Shiv Chhatrapati Shikshan Sanstha's

Rajarshi Shahu Mahavidyalaya, Latur (Autonomous)

Curriculum

2022-2023

B. C. A.

(CC/AECC/SEC/GE)

UG First Year Semester I & II

Under CBCS

Three Year Degree Programme in B. C. A.

(Six Semester Course)

Syllabi approved by the Board of Studies in

B.Sc.C.S. with effect from June, 2022

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
UG I
Semester Pattern
Semester I & II

Semester : I

Course Code	Course Title	Lectures		Marks		Credits
		Per Week	Total	Internal	End Semester	
Ability Enhancement Core Course (AECC)						
U-COE-101	Communicative English-I	3	40	20	30	2
Core Course (CC)						
U-BAC-179	Basics of Computer	4	60	20	30	3
U-PSP-180	Problem Solving Through C	4	60	20	30	3
U-WDE-181	Web Designing	4	60	20	30	3
U-STF-182	Statistical Foundation	4	60	20	30	3
Practical based on Core Course						
U-LAC-183	LAB. COURSE – I (Basics of Computer)			--	50	2
U-LAC-184	LAB. COURSE – II (C programming)			--	50	2
U-LAC-185	LAB. COURSE – III (Web Designing)			--	50	2
U-LAC-186	LAB. COURSE – IV (Statistics)			--	50	2
Total Marks and Credits (Sem-I)					450	22

Semester: II

Course Code	Course Title	Lectures		Marks		Credits
		Per Week	Total	Internal	End Semester	
Ability Enhancement Core Course (AECC)						
U-COE-201	Communicative English-II	3	40	20	30	2
Core Course (CC)						
U-OSC-279	Operating System Concepts	4	60	20	30	3
U-DST-280	Data Structures	4	60	20	30	3
U-OUC-281	OOP Using C++	4	60	20	30	3
U-DIL-282	Digital Logic	4	60	20	30	3
Generic Elective (GE)						
U-MOE-235	Moral Education	--	40	20	30	NCBC
Practical Based on Core Course						
U-LAC-283	LAB. COURSE – V (Operating System Concept)			--	50	2
U-LAC-284	LAB. COURSE – VI (Data Structures)			--	50	2
U-LAC-285	LAB. COURSE – VII (OOP using C++)			--	50	2
U-LAC-286	LAB. COURSE – VIII (Digital Logic)			--	50	2
Total Marks and Credits (Sem-II)					450	22

Note: Students can opt any online course (SWAYAM, NPTEL, MOOCS) will be given additional credits in total credits.

SEMESTER – I

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester I)

Course Title – Communicative English-I   **Course Code-** U-COE-101

Max. Marks: 50   **Credits: 02**

Total Lectures: 60

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester I)

Course Title – Basics of Computer
Max. Marks: 50
Total Lectures: 60

Course Code- U-BAC-179
Credits: 03

Prerequisite Course(s): Bridge Course

Introduction:

This course introduces computer concepts, including fundamental functions and operations of the computer. Topics include identification of hardware components, basic computer operations, and use of software applications like Word processor, spread sheet solution and presentation tools.

Learning Objectives:
The course is designed to aim at imparting
- fundamental concepts of computer system
- a common individual to be a part of computer users list by making them digitally literate
- the computer organization, types of memory used and operating environment of GUI OS
- the knowledge of word processor, spreadsheet solution and presentation techniques
- Independent and self-motivated study in information technology.

Course Outcomes:
On successful completion of the course the students will be able to:
- Understand working of Hardware and Software and the importance of operating systems
- Prepare his / her personal / business letters.
- Manage and maintain small account in the form of spreadsheet
- Prepare and present their creativity digitally and enjoy the world of information technology

---

SYLLABUS
UNIT I:
Introduction, Basic structure, ALU, Memory, CPU, Generations of computer, Classification of computers, **Input Devices:** Keyboard, Point & Draw Devices, Data Scanning devices, Digitizer, Electronic Card Reader, Voice Recognition devices, **Output Devices:** Monitor, Printer, Plotter, Screen image projector, voice response system

UNIT II:
Memory: Cache memory, Primary Memory: RAM, ROM and types of ROM, Secondary Memory, Storage devices: Tape, FDD, HDD, CD ROM, **Computer Software:** Definition of software**,** Types of software**,** Compilers, Interpreters, Assemblers, Linkers, Loaders **Introduction to Operating System:** Introduction, Function of OS, Types of OS, Files and directories

UNIT III:
MS-Word - Starting MS-Word, MS-Word Screen and its Components, **Working with Files**, **Working with Text:** Formatting, Moving, copying and pasting text Styles, **Lists:** Bulleted and

numbered lists, **Table Manipulations**, **Graphics:** Adding clip Art, add an image from a file, Spelling and Grammar, AutoCorrect, **Page formatting:** Header and footers, page numbers, Protect the Document, Mail Merge

MS-Excel- Starting MS-Excel, Basics of Spreadsheet, modifying a worksheet – Moving through cells, adding worksheets, rows and columns Resizing rows and columns, selecting cells, Moving and copying cells, freezing panes - Formatting cells: Dates and times, Auto formatting. Formula and Functions. Linking worksheets - Sorting and Filling, Auto fill, Graphics – Adding clip art, add an image from a file, Charts – Using chart Wizard

UNIT IV:

MS-Power Point -Create a Presentation from a template- Working with Slides – Insert a new slide, applying a design template, changing slide layouts -Slides: Reordering slides, hide slides, Create a Custom slide show. Adding Content – Resizing a text box, Text box properties, delete a text box - Video and Audio effects, Adding clip art, adding an image from a file

Reference Books:

1. Fundamentals of Computer Science-P.K. SINHA.
2. Fundamentals of Computer Science-V RAJARAMAN.
Pradeep K. Sinha and Priti Sinha: Computer Fundamentals (Sixth Edition), BPB Publication

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester I)

Course Title – Problem Solving Through C	**Course Code: U-PST-180**
Max. Marks: 50	**Credits: 03**
Total Lectures: 60	

Prerequisite Course(s): None

Introduction:

The syllabus is prepared with the view of preparing the B.C.A. Graduates capable of writing readable C programs to solve computational problems. The course content is decided to cover the essential programming fundamentals which can be taught within the given slots in the curriculum.

Learning Objectives:
The course is designed to provide
- Knowledge about algorithm/flowchart to find solution on an problem
- writing C programs with branching and looping statements, which uses Arithmetic, Logical, Relational operators
- an working of arrays, structure or union for storing the data
- modularity to the programs written in C.
- use pointers for in various concepts like arrays, structures and functions.

Course Outcomes:
 On successful completion of the course the students will be able to:
- Analyze a computational problem and develop an algorithm/flowchart to find its solution
- Develop readable C programs with branching and looping statements, which uses Arithmetic, Logical, Relational operators
- Write readable C programs with arrays, structure or union for storing the data to be processed
- Implement modularity in the C programe
- Write C programs which will make use of use pointers for array, functions, structures etc.

SYLLABUS

UNIT 1:

Basics Programming concepts- Algorithm, Flowchart, **Overview of C:** Introduction, Importance of C, Sample 'C' programs, Basic structure of C programming, Executing a 'C' program, **Data Types in C:** C tokens, Keywords, Identifiers, Constants, Variables, Data types, Declaration of variables, Assigning values to variables, Defining symbolic constants, Simple Programs, Input and Output statements: Input and Output statements, Reading character, Writing character, formatted input, formatted output statements.

UNIT 2:

Operators and Expressions: Arithmetic operators, Relational operators, Logical operators, Assignment operators, Increment and Decrement operators, Conditional operators, Bitwise

operators, Special operators, Type Conversion in expressions, Operator precedence, Mathematical functions.

Branching and Looping: Simple 'if' statement, Simple, Nested, Ladder 'if–else' statement. The 'Switch' statement, break, continue, goto, exit statement, 'While' statement, 'do-while' statement, 'for' statement, Simple programs on branching and looping.

UNIT 3:

Arrays: Introduction, One dimensional, Two dimensional and Multidimensional arrays, Initialization of arrays, Handling of Character Strings: Declaring and Initializing string variables, reading string from terminal, writing string to screen, Arithmetic operations on characters, putting strings together, Comparison of two strings, string handling functions: strlen, strcpy, strcat, strcmp.

UNIT 4:

Pointers: Understanding Pointers, Accessing the address of variables, Declaring and initializing pointers, Accessing a variable through its pointers. **Structure and Union:** Introduction, Defining Structure, declaring structure variables and structure members, arrays as structure, arrays within structure, Union. **Functions:** Definition of function. Return values and their types, Function calls, Function declaration, Categories of function explanation with example, Nesting of function, Recursion, Command line argument,

References

1. E. Balgurusamy: Programming in ANSI C (TMH)
2. Kamthane: Programming with ANSI and TURBO C (Pearson Education)
3. V. Rajaraman: Programming in C (PHI – EEE)
4. Kernighan & Ritche: The C Programming Language (PHI)
5. Yashwant Kanitkar: Let us C
6. P.B. Kottur: Programming in C (Sapna Book House)

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester I)

Course Title – Web Designing   **Course Code-** U-WDE-181
Max. Marks: 50   **Credits: 03**
Total Lectures: 60

Prerequisite Course(s): Bridge Course

Introduction:

The syllabus is prepared with the view of preparing the B.C.A. Graduates capable of designing simple web pages which will increase the confidence among them regarding web site development. The course content is decided to cover the essential web page fundamentals, CSS and java script. The use of javascript helps them to enhance the knowledge about TypeScript, AngularJS in their future studies.

Learning Objectives:
The course is designed to
- Provide necessary information about various tags used to design and develop simple web documents/ web site.
- Knowledge about creation of web pages
- Making use of CSS for formatting of web document
- Knowledge of JavaScript

Course Outcomes:
On successful completion of the course the students will be able to:
- Create simple static web site.
- Apply various formatting effects with the help of CSS.
- Handle events/ different functions using HTML and javascript
- Write simple client side validations using java script

SYLLABUS

UNIT I

Web Publishing: Phases of website development**,** Web browser**,** Cross browser testing**,** how to publish a website**,** Web server**,** WWW**,** URL **HTML Document:** Overview, Rules & Guidelines of HTML**,** Structure of HTML document **The Markup Tags:** Basic HTML Tags, Physical style tags, Paragraphs, Lists, Font, HR, Heading levels, Center, Div, Span, Address

UNIT II
Linking: Hyperlinks**,** Mailto anchor **Inserting images:** Image file formats**,** <img> tag with its attributes, Images as background**,** Internal and External image, Image map: server side and client side image maps**,** Image as hyperlink, Working with Hyperlinks, Images and Multimedia **Adding multimedia Elements:** Audio file formats**,** Adding audio in html document**,** Video file formats**,** Adding video in html document **Tables:** <table> tag with its attributes**,** Rowspan, colspan, Frames: Overview of frames, <frameset> tag and all attributes, Frame targeting, Floating frames,

Working with Forms and controls Forms, <form> tag with its attributes, Form controls, <input> tag with its attributes.

UNIT III
Concept of CSS, Creating Style Sheet, CSS Properties, CSS Styling (Background, Text Format, Controlling Fonts), Working with block elements and objects, Working with Lists and Tables, CSS Id and Class, Box Model (Introduction, Border properties, Padding Properties, Margin properties), CSS Advanced (Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector), Color, Creating page Layout and Site Designs.

Unit IV
Javascript: Adding script to document**,** working with local and global variable**,** Javascript, Data types, Operators, Control statements, Looping statements, **Document object model (DOM),** DOM Objects (window, n**avigator, history, location),** Predefined functions, math & string functio**ns, Array in Java scripts,** Event handling in Java script**.**

Reference Books:
1. Web Publishing by Monica D'Souza
2. HTML 5 in simple steps, Kogent Learning Solutions Inc., Dreamtech Press.
3. A beginner's guide to HTML, NCSA
4. HTML & CSS: The Complete Reference, Fifth Edition, by Thomas Powell
5. HTML and JavaScript – Ivan Bayross
6. Mastering HTML, CSS & Javascript Web Publishing

Reference websites:
1. www.w3schools.com
2. www.tutorialspoint.com

Course Title – Statistical Foundation	**Course Code: U-STF-182**
Max. Marks: 50	**Credits: 03**
Total Lectures: 60	

Prerequisite Course(s): None

Introduction: This course is design acquaint students with various statistical methods. The aim of the course is to cultivate statistical thinking among students. This will help students to prepare for future courses having quantitative components.

Learning Objectives:
The course is designed to
- Acquire a strong foundation in Statistical analytics
- acquaint students with various statistical methods.
- cultivate statistical thinking among students.
- apply the fundamental principles, concepts and methods.

Course Outcomes:
 On successful completion of the course the students will be able to:
- Understand and appreciate descriptive statistics.
- Understand the concepts of probability and random variables.
- Apply fundamental techniques of data representation and graphical representation of data.
- Work with range of data also find the variance and coefficient depends upon available data.

SYLLABUS

UNIT I

INDTRODUCTION AND GRAPHICAL REPRESENTION: Definitions of Statistics, Importance of statistics., Advantages and Limitations, Scope of Statistics, Collection of Data, Types of Data, Attributes and variables, Construction of Frequency, Cumulative and Relative, Frequency distributions, **Graphical representation of Frequency distribution:** Histogram, Frequency Polygon, Frequency Curve and Cumulative Frequency curves (Ogive curves), Diagrammatic representations: Simple bar, Subdivided bar, Pie diagrams.

UNIT II

MEASURES OF CENTRAL TENDENCY: Concept of central tendency, **Arithmetic Mean:** Definition, Formulae and computation for ungrouped and grouped, data, Merits and Demerits. **Median:** Definition, Formulae and Computation for ungrouped and grouped data, Merits and Demerits., **Quartiles:** Definition, Formulae and Computation for ungrouped and grouped data. **Mode:** Definition, Formulae and Computation for ungrouped and grouped data, Merits and Demerits.

UNIT III

MEASURES OF DISPERSION: Concept of Dispersion., **Range:** Definition, Formulae and Computation for ungrouped and grouped data., **Standard Deviation:** Definition, Formulae and Computation for ungrouped and grouped data., **Variance:** Definition, Formulae and Computation for ungrouped and grouped data., **Coefficient of variance:** Definition, Formulae and Computation for ungrouped and grouped data

UNIT IV

CORRELATIONS AND TIME SERIES: Definition of Correlation, Types of Correlation, Karl Pearson's coefficient of correlations for ungrouped data and problems, Definition and components of time series, Measures of trends, Moving average method and least square method and problems.

REFERENCE BOOKS:

1. Fundamentals of Statistics by A.M. Gun, M. K. Gupta and B. Dasgupta

2. Statistical Methods by S.P. Gupta.

3. Business Statistics by S. Shaha

4. Modern Elementary Statistics by J.E. Freund

5. Fundamentals of Statistics by S C Gupta.

6. Fundamentals of Applied Statistics by Gupta and Kapoor.

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester I)

Course Title – Lab Course – I (Basics of Computer)   **Course Code: U-LAC-183**
Max. Marks: 50   **Credits: 02**

Learning Objectives:
The course is designed to aim at imparting
- fundamental concepts of computer system
- a common individual to be a part of computer users list by making them digitally literate
- the computer organization, types of memory used and operating environment of GUI OS
- the knowledge of word processor, spreadsheet solution and presentation techniques
- Independent and self-motivated study in information technology.

Course Outcomes:
On successful completion of the course the students will be able to:
- Understand working of Hardware and Software and the importance of operating systems
- Prepare his / her personal / business letters.
- Manage and maintain small account in the form of spreadsheet
- Prepare and present their creativity digitally and enjoy the world of information technology

Proposed Practical List

1. Study of elements of Windows OS
2. Study of Windows explorer: Creating and moving files and folders, Create a new folder and a file, Copy the created file to a new folder. d. Rename, Copy and Delete the created file and folder.
3. Create an application and prepare a neat curriculum Vitae for applying a job in a company. Apply Page format and Paragraph format to the above document.
4. Create a Mathematical question paper using, at least five equations, a. With fractions, exponents, summation function b. With at least one m*n‟ matrix c. Basic mathematical and geometric operators. d. Use proper text formatting, page color and page border.
5. Create a flowchart using, (a. Proper shapes like ellipse, arrows, rectangle, and parallelogram.
6. Create a table using table menu. Use proper table border and color. Insert proper content into the table with proper text formatting.
7. Create a table using two columns, (a. The left column contains all the short-cut keys and right side column contains the function of the short-cut keys)
8. Create two letters with the following conditions in MS Word and find the difference. (a. Write a personal letter to your friend using at least 100 words and two paragraphs. The date must be in top-right corner.
9. Create a table "Student result" with following conditions. (a. The heading must contain, Sl. No., Name, Mark1, Mark2, Mark3, Total, average and result with manual entry. b. Use formulas for total and average.).

10. Create a sales table using any suitable data and draw the bar-graph to compare the sales of the items for three years: (Sample Data given below)

Item	2019	2020	2021
Item1	1000	1200	1100
Item2	950	1200	1150
Item3	1100	1300	1250
Item4	1200	1250	1300
Item5	1300	1000	1440

11. Create a power-point presentation with minimum 5 slides. (a. The first slide must contain the topic of the presentation and name of the presenter. b. Must contain at least one table. c. Last slide must contain "thank you".)

12. Create a power-point presentation with minimum 10 slides having animation and images

Course Title – Lab Course – II (Problem Solving Through C) **Course Code: U-LAC-184**
Max. Marks: 50   **Credits: 02**

Learning Objectives:
The course is designed to provide
- Knowledge about algorithm/flowchart to find solution on an problem
- writing C programs with branching and looping statements, which uses Arithmetic, Logical, Relational operators
- an working of arrays, structure or union for storing the data
- modularity to the programs written in C.
- use pointers for in various concepts like arrays, structures and functions.

Course Outcomes:
 On successful completion of the course the students will be able to:
- Analyze a computational problem and develop an algorithm/flowchart to find its solution
- Develop readable C programs with branching and looping statements, which uses Arithmetic, Logical, Relational operators
- Write readable C programs with arrays, structure or union for storing the data to be processed
- Implement modularity in the C programe
- Write C programs which will make use of use pointers for array, functions, structures etc.

Proposed Practical List

1. Write a C program to perform basic arithmetic operations.

2. Write a C program to find the area of a circle.

3. Write a C program to convert temperature in Fahrenheit to Celsius and Celsius to Fahrenheit.

4. Write a C program to find the GCD and LCM of two integer numbers.

5. Write a C program to check whether the given integer is even or odd using if condition statement.

6. Write a C program to accept three integers and determine largest among them using if statement.

7. Write a C program to simulate a simple calculator with addition, subtraction, multiplication, division and it should display the error message for division for 0 using switch case.

8. Write a C program to print number from 1 to 20 which are divisible by 5 and display their sum and count using for loop.

9. Write a C program to reverse a given integer number and check whether the number is palindrome or not using while loop.

10. Write a C program to check whether given number is prime or not using while loop.

11. Write a C program the pattern given below using nested for loop

```
                    1
                1       2
            1       2       3
        1       2       3       4
    1       2       3       4       5
```

12. Write a C program to read 10 integers into an array and find sum and average of all numbers.

13. Write a C program to find the addition of two matrices.

14. Write a C program to calculate the factorial of a number using function.

15. Write a program to perform addition of two numbers using command line argument.

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester I)

Course Title – Lab Course – III (Web Designing)	**Course Code: U-LAC-185**
Max. Marks: 50	**Credits: 02**

Learning Objectives:

The course is designed to

- Provide necessary information about various tags used to design and develop simple web documents/ web site.
- Knowledge about creation of web pages
- Making use of CSS for formatting of web document
- Knowledge of JavaScript

Course Outcomes:

On successful completion of the course the students will be able to:

- Create simple static web site.
- Apply various formatting effects with the help of CSS.
- Handle events/ different functions using HTML and javascript
- Write simple client side validations using java script

Proposed Practical List

1. Prepare a survey document of five websites and find out prerequisite for designing it.
2. Design web pages for your college containing a description of the courses, departments, faculties, library etc, Make use of <a> and list tags.
3. Create your class timetable using <table> tag.
4. Create Student feedback form (use textbox, text area, checkbox, radio button, select box etc.)
5. Create a web page using frame. Divide the page into two parts with Navigation links on left hand side of page (width=20%) and content page on right hand side of page (width = 80%). On clicking the navigation Links corresponding content must be shown on the right hand side.
6. Design a web page of your home town with an attractive background color, text color, an Image, font etc. (use internal CSS).
7. Use Inline CSS to format your resume that you have created in experiment-2.
8. Use External CSS to format your class timetable as you have created in experiment-3.
9. Use External, Internal, and Inline CSS to format college web page that you have created in experiment-2 or experiment-4.
10. Write a JavaScript program to display system date.
11. Write a JavaScript program to develop simple calculator for arithmetic operations.
12. Write a JavaScript program to determine whether a given year is a leap year or not.
13. Write a JavaScript program to convert temperatures to and from Celsius, Fahrenheit.
14. Design signup form to validate username, password using Java script.

Course Title – Lab Course – IV (Statistical Foundation)	**Course Code: U-LAC-186**
Max. Marks: 50	**Credits: 02**

Learning Objectives:
The course is designed to
- Acquire a strong foundation in Statistical analytics
- acquaint students with various statistical methods.
- cultivate statistical thinking among students.
- apply the fundamental principles, concepts and methods.

Course Outcomes:
On successful completion of the course the students will be able to:
- Understand and appreciate descriptive statistics.
- Understand the concepts of probability and random variables.
- Apply fundamental techniques of data representation and graphical representation of data.
- Work with range of data also find the variance and coefficient depends upon available data.

Proposed Practical List

1. Construction of histogram for given statistical data.

2. Construction of frequency polygon for given statistical data.

3. Construction of frequency curve for given statistical data.

4. Construction of ogive for given statistical data.

5. Construction of simple bar diagram for given statistical data.

6. Construction of subdivided bar diagram for given statistical data

7. Construction of pie diagram for given statistical data

8. To compute various measure of central tendency (mean, median, mode)

9. To compute various measure of dispersion (range, standard deviation, CV)

10. To compute coefficient of correlation

SEMESTER - II

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester II)

Course Title – Communicative English – II	**Course Code: U-COE-201**
Max. Marks: 50	**Credits: 02**
Total Lectures: 60	

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester II)

Course Title – Operating System Concepts	**Course Code-** U-OSC-279
Max. Marks: 50	**Credits: 03**
Total Lectures: 60	

Prerequisite Course(s): Basics of Computer

Introduction: This course is design to provide the basic concept and functions of operating system. Through this course the students can understand the concept of process, thread, resource and memory management techniques used by operating system for smooth working of OS. Students also come to know the I/O and File management techniques utilized by Operating System.

Learning Objectives:
The course is designed to
- introduce basic concepts and functions of operating systems.
- understand the concept of process, thread and resource management.
- understand the concepts of process synchronization and deadlock.
- understand various Memory, I/O and File management techniques.

Course Outcomes:
At the end of the course student should be able to
- Understand role of Operating System in terms of process, memory, file and I/O
- management.
- Apply and analyze the concept of a process, thread, mutual exclusion and deadlock.
- Apply and analyze the concepts of memory management techniques.
- Apply and analyze different techniques of file and I/O management.

SYLLABUS

UNIT I:
Introduction: Definition of O.S., Types of O.S., O.S. as resource manager, Operating-System Structure, Operating-System Operations, Process Management, Memory Management, Storage Management, O.S. Process view, Hierarchical view, **Introduction to windows O.S.:** Introduction, History, Files and Folders, **Basics of Windows:** desktop, my computer, etc, **Features of MS-Windows:** GUI, Multitasking, Multi-user, network etc., Important files of windows, Introduction to Open-Source Operating Systems.

UNIT II:
Memory management: Single continues allocation, Introduction to multiprogramming, Partitioned Memory management, Paged memory management, demand paged memory management, Segmented Memory management

Unit III:
Processor Management: Process Concept, Process Scheduling: Basic Concepts, Scheduling Criteria State model, Job Scheduling, Process Scheduling, Multiprocessor system, Process synchronization, Deadlocks: Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention and Detection

Unit IV:

Device management: Techniques for Device management, Devise management characteristics, Channels and control units, Device allocation consideration, **Information management: S**imple file system, General model of a file system, Symbolic File System, Basic File System.

Reference Books:

1. Operating systems - A concept-based Approach: D.M Dhamdhere 3rd Edition, Tata McGraw-Hill, 2012.
2. Operating System Concepts: Avi Silberschatz, Peter Baer Galvin, Greg Gagne, 9th Edition, John wiley & Sons, Inc.

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester II)

Course Title – Data Structures	**Course Code: U-DST-280**
Max. Marks: 50	**Credits: 03**
Total Lectures: 60	

Prerequisite Course(s): Problem Solving Through C

Introduction: This course is design to acquaint students with various types data structures. The aim of the course is to cultivate analysis of data structure among the students. This will help students to analyze various problems on linear and nonlinear data structure. Through this course the students can apply the learned concepts in different domains like DBMS, Compiler Construction, data science etc.

Learning Objectives:
The course is designed to
- introduce basics of data structure.
- understand the concept of stack, queue and tree.
- teach various storage mechanisms of data.
- introduce various techniques for representation of the data in the real world.
- teach different sorting and searching techniques.

Course Outcomes:
At the end of the course student should be able to
- implement various linear and nonlinear data structures.
- handle operations like insertion, deletion, searching and traversing on various data structures.
- select appropriate sorting and searching technique for given problem.
- apply the learned concepts in various domains like DBMS, Compiler Construction.
- choose appropriate data structure for specified problem domain.

SYLLABUS

UNIT I
Data **structures**: Need of data structure, types of data structures, data structure operations, **Algorithms**: Definition and Characteristics, Time and space complexity, **Arrays:** Array terminology, characteristics of arrays, storage representation of 1D, 2D and Multi-dimensional arrays, operations on arrays

UNIT II
Stacks: Definition, Stack implementation, Operations on stack, Evaluation of arithmetic Expressions, Conversion of Expressions - Prefix, Infix and Postfix, Function Calling, Recursion-

direct & indirect recursion, Applications of Stacks, Queues: Definition, Operations on queues, Types of queues: Circular queue, Dequeue and Priority queues, Applications of queues

UNIT III
Linked lists: Concept of linked list, Operations on Linked list, Types of linked list: Singly linked list, Circular linked list, Doubly linked list, Implementation of stack and queue using linked list, Applications of linked list.

UNIT IV
Trees: Tree terminology, Traversing binary trees, Graphs: Graph terminologies, Graph representation: sequential and linked, Traversing a graph: Depth First Search and Breadth First Search **Sorting**: Selection sort, Insertion sort, Bubble sort, **Searching**: Linear search and Binary search

Reference books
1. Data Structure using C by A.M. Tanenbaum, Yecidyan lang
2. Fundamentals of Data Structures by Ellis Horowitz and Sartaj Sahni, Galgotia Publications
3. Introduction to Data Structures in C by Ashok N. Kamthane, Pearson Education
4. Theory and Problems of Data Structures by Seymour Lipschutz, Schaum's Outline Series

Online Resources :
2. http://nptel.ac.in/courses/106106127/
3. http://www.nptel.ac.in/courses/106102064

Course Title – OOP Using C++	Course Code: U-OUC-281
Max. Marks: 50	**Credits: 03**
Total Lectures: 60	

Prerequisite Course(s): Problem Solving Through C

Introduction: This course is design provide the concept of top-down approach in programming. Also it aims to provide Object Oriented Programming features which are very much important in any advanced programming language. The OOPs concepts are addressed through the C++ programming. The emphasis is given on the concept like Class, Overloading and overriding, inheritance etc.

Learning Objectives:
The course is designed to
- Describe the object-oriented programming approach in connection with C++
- Understand Object oriented concepts like data abstraction, encapsulation, inheritance, virtual function, etc.
- Understand the difference between the top-down and bottom-up approach
- Solve the real-world scenarios using top-down approach

Course Outcomes:
At the end of the course student should be able to
- solve computational problems using basic constructs like if-else, control structures, array.
- Students will be able to implement relationships between classes.
- Identify importance of object-oriented programming and difference between structured oriented and object-oriented programming features.
- Make use of objects and classes for developing programs.
- Use various object-oriented concepts to solve different problems

---

SYLLABUS

UNIT 1:

Introduction, Object Oriented Paradigm, Basic concepts of OOP: Object, Class, Data Abstraction, Encapsulation, Inheritance, Polymorphism, Overloading, Dynamic binding, Message Passing. C++ Features: The iostream class, C++ comments, C++ keywords, variable declaration, Manipulators: endl, setw, setprecision, Scope resolution operator, new and delete operators. Functions: function declaration, calling the function, function definition, passing argument to, returning value from function, pass by value, pass by reference, default arguments, overloaded functions, inline functions.

UNIT 2:

Classes & objects, class declaration, class members, class member visibility: private, public, protected. constructors, destructors, default constructor, constructor with argument, constructor with default arguments, dynamic constructors, copy constructor, member functions defined

outside the class, objects as arguments, returning objects from functions, class conversion, manipulating private data members, array as class member data, Array of objects

UNIT 3:

Operator Overloading: Defining operator overloading, overloading unary operator, overloading binary operator, arithmetic operator, comparison operators, arithmetic assignment operator, Data conversion: conversion between basic to class types, conversion between objects and basic types, conversion between objects of different classes.

UNIT 4:

Inheritance: Derived class & Base class: Specifying the derived class, accessing the base class members, derived class constructor, overriding member functions, public & private inheritance, access combinations, classes & structures, access specifies, **Level of inheritance:** Simple / Single inheritance, Multilevel inheritance, multiple inheritance, hybrid inheritance, member functions in multiple inheritance, constructors in multiple inheritance, Containership: classes within classes. **Virtual Functions**: Normal member function accessed with pointers, virtual member function accessed with pointers, dynamic binding, pure virtual functions, Friend function, this pointer, **Exception Handling:** Introduction, Exception Handling Mechanism, Handling Multiple Exceptions.

Reference Books:
1. Object Oriented Programming with C++, E. Balagurusami, Fourth Edition, Tata Mc-Graw Hill
2. Complete reference C++, Herbert Schildt, Tata McGraw Hill
3. Object Oriented Programming in Turbo C++, Robert Lafore, Fourth Edition Galgotia Publications.
4. The C++ Programming Language, Bjarne Stroustrup, Third Edition, Addison-Wesley Publishing Company

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester II)

Course Title – Digital Logic	**Course Code: U-DIL-282**
Max. Marks: 50	**Credits: 03**
Total Lectures: 60	

Prerequisite Course(s): None

Introduction: This course is design to provide the basic concept of digital electronics. The variety of gates used in building of IC's and basic building block. This course is also given basic of number systems used computers. The course also provides the knowledge of Flip-Flop and its importance in computer architecture. With the help of this course the students also make aware of microprocessor and components of microprocessor.

Learning Objectives:
The course is designed to
- Describe the object-oriented programming approach in connection with C++
- Understand Object oriented concepts like data abstraction, encapsulation, inheritance, virtual function, etc.
- Understand the difference between the top-down and bottom-up approach
- Solve the real-world scenarios using top-down approach

Course Outcomes:
At the end of the course student are able to
- Convert different type of codes and number systems which are used in digital communication and computer systems.
- Identify different types of logic families which are the basic unit of different types of logic gates.
- identify and describe introduction to digital concepts
- apply the logic in different programming logic building.
- Identify the microprocessor and components of microprocessor

---

SYLLABUS

UNIT I:
Number system and codes: Binary number system, decimal number system, octal number system, hexadecimal number system. Bases inter conversions, Binary Subtraction using 1'2 and 2's complement method, Decimal subtraction using 9's and 10's complement method, Codes: BCD, GRAY, EXCESS-3

UNIT II:
Boolean algebra and gate networks: Fundamental concepts of Boolean algebra, Ssymbol, boolean equation, truth tables of different types of gate such as inverter or NOT gate, AND gate, OR gate, NAND gate, NOR gate, X-OR gate, X-NOR gate, Basic laws of Boolean algebra and simplification of Boolean, Universal property of NAND and NOR gate

UNIT III:
Combinational logic circuit: Half adder and FULL adder, parallel binary adders, HALF subtractor and FULL subtractor, Multiplexer and demultiplexer with types & examples, Encoder and decoder with types

UNIT IV:
Sequential logic circuit with Microprocessor: Flip-flop-latches, edge triggered and level triggered flip flop with types SR flip flop, D flip flop, JK flip flip and T flip-flop, Buffer registers, modes of operation of registers (SISO, SIPO, PISO, PIPO), Introduction to microprocessors, Basic components of microprocessors.

Reference Books
1. Floyd, Thomas L: "Digital Computer Fundamentals", 3 rd Edition 1997.
2. Malvino, Pual Albert and Leach, Donald P: "Digital Computer
Fundamentals", 3rd Edition, 1995.TMH.
3. Modern digital electronics by R.P.Jain
4. Bartee, Thomas C, "Digital Computer Fundamentals" 6 th Edition, 1995.TMH

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester II)

Course Title – Lab Course-V (Operating System Concepts) **Course Code: U-LAC-283**
Max. Marks: 50   **Credits: 02**

Learning Objectives:
The course is designed to
- introduce basic concepts and functions of operating systems.
- understand the concept of process, thread and resource management.
- understand the concepts of process synchronization and deadlock.
- understand various Memory, I/O and File management techniques.

Course Outcomes:
At the end of the course student should be able to
- Understand role of Operating System in terms of process, memory, file and I/O
- management.
- Apply and analyze the concept of a process, thread, mutual exclusion and deadlock.
- Apply and analyze the concepts of memory management techniques.
- Apply and analyze different techniques of file and I/O management.

Proposed Practical List

1. Comparative study of various operating systems

2. Study of DOS internal commands

3. Study of DOS external commands

4. Study of Autoexec batch file

5. Installation of Windows O.S.

6. Working with Notepad

7. Working with Wordpad

8. Creating and managing folders

9. Sharing files and folders

10. Creating and Managing Users and authentication

11. Demonstrating scheduling algorithms.

12. Demonstrating Process algorithm.

Course Title – Lab Course – VI (Data Structures)	**Course Code: U-LAC-284**
Max. Marks: 50	**Credits: 02**

Learning Objectives:
The course is designed to
- introduce basics of data structure.
- understand the concept of stack, queue and tree.
- teach various storage mechanisms of data.
- introduce various techniques for representation of the data in the real world.
- teach different sorting and searching techniques.

Course Outcomes:
At the end of the course student should be able to
- implement various linear and nonlinear data structures.
- handle operations like insertion, deletion, searching and traversing on various data structures.
- select appropriate sorting and searching technique for given problem.
- apply the learned concepts in various domains like DBMS, Compiler Construction.
- choose appropriate data structure for specified problem domain.

Proposed Practical List

1. Implementation of Array as an data structure.
2. Implementation of insertion and deletion algorithms of stack
3. Implementation of simple queue using an array.
4. Implementation of the following
   a. Simple queue
   b. Circular queue
5. Implementation of Linked list algorithm for insertion and deletion of an item from list
6. Implementation of Binary Tree traversal algorithms
7. Recursive implementation of
   a. Factorial
   b. Fibonacci
8. Implementation of Linear Search algorithm to find an item in a list.
9. Implementation Bubble Sorting Algorithms
10. Implementation Insertion Sorting Algorithms

Course Title – Lab Course – VII (OOP Using C++)   **Course Code: U-LAC-285**
Max. Marks: 50   **Credits: 02**

Learning Objectives:
The course is designed to
- Describe the object-oriented programming approach in connection with C++
- Understand Object oriented concepts like data abstraction, encapsulation, inheritance, virtual function, etc.
- Understand the difference between the top-down and bottom-up approach
- Solve the real-world scenarios using top-down approach

Course Outcomes:
At the end of the course student should be able to
- solve computational problems using basic constructs like if-else, control structures, array.
- Students will be able to implement relationships between classes.
- Identify importance of object-oriented programming and difference between structured oriented and object-oriented programming features.
- Make use of objects and classes for developing programs.
- Use various object-oriented concepts to solve different problems

Proposed Practical List

1. Program to demonstrate I/O statements and manipulators
2. Program for scope resolution operator
3. Write a program using area() function that returns the area of a circle with given radius.
                float area(float r)
4. Write a program using min() function that returns the smallest of two integers
                int min(int x, int y)
5. Program to demonstrate simple concept of class and object concept

6. Program to demonstrate visibility modes used in class

7. Program to demonstrate constructor and its types

8. Program to demonstrate destructor

9. Program to find area and circumference of rectangle and triangle using function overloading

10. Program to demonstrate operator overloading concept

11. Program to demonstrate concept of single inheritance

12. Program to demonstrate concept of multiple inheritance

13. Program to demonstrate virtual function

14. Program to demonstrate exception handling mechanism

Rajarshi Shahu Mahavidyalaya(Autonomous), Latur
U G First Year (Semester II)

Course Title – Lab Course – VIII (Digital Logic)	**Course Code: U-LAC-286**
Max. Marks: 50	**Credits: 02**

Learning Objectives:

The course is designed to

- Describe the object-oriented programming approach in connection with C++
- Understand Object oriented concepts like data abstraction, encapsulation, inheritance, virtual function, etc.
- Understand the difference between the top-down and bottom-up approach
- Solve the real-world scenarios using top-down approach

Course Outcomes:

At the end of the course student are able to

- Convert different type of codes and number systems which are used in digital communication and computer systems.
- Identify different types of logic families which are the basic unit of different types of logic gates.
- identify and describe introduction to digital concepts
- apply the logic in different programming logic building.
- Identify the microprocessor and components of microprocessor

Proposed Practical List

1. To perform and verify the truth tables of basic gates and derived gates

2. To perform and verify the truth tables of EX-OR and EX-NOR gate

3. To perform universal property of NAND gate

4. To perform universal property of NOR gate

5. To Implement and verify the truth table of any two Boolean equation

6. To perform and verify the truth table of half adder

7. To perform and verify the truth table of half subtractor

8. To perform and verify the truth table of multiplexer

9. To perform and verify the truth table of De-multiplexer

10. To perform and verify the truth table of encoder

11. To perform and verify the truth table of Decoder