SEMESTER- II			
Allied II	**Digital Principles**		
Code: 18UCSA21	**Hrs / week : 4**	**Hrs / Semester: 60**	**Credits : 3**

Vision:

To Understand the basic concepts used in the design and analysis of digital systems .

Mission:

Acquire knowledge in Boolean functions and MSI and LSI logic circuits .

Course Outcome:

CO.No.	Upon completion of this course, students will be able to	PSO addressed	CL
CO-1	understand various number systems and boolean functions.	9	Un
CO-2	apply various methods to simplify boolean function.	4	Cr
CO-3	construct digital circuits for boolean functions with logic gates.	6	Cr
CO-4	design combinational circuits with logic gates.	6	Cr
CO-5	apply classical techniques for the logical design of combinational and sequential circuits	6	Ap
CO-6	define sequential logic circuits.	6	Re
CO-7	understand the basic operation of flip-flops.	2	Re
CO-8	understand the various registers-transfer methods .	2	Re

SEMESTER- II			
Allied II	**Digital Principles**		
Code: 18UCSA21	**Hrs / week : 4**	**Hrs / Semester: 60**	**Credits : 3**

Unit I

Binary Systems : Digital Computers and Digital Systems **–** Binary numbers – Number base conversion – Octal and Hexadecimal numbers – Complements – Binary Codes –Basic theorems and properties of boolean algebra– Boolean functions – Canonical and Standard forms – Digital Logic Gates .

Unit II

Simplification of Boolean Functions : The Map method – Two and Three variable Maps – Four Variable Map – Five and Six Variable Maps – Product of Sums Simplification – NAND and NOR Implementation – Other two-level Implementations – Don't care conditions – The Tabulation method – Determination of Prime – Implicants – Selection of Prime – Implicants

Unit III

Combinational Logic : Introduction – Design Procedure – Adders – Subtractors – Code Conversion– Multilevel NAND Circuits – Multilevel NOR Circuits – Exclusive-OR and Equivalence Functions.

Unit IV

Combinational Logic with MSI and LSI : Introduction – Binary Parallel Adder – Decimal Adder – Magnitude Comparator – Decoders – Multiplexers

Unit V

Registers and Counters: Sequential logic -Introduction – Flip-Flops -Basic Flip-Flop Circuit-Clocked RS Flip-Flop-D-Flip-Flop-JK Flip-Flop- T-Flip-Flop- Registers – Shift Registers

Text Book :

1. M. Morris Mano, Digital Logic and Computer Design, , Fourth Edition Prentice Private Limited

Chapters : 1.2-1.6, 2.3-2.5,2.7,3.1-3.11,4.1-4.5,4.7-4.9,5.1-5.6,6.1,6.2,7.1-7.3

Books for Reference:

1.Charles H.Roth, Jr. “Fundamentals of Logic Design”, 7th Edition, Jaico Publishing House, 1996.

2.Donald D.Givone, “Digital Principles and Design”, Tata McGraw-Hill, 2007.

3.Donald P.Leach and Albert Paul Malvino, Digital Principles andApplications, Seventh ed., Tata McGraw Hill Publishing Company Limited, New Delhi, 2003.

SEMESTER- II			
Allied-Practical I	**Office Automation Lab**		
Code: 18UCSAR2	**Hrs / week :3**	**Hrs / Semester: 45**	**Credits :2**

List of Practicals :

1. Type a paragraph and use various formatting.
2. Usage of Numbering, Bullets, Indents and Headers in a Word Document
3. Prepare a Calendar in a Word Document
4. Design a wedding invitation in Word Document
5. Usage of Spell Check, Find and Replace
6. Picture Insertion and Alignment
7. Use mail merge in word.
8. Prepare class time table.
9. Prepare a semester wise mark statement for a computer class of 20 students using any spreadsheet' worksheet. Total, average and rank the student marks. Give proper headings. Make the column headings bold and italic.
10. Consider the sample employee worksheet and calculate their salary. Plot it using chart.
11. Use any spreadsheet to use mathematical, statistical and logical functions
12. Use any spreadsheet to plot a chart for marks obtained by the students.

SEMESTER- I			
Core – I	**C Programming**		
Code: 18UCSC11	**Hrs / week : 4**	**Hrs / Semester: 60**	**Credits : 4**

Vision:

Understand the basic concepts of Structured programming language

Mission:

Able to design, code, test and debug an application

Course outcome:

CO No.	Upon completion of this course, students will be able to	PSO addressed	CL
CO-1	draw the flow chart for the given problem and algorithm	1	Un
CO-2	describe the various operators and library functions and to define I/O functions	3	Un
CO-3	compare and contrast loops	4	An
CO-4	implement recursion	8	Ap
CO-5	understand the concept of storage classes	9	Un
CO-6	implement different operations on arrays	3	Ap
CO-7	develop an application using pointer.	5	Cr
CO-8	develop application using structure and pointers	10	Cr

SEMESTER- I			
Core – I	**C Programming**		
Code: 18UCSC11	**Hrs / week : 4**	**Hrs / Semester: 60**	**Credits : 4**

Unit I:

Algorithms - Flow charts: Developing algorithms and flowcharts for solving simple problems.

C Fundamentals: The C Character Set - Identifiers and Keywords - Data Types –Constants– Variables and Arrays - Declarations - Expressions - Statements - Symbolic Constants.

Unit II:

Operators and Expressions: Arithmetic Operators - Unary Operators - Relational and Logical Operators - Assignment Operators - The Conditional Operator - Library Functions.

Data Input and Output: Single Character Input-The getchar Function-Single Character Output-The putchar Function-Entering Input Data-More about the scanf function-Writing output data – The printf function- The scanf Function-More about the printf function -The gets and puts Functions.

Unit III:

Control Statements: Branching: The if-else Statement-Looping: The While Statement-More Looping: The do-while Statement-Still More Looping: The for Statement-Nested Control Structures-The switch Statement-The break Statement-The continue Statement-The comma Operator-The go to Statement.

Functions: Defining a Function-Accessing a Function-Function Prototypes- Passing Arguments to a Function- Recursion.

Unit IV:

Program Structure: Storage Classes- Automatic Variables- External (Global) Variables- Static Variables

Arrays: Defining an Array-Processing an Array - Passing Arrays to Functions- Multidimensional

Arrays - Arrays and Strings.

Unit V:

Pointers: Fundamentals-Pointer Declarations- Passing Pointers to Functions- Pointers and One-Dimensional Arrays-Dynamic Memory Allocation- Operations on Pointers-

Pointers and Multidimensional Arrays -Arrays of pointers.

Structures and Unions: Defining a Structure - Processing a Structure - User Defined Data types (typedef) - Structures and Pointers - Passing Structures to Functions - Unions.

Text Book:

1. Byron Gottfried ,Programming with C ,, McGraw Hill Education (India) Private Limited, 3rd Edition .**Chapters: 2,3,4,6,7,8,9,10,11 and 12.**

Books for Reference:

1. Ashok N. Kamthane, Programming with ANSI and Turbo C, Pearsoneducation, 2006.
2. Gary.J.Bronson, A first Book of ANSI C 3rd Edition, Thomson learning 2001.
3. Kumar Agrawal, Programming in ANSI C., Tata McGraw Hill, 2006.
4. VenugopalPrasad, Programming with C, Tata McGraw Hill, 2006.
5. E. Balagurusamy, Programming in ANSI C Sixth Edition,, McGraw Hill Education (India) Private Limited, 2012.
6. S. Jaiswal, "Information Technology Today", Galgotia Publications,First Edition, 1999.

SEMESTER- II			
Core II	**C++ Programming**		
Code: 18UCSC21	**Hrs / week : 4**	**Hrs / Semester: 60**	**Credits : 4**

Vision:

Understand the basic concepts of object orient programming language

Mission:

Able to design,code,test and debug an application

Course outcome:

CO No.	Upon completion of this course, students will be able to	PSO addressed	CL
CO-1	know about object oriented features.	8	Un
CO-2	understand the various operators and i/o functions	3	Re
CO-3	write program using inline and friend function and to implement overloading constructor	3	Cr, AP
CO-4	understand array of objects and to demonstrate operator overloading	8,9	Un, AP
CO-5	compare different inheritance methods	3	An
CO-6	develop linked list	5	Cr
CO-7	understand virtual function	8	Un
CO-8	create an application using file operations	10	Cr

SEMESTER- II			
Core II	**C++ Programming**		
Code: 18UCSC21	**Hrs / week : 4**	**Hrs / Semester: 60**	**Credits : 4**

Unit I

The Big Picture: Why Do Need Object Oriented Programming- Characteristics of Object Oriented Languages - C++ and C-Laying the Groundwork.

C++ Programming Basics:Basic Program Construction - Output Using cout - Preprocessor Directives – Comments - Integer Variables- Character Variables - Input with cin- Type float- Manipulators-Variable type Summary-Type conversion-Arithmetic Operators-Library Functions.

Unit II

Function: Simple Functions - Passing Arguments to Functions -Returning Values from Functions - Reference Arguments - Overloaded Functions - Inline Functions - Default Arguments- Variables and Storage Classes - Returning by Reference.

Objects and Classes: A Simple Class - C++ Objects as Physical Objects - C++ Objects as Data Types-Constructors - Objects as Function Arguments -Returning Objects from unction-A Card Game Example-Structures and Classes-Classes, Object, and Memory-Static Class Data-What Does It All Mean?

Unit III

Arrays: Array Fundamentals - Array as Class Member Data - Array of Objects - String.

Operator Overloading: Overloading Unary Operators - Overloading Binary Operators-Data Conversion- Pitfalls of Operator Overloading and Conversion.

Unit IV

Inheritance Derived Class and Base Class - Derived Class Constructors - Overriding Member Functions-Inheritance in the English Distance Class - Class Hierarchies-Public and Private Inheritance-Levels of Inheritance-Multiple Inheritance-Ambiguity in Multiple Inheritance - Containership: Classes with Classes - inheritance and Program Development.

Pointers: Addresses and Pointers - Pointer Variables - Pointers and Array - Pointers and Functions - Pointers and Strings - Memory Management: new and delete - Pointers to Objects - A Linked List Example - Pointers to Pointers - Debugging Pointers.

Unit V:

Virtual Functions: Virtual Functions - Friend Functions - Static Functions - assignment and Copy Initialization -The *this* Pointer.

Files And Streams: Streams -String I/O -Character I/O - Object I/O -I/O with Multiple Objects -File Pointers - Disk I/O with Member Functions - Error Handling - Redirection - Command Line Arguments - Printer Output - Overloading the Extraction And Insertion Operators.

Text Book:

1. Robert Lafore, Object-Oriented Programming in C++, 4th Edition,Pearson and Dorling Kindersley Publications.

Books for Reference:

1. E.Balagurusamy, Object Oriented Programming C++ 5thEdition., Tata McGraw-Hill, 2011.
2. D.Ravichandran, Programming with C++ 2ndEdition., Tata McGraw-Hill, 2010.
3. Y.Venugopal RajkumarRavishankar, Mastering C++, Tata McGraw –Hill, 2011.
4. Debasish Jana, C++ and object oriented programming paradigm 2nd Edition, PHI publications, 2005.
5. Deiteland Deitel , C++ How to Program , Fourth Edition, Prentice Hall,2004

SEMESTER- III			
Core – III	**Java Programming**		
Code: 18UCSC31	**Hrs / week :5**	**Hrs / Semester: 75**	**Credits :4**

Unit I:

The History and Evolution of Java:

Creation of java - Operators – Control statements – Class , Methods , Inheritance

Packages and Interfaces:

Packages-Access Protection – Importing Packages- Interfaces.

Unit II:

Exception Handling:

Exception-Handling Fundamentals-Exception Types-Uncaught Exceptions-Using try and catch-Multiple catch clauses-Nested try Statements-throw-throws-finally-Java‘s Built-in Exceptions.

Multithreaded Programming:

Java Thread Model-Main Thread-Creating a Thread-Creating Multiple Threads- Using is Alive() and join ()-Thread Priorities-Synchronization - Interthread Communication-Suspending, Resuming, and Stopping Threads-Using Multithreading.

Unit III:

The Applet Class: Applet Basics -Applet Architecture - Applet Skeleton - Simple Applet Display Methods - Requesting Repainting - HTML APPLET tag - Passing Parameters toApplet.

Event Handling:

Event Handling Mechanisms - Delegation Event Model - Event Classes(The Action Event, Item Event, Key Event , Mouse Event) - Sources of Events - Event Listener Interfaces(Action Listener ,Item Listener, Key Listener, Mouse Listener) - Adapter Classes

Introducing the AWT:

AWT Classes-Window fundamentals -working with Frame Windows -Working with Graphics.

Unit IV:

Using AWT Controls:

Controls Fundamentals-Labels-Using Buttons-Applying Check Boxes-Check Box Group-Choice Controls-Using a Text Field-Using a Text Area-Understanding Layout Managers-[Flow Layout Only]-Menu Bars and Menus.

RMI:

Remote Method Invocation – Text Formatting

Unit V:

JDBC Package:

JDBC – JDBC versus ODBC – Types of JDBC drivers – Connection – Statement – PreparedStatement.

ResultSet:

Fields of ResultSet – Methods of ResultSet – Executing a query - ResultSetMetaData – DatabaseMetaData.

Database in JDBC:

Basicdatatypes in JDBC – Advanced datatypes in JDBC – fields of Statement – methods of Statement

Text Books:

1. Herbert Schildt, The Complete Reference Java™ ,8th Edition,TATAMcGRAW- HILL EDITION, 2011. Chapters: 1, 9, 10, 11,21,22,23,24,29,30,31 (Unit I,II,III,IV)
2. S.Horstmenn and Gary Cornell, Core Java2 Volume II Advanced Features, The Sun Microsystems press Java Series,2002. Chapter: 4.(Unit V)

Books for Reference:

1. Steven Holzner, Java 2 Programming Black Book, Dream Tech Press,2005.
2. Joseph O'Neil, JavaBeans Programming from the GroundUp, TMGH, New Delhi,1998
3. KathyWalrath, The J2EE Tutorial, Pearson Education Asia, 2003.

SEMESTER- I			
Core Practical I	**C Programming Lab**		
Code: 18CSCR1	**Hrs / week :5**	**Hrs / Semester:75**	**Credits :3**

List of Practicals :

1. Solve Quadratic Equation- control statements

2. Sum of Digits & reverse the number.

3. Prime number Checking

4. Sine Series evaluation

5. Binary search

6. Sorting an Array of numbers

7. Linear Searching using function

8. Sorting an array of names

9. Counting no. of vowels, consonants, words and white spaces in a line of text.

10.EB Bill using Structure

11.Exchanging values using pointers

12.Matrix multiplication using pointers

SEMESTER- II			
Core Practical II	C++ Programming Lab		
Code: 18UCSCR2	Hrs / week : 5	Hrs / Semester: 75	Credits : 4

List of Practicals :

1.Write a program in C++ to perform Area calculation using Function overloading (Minimum three functions).

2. Write a program to implement constructor overloading

3.Write a program to swap two values between two class objects using friend function.

4.Write a program in C++ to display the details of employees using array of objects.

5.Write a C++ program to overload Binary + operator which adds two complex numbers.

6.Write a C++ program to overload Relational operator = = to compare two strings.

7.Write a C++ program using class and objects to find row and column total of a matrix.

8.Using class and objects, find the sum of two matrices using pointers.

9.Write a program using multiple inheritances to process students mark list.

10.Write a program using multi level inheritance to process telephone billing.

11.Write a program in C++ using virtual function.

12.Write a program in C++ to process mark listing using binary file.

13.Write a program to open a file in output and input mode. Accept data and write to the file. Display the contents of the file.

SEMESTER- III			
Core – Practical III	**Java Programming Lab**		
Code: 18UCSCR3	**Hrs / week : 6**	**Hrs / Semester: 90**	**Credits : 4**

List of Practicals :

1. Implement Overloading Constructor and Overloading Method
2. Writing a Program to apply method Overriding concept.
3. Development of Java Packages
4. To create and implement an interface.
5. To create a thread i. Using Thread class ii. Using runnable interface
6. To design a calculator arithmetic operations.
7. To create an applet with four Checkboxes with labels and a Text area object.
8. To create a window with a checkbox group with boxes for the colors, Violet, Indigo, Yellow, Orange, Red, Blue and Green. When the button is selected the background color must change accordingly.
9. To demonstrate the use of choice box.
10. To throw the following exception, i. Negative Array Size ii. Array Index out of bounds
11. To illustrate mouse event handling.
12. To create a File menu with options new, save, and close, edit menu with options cut, copy and paste.
13. To prepare the mark sheet using JDBC.

SEMESTER- IV			
Core – Practical IV	**Python Programming Lab**		
Code: 18UCSCR4	**Hrs / week :6**	**Hrs / Semester: 90**	**Credits :4**

List of Practicals :

1. Write a Python program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
2. Write a Python Program to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria:
    a. Grade A: Percentage >=80
    b. Grade B: Percentage>=70 and <80
    c. Grade C: Percentage>=60 and <70
    d. Grade D: Percentage>=40 and <60
    e. Grade E: Percentage<40
3. Write a Python Program using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. Write a Python Program to display the first n terms of Fibonacci series.
5. Write a Python Program to find factorial of the given number.
6. Write a Python Program to find sum of the following series for n terms: 1 – 2/2! + 3/3! - - - - - - n/n!
7. Write a Python programs using String functions.
8. Write a Python program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.
9. Write a Python program to get the largest number from a list.
10. Write a Python program to get a list, sorted in increasing order by the last element in each tuple from a given list of non-empty tuples.
11. Write a Python program to remove duplicates from a list.
12. Write a Python program to create a CSV File based on user input.
13. Write a Python program to read a CSV File already created and display the contents

SEMESTER- III	
Self Study 1 Computer Architecture	
Course Code:21UCSSS1 (Compulsory)	**Credits : 2**

Objectives:

- To study basic computer organization.
- To understand the basic Arithmetic operations algorithms.
- To understand the memory organization.

Course Outcomes:

CO No.	Upon completion of this course, students will be able to	PSO addressed	CL
CO-1	discuss the organization of basic computer	1	Un
CO-2	explain various types of instructions.	1	Un
CO-3	explain general register organization and stack organization	1	Un
CO-4	explain algorithms for arithmetic operations of various integer number systems	1	Un
CO-5	explain algorithms for arithmetic operations of floating number systems	1,4	Un
CO-6	discuss memory hierarchy with different types of memories.	1,2	Un

SEMESTER- III	
Self Study 1	**Computer Architecture**
Course Code:21UCSSS1(Compulsory)	**Credits : 2**

Unit I:

Basic computer organization and design :

Instruction codes –computer registers –computer instructions –timing and control –instruction cycle-memory reference instructions

Unit II:

Central processing Unit:

General register organization –stack organization-instruction formats –addressing modes- data transfer and manipulation-program control-Reduced Instruction Set Computer.

Unit III:

Computer Arithmetic:

Addition and subtraction – multiplication algorithms-division algorithms

Unit IV:

Computer Arithmetic:

floating point arithmetic operations- Decimal Arithmetic unit- Decimal Arithmetic operations

Unit V:

Memory organization:

Memory hierarchy –main memory –auxiliary memory-associative memory – cache memory – virtual memory

Text Book :

1. M. Morris Mano .*Computer System Architecture.* New Delhi: Pearson Education. Third Edition 2017 .

Books for Reference:

1. P.V.S. Rao .*Computer system Architecture* .New Delhi: PHI Learnings.Second Printing. 2011
2. John P.Hayes .*Computer Organization and Architecture.* India: Tata McGraw Hill. Third Edition 2002
3. John D. Carpinelli*Computer Systems Organization & Architecture.* India: Tata McGraw Hill. First edition 2002.