grouping styles-extensible markup language (XML). Dynamic HTML: Documentobjectmodel(DCOM)-AccessingHTML &CSSthroughDCOMDynamiccontent styles& positioning. | | | | | | | | | | |
| **UNITIV** | **JavaScript:**JavaScript:Introduction,Client-SideJavaScript,Server-SideJavaScript, JavaScriptObjects,JavaScriptSecurity,Operators,ConditionalandLoopingStatements-Break, continue, User Defined Function. Array, Date, Math, Number, Object, String, regExp. | | | | | | | | | | |
| **UNITV** | Document and its associated objects: document, Link, Area, Anchor, Image, , Layer . Events andEventHandlers:GeneralInformationaboutEvents,DefiningEventHandlers,event.**AJAX** :Introduction,advantages&disadvantages,Purposeofit,ajaxbasedwebapplication,alternatives of ajax. | | | | | | | | | | |
| **CourseOutcomes** | | | | | | | | | | | |
| CO1 | DevelopandpublishWebpagesusingHypertextMarkupLanguage(HTML). | | | | | | | | | | |
| CO2 | OptimizepagestylesandlayoutwithCascadingStyle Sheets(CSS). | | | | | | | | | | |
| CO3 | Analyze andapplytherole of languagesto create acapstone | | | | | | | | | | |
| CO4 | Developwebsitesusingclient-sidewebprogrammingslanguageslikeHTML,DHTML,CSS, XML, JavaScript, and AJAX. | | | | | | | | | | |
| CO5 | Create webapplicationsusingformsandvalidation offormfields | | | | | | | | | | |
| **TextBook** | | | | | | | | | | | |
| 1 | MASTERINGHTML,CSS&JavaScriptWebPublishing-2016byLaura Lemay(Author), Rafe Colburn (Author), Jennifer Kyrnin (Author) | | | | | | | | | | |
| 2 | HTML,CSS,andJavaScriptAllinOne–2020byJulieC.Meloni(Author),JenniferKyrnin (Author) | | | | | | | | | | |
| 3 | WebDesign With HTML&CSS :HTML&CSSCompleteBeginner's Guide-2021 byPrem Kumar (Author) | | | | | | | | | | |

| CC | | Allied | L | T | P | C | H/W |
|---|---|---|---|---|---|---|---|
| **Coursecode:** | 23BCEAP3 | **Markup and Scripting Languages LAB** | | **-** | **P** | **2** | **2** |
| **Objectives** | ➢ LearnwebpageimplementationusingbasicandadvancedHTML<br>➢ LearnFormsonthewebpageandformvalidationusingclient-side scripting | | | | | | |

1. Design a web page using different text formatting tags.
2. Designawebpagewithlinkstodifferentpagesandallownavigationbetweenweb pages.
3. Design a web page demonstrating all Style sheet types.
4. Design a web page with Image maps.
5. Design a web page demonstrating different semantics.
6. Design a web page with different tables.
7. Design a webpage with a form that uses all types of input controls.
8. Design a web page embedding with multimedia features.
9. Write a JavaScript program to find the factorial value.
10. Write a Java Script program to print the Fibonacci series.
11. Design a form and validate all the controls placed on the form using Java Script.
12. Write a JavaScript program to display all the prime numbers between1and100.
13. Write a JavaScript program to accept a number from the user and display the sum of its digits.
14. WriteaprograminJavaScripttoacceptasentencefromtheuseranddisplaythenumberof words in it. (Do not use split () function).
15. Write a javascript program to design simple calculator.

**CourseOutcomes:**

| CO-NO. | COURSEOUTCOMES |
|---|---|
| CO-1 | StudyandImplementWebPagesusingBasicandAdvancedHTML |
| CO-2 | DifferentiatebetweenfunctionalitiesofBasicCSSandAdvancedCSS |
| CO-3 | Implementbasic JavaScript. |
| CO-4 | Developprogramusingbasic functionsinJavascriptand XHTML |
| CO-5 | Create webapplicationsusingformsandvalidation offormfields |

| Subject Code23BCEA4 | Subject Name | Category | L | T | P | S | Credits | Inst. | MARKS | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | CIA | External | Total |
| | **Operating system** | **ALLIED** | 3 | - | - | - | 3 | 3 | 25 | 75 | 100 |

| | |
|---|---|
| **Objectives** | ➢ Understand the basic components of Operating Systems and their interactions.<br>➢ Understand the basics of Process Management, Memory Management , Deadlock Management and File Systems. |
| **Unit –I** | Introduction: What is an operating system? History of operating system, computer hardware, different operating systems, operating system concepts, system calls, operating system structure.<br>Processes and Threads: Processes, threads, interprocess communication, scheduling, IPC problems. |
| **Unit – II** | Memory Management: No memory abstraction, memory abstraction: address spaces, virtual memory, page replacement algorithms, design issues for paging systems, implementation issues, segmentation.<br>File Systems: Files, directories, file system implementation, file-system management and optimization, MS-DOS file system, UNIX / Linux file system, CD ROM file system. |
| **Unit – III** | Deadlocks: Resources, introduction to deadlocks, the ostrich algorithm, deadlock detection and recovery, deadlock avoidance, deadlock prevention, issues.<br>Case Study: Overview of Linux, Linux Goals , Interfaces to Linux ,The Shell , Linux Utility Programs , Kernel Structure. Android and Google - History of Android - Design Goals -Android Architecture - Linux Extensions -Android Applications.History of Windows-MS-DOS-based Windows,NT-based Windows , Modern Windows. |
| **Unit – IV** | Linux :Basic features, advantages, installing requirement, basic architecture of Linux system. Commands for files and directories cd, cp, mv, rm, mkdir,more, less, creating and viewing files, using cat, file comparisons, View files, disk related commands, checking disk free spaces, Essential linux commands. |
| **Unit – V** | Understanding shells, Processes in linux – scheduling of processes at command, batch commands, kill, ps, who,sleep, Printing commands, grep, fgrep, find, sort,cal, banner, touch, file related commands – ws, sat, cut, grep, dd, etc.<br>Mathematical commands – bc, expr, factor,units. Vi, joe, vim editor.<br><br>Shell programming: Shell programming basic, various types of shell, shell programming in bash, conditional and looping statements, case statements, parameter passing and arguments, shell variables, shell keywords, use of grep in shell, awk programming. |

**Books for Reference:**

*Modern Operating Systems*-Andrew S. Tanenbaum,Herbert Bos- 4th Edition-Pearson Prentice Hall
*Operating Systems Concepts*-Abraham Silberschatz-Peter Baer Galvin- Greg Gagne-8th Edition
*Operating Systems Internals And Design Principles*- William Stallings-Eighth Edition
Linux Command Line and Shell Scripting Bible-Christine Bresnahan and Richard BLUM

| | |
|---|---|
| **Outcomes** | ➢ Explain the structure and functions of operating systems along with their components, types and working.<br>➢ Elaborate the system calls for process management and file management.<br>➢ Make use of appropriate Linux commands. |

| Course Code: 23BCEAP4 | Allied | T/P | C | H/W |
|---|---|---|---|---|
| | **Operating System Lab** | **P** | **2** | **2** |

1. Linux commands: Working with Directories:

a pwd, cd, absolute and relative paths, ls, mkdir, rmdir

b file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod

2. Linux commands: Working with files:

a ps, top, kill, pkill, bg, fg

b grep, locate, find, locate

c date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which

d Compression: tar, gzip

3. Windows (DOS) Commands

a Date, time, prompt, md, cd, rd, path.

b Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move.

c Diskcomp, diskcopy, diskpart, doskey, echo

d Edit, fc, find, rename, set, type, ver

4. Write a Shell script that displays list of all the files in the current directory to which the user has read, write and execute permissions.?

5. Write a shell script that takes argument and reports on whether it is directory, a file, or something else.

6. Write a Shell script to list all of the directory files in a directory.

7. Write a awk script to find the number of characters, words and lines in a file?

8. Write a shell script to perform the following string operations:

(a) To extract a sub-string from a given string

(b) To find the length of a given string

9. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

10. Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory.

11. Write a Shell script to find factorial of a given integer.

12. Write a Shell script to find biggest no from two nos.

13. Write a Shell script to find the give no is odd or even.

14. Installation of Linux operating system on virtual machine. 15. Installation of Windows operating system.