PROGRAMMING LANGUAGE COURSES (PLC)

INTRODUCTION TO WEB PROGRAMMING

Course Code: PLC141/241 **Credits: 2:0:1**

Pre-requisites: Nil **Contact Hours: 28L+14P**

Course Coordinator: Ashwitha A

Course Content

Unit I

Introduction to Web Programming: Structural Elements, Title Element, meta Element, HTML Attributes, Body Element, Differences between Old HTML and HTML5, HTML Coding Conventions. Comments, Block Elements, block quote Element, Whitespace Collapsing, pre Element, Phrasing Elements, Editing Elements, q and cite Elements, dfn, abbr, and time Elements br and wbr Element, sub, sup, s, mark, and small Elements, strong, em, b, u, and i Elements, Span Element.

- Pedagogy/Course delivery tools: Chalk and talk, Power Point Presentation, Demonstration of code
- Links: https://www.w3schools.com/html/html_attributes.asp

Unit II

HTML5: Hello HTML5, Loose Syntax Returns, XHTML5, HTML5: Embracing the Reality of Web Markup, Presentational Markup Removed and Redefined, HTML5 Document Structure Changes, Adding Semantics: Marking Text, Indicating Dates and Time, Inserting Figures, HTML5's Open Media Effort: Audio, Video, Client-Side Graphics with: Drawing and styling lines and shapes.

- Pedagogy/Course delivery tools: Chalk and talk, Power Point Presentation, Demonstration of code
- Links: https://www.w3schools.com/charsets/tryit.asp?deci=193&ent=Aacute

Unit III

Introduction to CSS: CSS Overview, CSS Rules, Example with Type Selectors and the Universal Selector, CSS Syntax and Style, Class Selectors, ID Selectors, span and div Elements, Cascading, style Attribute, style Container, External CSS Files, CSS Properties, Color Properties, RGB Values for Color, Opacity Values for Color, HSL and HSLA Values for Color, Font Properties, line-height Property, Text Properties, Border Properties, Element Box, padding Property, margin Property.

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation, Demonstration of code
- Links: https://www.w3schools.com/css/css_syntax.asp

Unit IV

Tables and CSS, Links and Images: Table Elements, formatting a Data Table: Borders, Alignment, and Padding, CSS Structural Pseudo Class Selectors, thead and tbody Elements, Cell Spanning, CSS display Property with Table Values, a Element, Different Types of href Values, Relative URLs, CSS for Links, Bitmap Image Formats: GIF, JPEG, PNG, img Element, Positioning Images, Shortcut Icon, iframe Element. Backgroun images, web fonts

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation, Demonstration of code
- Links: https://www.w3schools.com/css/css_table.asp, https://www.w3schools.com/css/css_link.asp

Unit V

Introduction to JavaScript: Functions, DOM, Forms, and Event Handlers History of JavaScript, Hello World Web Page, Buttons, Functions, Variables, Identifiers, Assignment Statements and Objects, Document Object Model, Form Attributes, Form Element, Controls, Text Control, E-mail adress generator web page, event-handler

- Pedagogy/Course delivery tools: Chalk and talk, Power Point Presentation, Demonstration of code
- Links: https://www.w3schools.com/js/js_functions.asp

Lab Component:

1. Create a web page using HTML to create your biodata that includes personal details (Name, date of birth, Address, contact number, email id), Qualification (10th and 12th marks/grades, with school/college information), List of achievements (Create a link to at least 1 achievement), insert your photo(image).
2. Develop web page for a typical wedding event using List tag. Apply HTML include
   a. Heading
   b. Image
   c. Paragraph
   d. Ordered list for Groceries, Vegetables of type numbers and alphabets with description
   e. Unordered list for Fruits, Stationery Items, Flowers of shape circle, square and diamond with description
3. Apply CSS border property to create the following table. Give proper caption for the table and do the following.

Firstname	Lastname	Age
Priya	Sharma	24
Arun	Singh	32
Sam	Watson	41

   a. Left Align the text, border spacing of 5px and cell padding of 15px
   b. Use row span and Column span
4. Create a web page to create a form using **CSS** that includes tables and user interface components such as text boxes, text areas, buttons, check boxes and

combo box. Create a feedback form to enable students to give their feedback regarding the teacher.

5. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country; its capital should be printed next in the list. Add **CSS** to customize the properties of the font of the capital (color, bold and font size).
6. Design a web page that contains your biodata and personal data. Apply **CSS** to include
   a. Button features
   b. Images
   c. Text
   d. Pagination
   e. At least 2 columns of detailed information
7. Write a **JavaScript** to design a simple calculator to perform the following operations: sum, product, difference and quotient
8. Write **JS** code to compute factorial of a given number and display the same in the alert box.
9. Design a web page to enter purchase details with respect to a grocery store. 1. Items purchased 2. Quantity 3. Item Code 4. Item Price
   On click of the submit button display the details in table format. Display the total price paid.
10. Design a web page to include text boxes for entering 2 numbers and buttons (factorial, prime, Fibonacci, Natural Numbers). Display alert box and change the background of the text box when the user focuses onto the text box. Also, Display the name of the button when the user moves over the buttons. When the button is clicked, perform the required computation and print the result in the web page. Create 3 programs for front end, styling and back end.

Text Books:

1. HTML & CSS: The Complete Reference Thomas A. Powell, Fifth Edition, Tata McGraw Hill (Unit 2)
2. WEB PROGRAMMING with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, First Edition (Unit 1, 3, 4, 5)

Reference Books:

1. Randy Connolly, Ricardo Hoar, “Fundamentals of Web Development”, 1st Edition, Pearson Education India. (ISBN:978-9332575271)
2. David Sawyer Mcfarland, “JavaScript & jQuery: The Missing Manual”, 1st Edition, O’Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014 (ISBN:978-9351108078)
3. Zak Ruvalcaba Anne Boehm, “Murach’s HTML5 and CSS3”, 3rd Edition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016. (ISBN:978-9352133246)

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Use different HTML tags for web page design. (PO-1, PO-3)
2. Develop web pages using HTML5 constructs (PO-1, PO-3)
3. Design web pages using basic features of CSS and HTML (PO-1, PO-2, PO-3)
4. Apply advanced features of CSS and HTML to create web pages. (PO-1, PO-2, PO-3)
5. Apply features of JavaScript to handle client side interactions. (PO-1, PO-2, PO-3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal test-I	30	CO1, CO2
Internal test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Lab Component Evaluation	20	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

INTRODUCTION TO PYTHON PROGRAMMING

Course Code: PLC142/242 **Credits: 2:0:1**

Pre-requisites: Nil **Contact Hours: 28L+14P**

Course Coordinators: Dr. J Geetha & Darshana A Naik

Course Content

Unit I

The way of the program: The Python programming language, what is a program? What is debugging? Syntax errors, Runtime errors, Semantic errors, Experimental debugging.

Variables, Expressions and Statements: Values and data types, Variables, Variable names and keywords, Conditional Statements, Evaluating expressions, Operators and operands, Type converter functions, Order of operations, Operations on strings, Input, Composition, The modulus operator.

Iteration: Assignment, Updating variables, the for loop, the while statement, The Collatz 3n + 1 sequence, Nested Loops for Nested Data.

Functions: Functions with arguments and return values, Lambda Functions

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link: https://www.learnbyexample.org/python/
  https://www.learnpython.org/
  https://pythontutor.com/visualize.html#mode=edit

Unit II

Tuples: Tuples are used for grouping data, Tuple assignment, Tuples as return values, Composability of Data Structures.

Lists: List values, Accessing elements, List length, List membership, List operations, List slices, Lists are mutable, List deletion, Objects and references, Aliasing, Cloning lists, Lists and for loops, List parameters, List methods, Pure functions and modifiers, Functions that produce lists, Strings and lists, list and range, Nested lists, Matrices.

Dictionaries: Dictionary operations, dictionary methods, aliasing and copying.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link: https://www.learnbyexample.org/python/
  https://www.learnpython.org/
  https://pythontutor.com/visualize.html#mode=edit

Unit III

Modules: Random numbers, the time module, the math module, creating your own modules, Namespaces, Scope and lookup rules, Attributes and the dot Operator.

Files: About files, writing our first file, Reading a file line-at-a-time, turning a file into a list of lines, Reading the whole file at once, working with binary files, Directories, fetching something from the web. Algorithms: Linear search, Binary search, merging two sorted lists.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link: https://www.learnbyexample.org/python/
  https://www.learnpython.org/
  https://pythontutor.com/visualize.html#mode=edit

Unit IV

Object oriented programming: Classes and Objects — The Basics, Attributes, Adding methods to our class, Instances as arguments and parameters, Converting an instance to a string, Instances as return values, Objects are mutable, Sameness, Copying.
Inheritance: Polymorphism, Generalization, Pure functions, Operator Overloading.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link: https://www.learnbyexample.org/python/
  https://www.learnpython.org/
  https://pythontutor.com/visualize.html#mode=edit

Unit V

Exceptions: Catching exceptions, Raising our own exceptions, the finally clause of the try statement.
Strings: Working with strings as single things, Working with the parts of a string, Length, Traversal and the for loop, Slices, String comparison, Strings are immutable, The in and not in operators, A find function, Looping and counting, Optional parameters, The built-in find method, The split method, Cleaning up your strings, The string format method.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link: https://www.learnbyexample.org/python/
  https://www.learnpython.org/
  https://pythontutor.com/visualize.html#mode=edit

Lab Component:

SL. No.	QUESTIONS
1	a) Write a python program to read 2 numbers from the keyboard and perform the basic arithmetic operations based on the choice. (1-Add, 2-Subtract, 3-Multiply, 4-Divide) b) Write a python program to find the factorial of number using while loop. c) Write a python program to add 10 numbers by inputting each from the keyboard using for loop.
2	a) Write a python function linearSearch() to read an array and search for the key element. Display the appropriate messages. Use the recursive function. b) Write a python program to define a function max_of_three() that takes three numbers as arguments and returns the largest of them using default arguments. c) Write a python program to define a function generate_n_chars() that takes an integer n and a character c and returns a string, n characters long. For example, generate_n_chars(5,”x”) should return the string “xxxxx“ using keyword only parameters.

3	a) Write a python program to implement a stack and queue using lists b) Write a python program to create a list of tuples having first element as the strings and the second element as the length of the string. Output the list of tuples sorted based on the length of the string. c) Write a python program to create a list and perform the following operations • Inserting an element • Removing an element • Appending an element • Displaying the length of the list • Popping an element • Clearing the list
4	a) Write a tiny Python program numDict.py that makes a dictionary whose keys are the words 'one', 'two', 'three', and 'four', and whose corresponding values are the numerical equivalents, 1, 2, 3, and 4 (ints, not strings). b) Write a Python program to store PROFILE_DATA(user_id, name, DOB, qualification, work_experience) in a dictionary and pretty print the dictionary contents. (import pprint)
5	a) Write a Python program to demonstrate built-in modules (Random,Time, Math, etc,) b) Create a user defined module using python to execute the following a) area of circle b) area of triangle c) area of rectangle.
6	a) Write a python program to create a text file and ask the user to enter 5-6 lines of text. Display the longest and the shortest word from the file. Display the length of these words. b) Develop a python program to sort the contents of a text file and write the sorted contents into a separate text file. [Hint: Use string methods strip(), len(), list methods sort(), append(), and file methods open(), readlines(), and write()].
7	a) Develop a python program that uses class Student which prompts the user to enter marks in three subjects and calculates total marks, percentage and displays the score card details. [Hint: Use list to store the marks in three subjects and total marks. Use __init__() method to initialize name, USN and the lists to store marks and total, Use getMarks() method to read marks into the list, and display() method to display the score card details.] b) Write a python program for the following: c) Create a class called time. Its three members all type int should be called hours, minutes and seconds. Write a python program that prompts the user to enter a time values separately. The Program should then store the time in the object and finally printout the total no of seconds represented by this value. Use appropriate member functions.
8	a) Write a python program to create a class called Mylist that shadows a python list: it should overload + operator to append the data to the list. Also provide constructor for your class that takes an existing list. b) Write a python program to implement the following using Inheritance Employee Clerk Software Engineer Team Leader
9	a) Write a python program to Build a Number guessing game. When user enters not an integer raise the Exception and print total number of Guesses. b) Write a python function named DivExp which takes TWO parameters a, b and returns a value c (c=a/b). Write suitable assertion for a>0 in function DivExp and raise an exception for when b=0. Develop a suitable program which reads two values from the console and calls a function DivExp.

10	a) Write a python program to implement the following using strings
	The third person singular verb form in English is distinguished by the suffix -s, which is added to the stem of the infinitive form: run -> runs. A simple set of rules can be given as follows:
	• If the verb ends in y, remove it and add ies
	• If the verb ends in o, ch, s, sh, x or z, add es
	• By default just add s
	• Test your function with words like try, brush, run and fix. Tip: Check out the string method endswith().
	b) Write a python function partition() that splits a list of soccer players into two groups. More precisely, it takes a list of first names (strings) as input and prints the names of those soccer players whose first name starts with a letter between and including A and M.
	>>> partition([‘Eleanor’, ‘Evelyn’, ‘Sammy’, ‘Owen’, ‘Gavin’])
	Eleanor
	Evelyn
	Gavin
	>>> partition([‘Xena’, ‘Sammy’, ‘Owen’])

Suggested Learning Resources

Text Books:

1. Downey, A., Elkner, J., & Meyers, C. (2012). How to think like a computer scientist: learning with python. Green Tea Press, Wellesley, Massachusetts.

Reference Books:

1. Al Sweigart, “Automate the Boring Stuff with Python”, 1stEdition, No Starch Press, 2015. (Available under CC-BY-NC-SA license at https://automatetheboringstuff.com/)
   (Chapters 1 to 18, except 12) for lambda functions use this link: https://www.learnbyexample.org/python-lambda-function/
2. Allen B. Downey, “Think Python: How to Think Like a Computer Scientist”, 2nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license at http://greenteapress.com/thinkpython2/thinkpython2.pdf
   (Chapters 13, 15, 16, 17, 18) (Download pdf/html files from the above link)

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Demonstrate proficiency in handling loops and creation of functions
2. Identify the methods to create and manipulate lists, tuples and dictionaries.
3. Develop programs using modules and files.
4. Interpret the concepts of Object-Oriented Programming as used in Python.
5. Demonstrate the use of built-in functions for string processing and exception handling.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal test-I	30	CO1, CO2
Internal test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Lab Component Evaluation	20	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

BASICS OF JAVA PROGRAMMING

Course Code: PLC143/243 **Credits: 2:0:1**

Pre-requisites: Nil **Contact Hours: 28L+14P**

Course Coordinator: Dr. Yogish H K

Course Content

Unit I

An Overview of Java: Introduction to Object-Oriented Programming, Simple Java Programs, identifiers, literals, Data Types, Variables, and Arrays: Java Is a Strongly Typed Language, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, Variables, Arrays, and Strings.

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation
- Links: https://pythontutor.com/visualize.html#mode=edit

Unit II

Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ?: Operator, Operator Precedence, Control Statements: Java's Selection Statements - if, if-else, nested if else, if else ladder, Iteration Statements – while, do-while, for, for-each, nested loops, Jump Statements – break, continue, return.

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation
- Links: https://pythontutor.com/visualize.html#mode=edit

Unit III

Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize() Method, A Closer Look at Methods and Classes: Overloading Methods, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Introducing Access Control, Understanding static, Introducing final

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation
- Links: https://pythontutor.com/visualize.html#mode=edit

Unit IV

Inheritance: Inheritance, using super, creating a Multilevel Hierarchy, When Constructors Are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance.

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation
- Links: https://pythontutor.com/visualize.html#mode=edit

Unit V

Packages and Interfaces: Packages, Access Protection, Importing Packages, Interfaces, 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.

- Pedagogy/Course delivery tools: Chalk and talk, PowerPoint Presentation
- Links: https://pythontutor.com/visualize.html#mode=edit

Lab Component:

1. Java Program to demonstrate arithmetic operators, relational operators and bitwise operators.
2. Java Program to find the largest and smallest of three numbers using a ternary operator.
3. Write a Java program to find Fibonacci series, using while and for loop.
4. Write a Java program to calculate a Factorial of a number
5. Write a java program to check if a given number is palindrome.
6. ATM program Java- representing ATM transactions such as withdraw the money, deposit the money, check the balance, and exit using switch statement.
7. Java Program to print the largest element in an array.
8. Java Program to Add Two Matrices
9. Write a Java program to implement a linear Search Algorithm.
10. Write a JAVA program to add two complex numbers using Class.
11. Create a JAVA class called Student with the following details as variables within it. USN, NAME, BRANCH, PHONE, PERCENTAGE, Write a JAVA program to create n Student objects and print the USN, Name, Branch, Phone, and percentage of these objects with suitable headings.
12. Write a JAVA program demonstrating Method overloading and Constructor overloading.
13. Find the area of the rectangle using constructors and method area().
14. Write a Java program to demonstrate the finalize() method that helps in garbage collection.
15. Design a super class called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a JAVA program to read and display at least 3 staff objects of all three categories.
16. Java Program to demonstrate the uses of super.
17. Describe the abstract class called shape, which has three subclasses say triangle, rectangle and circle. Define one method area() in abstract class and override this area() in these three subclasses to calculate areas of triangle, rectangle and circle.
18. Write a JAVA program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero. Also demonstrate working of ArrayIndexOutOfBoundException.
19. Write a JAVA program demonstrating finally block for handling exceptions.
20. Java program to implement multiple inheritance.

Text Books:

1. Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007

Reference Books:

1. Bruce Eckel, Thinking in Java, 4th Edition, Tata McGraw Hill

Course Outcomes (COs):

At the end of the course, students will be able to:

1. Understand Object Oriented Programming Concepts (PO-1)
2. Apply Java programming constructs to solve the given problems (PO-1, PO-2, PO-3)
3. Apply the concept of Classes and Objects to solve the given problems. (PO-1, PO-2, PO-3)
4. Use the concepts of polymorphism and inheritance to solve the given problems. (PO-1, PO-2, PO-3)
5. Design the solutions for the given real world problems using concepts of packages, interfaces and Exception Handling. (PO-1, PO-2, PO-3)

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal test-I	30	CO1, CO4
Internal test-II	30	CO2, CO3, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Lab Component Evaluation	20	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

INTRODUCTION TO C++ PROGRAMMING

Course Code: PLC144/244 **Credits: 2:0:1**

Pre-requisites: Nil **Contact Hours: 28+14**

Course Coordinator: Ashwitha A

Course Content

Unit I

Introduction to Object Oriented Programming: Computer programming background, C++ overview. First C++ Program, Basic C++ syntax, Object Oriented Programming: What is an object, Classes, methods and messages, abstraction and encapsulation, inheritance, abstract classes, polymorphism.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link: Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
  Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

Unit II

Tokens, Expressions and Control Structures, Array in C++: Tokens, Keywords, Identifiers and constants, Operators in C++, Scope resolution operator Expressions and their types, Special assignment expressions, Decision making statements: if, if-else, switch, Loops: while loop, do-while loop, for loop, Array: Introduction, initializing single dimension array, Linear search operation on array elements.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Link : Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
  Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

Unit III

Functions In C++: Function prototyping, Call by Value, Call by reference, Return by reference Inline functions, Default arguments, Function overloading.

Inheritance & Polymorphism: Derived class Constructors, Destructors, Types of Inheritance Defining Derived classes, Single Inheritance, Multiple, Hierarchical Inheritance, Hybrid Inheritance.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Links: Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
  Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

Unit IV

I/O Streams: C++ Class Hierarchy, File Stream, Text File handling, Binary File handling during file operations.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Links: Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
  Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

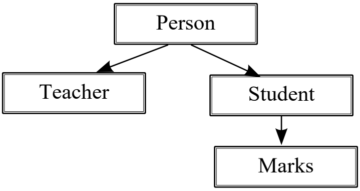
Unit V

Exception Handling: Introduction to Exception, Benefits of Exception handling, Try and catch block Throw statement, Pre-defined exceptions in C++.

- Pedagogy/Course delivery tools: Chalk and talk, Power point presentation, Videos
- Links: Basics of C++ - https://www.youtube.com/watch?v=BClS40yzssA
  Functions of C++ - https://www.youtube.com/watch?v=p8ehAjZWjPw

Lab Component:

SL. No.	QUESTIONS
1	a) Write a C++ program to find the area and circumference of a circle b) Write a C++ program to find the simple interest c) Write a C++ program to find the area of a triangle given its sides d) Write a C++ program to get the name, age and salary of a person and display the same.
2	a) Write a C++ program to find the factorial of a number b) Write a C++ program to find whether the entered number is palindrome or not. c) Write a C++ program to find the sum of all the natural numbers from 1 to n. d) Write a C++ program to find sum of all the elements, maximum and minimum element in an array
3	a) Write a C++ program to search an element in an array using linear search b) Write a C++ program to find whether an entered number is prime or not using a function (with value, with return type) c) Write a C++ program to swap 2 values by writing a function that uses call by reference technique.
4	a) Write a C++ program to overload function for computing the area triangle, circle and square b) Write a C++ program to overload a function to add two numbers of different data types (int, float, double)
5	a) Write a C++ program to perform square of a number using inline function b) Write a C++ program to create a class called bank_acct with following data member (cust_name, cust_accno, balance) and member functions (read_details, deposit, withdraw, display balance). Read and display details using array of objects and implement deposit and withdraw using inline.
6	a) Write and execute a C++ Program to display names, roll no's, and grades of 3 students who have appeared in the examination. Create a class with data members as Name, Roll no and Marks for 3 subjects. Write a method to calculate the grade of a student. b) Create a C++ class that includes constructors to do the following. • Create an uninitialized string. • Initialize an object with a string constant at the time of creation. • Create an object and initialize with another object. Also write a function to concatenate two strings.

7	a) Write a C++ program to implement the following inheritance.  • Assume suitable data members and member functions for all the classes. • Display the number of publications for a teacher and percentage marks for a student.
8	Write a C++ program to demonstrate multilevel inheritance for the following: Suppose we have three classes Vehicle, FourWheeler, and Car. The class Vehicle is the base class, the class FourWheeler is derived from it and the class Car is derived from the class FourWheeler. Class Vehicle has a method 'vehicle' that prints 'I am a vehicle', class FourWheeler has a method 'fourWheeler' that prints 'I have four wheels', and class Car has a method 'car' that prints 'I am a car'. So, as this is a multi-level inheritance; we can have access to all the other classes methods from the object of the class Car. We invoke all the methods from a Car object and print the corresponding outputs of the methods. So, if we invoke the methods in this order, car(), fourWheeler(), and vehicle(), then the output will be I am a car I have four wheels I am a vehicle
9	a) Write a C++ program to create a text file, check file created or not, if created it will write some text into the file and then read the text from the file. b) Write a C++ program to read the contents from a text file, count and display the number of alphabets present in it.
10	a) Write a program that creates a Calculator class. The class contains two variables of integer type. Design a constructor that accepts two values as parameter and set those values. • Design four methods named Add (), Subtract (), multiply (), Division ( ) for performing addition, subtraction, multiplication and division of two numbers. • For addition and subtraction, two numbers should be positive. If any negative number is entered then throw an exception in respective methods. So design an exception handler (ArithmeticException) in Add () and Subtract () methods respectively to check whether any number is negative or not. • For division and multiplication two numbers should not be zero. If zero is entered for any number then throw an exception in respective methods. So design an exception handler (ArithmeticException) in multiply () and Division () methods respectively to check whether any number is zero or not.

Text Books:

1. Bhushan Trivedi, "Programming with ANSI C++", Oxford Press, Second Edition, 2012.
2. Balagurusamy E, Object Oriented Programming with C++, Tata McGraw Hill Education Pvt.Ltd, Fourth Edition 2010.

Course Outcomes (COs):

At the end of the course the student will be able to:

1. Explain the characteristics of Object oriented programming approach.
2. Develop programs based on decision making statements and arrays.
3. Achieve code reusability and extensibility by means of Inheritance and Polymorphism.
4. Demonstrate C++ functions to perform operations on a file.
5. Illustrate the use of Exception handling feature in C++ for handling errors at runtime.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes attained**
Internal test-I	30	CO1, CO2
Internal test-II	30	CO3, CO4, CO5
Average of the two internal tests shall be taken for 30 marks.		
Other Components		
Lab Component Evaluation	20	CO1, CO2, CO3, CO4, CO5
Semester-End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5