Suggested Teaching Guidelines for

Programming Fundamentals for Design and VerificationPG

DVLSI March-2024

Duration: 30 class room hours and 40 Hrs Lab

Objective: Introduce C and C++ Programming.

Prerequisites: Awareness of Programming Concepts.

Evaluation method: CCEE Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam – 20% weightage

List of Books / Other training material

Text Book:

No specific courseware for module, faculty may share some course materials.

Reference book:

1. Thinking in C++ by Bruce Eckel
2. Fundamentals of Data Structures in C++ by Horowitz, Sahani & Mehata
3. The C programming Language by Kernighan and Retchie
4. Advanced Unix Programming by Rochkind
5. The C++ Programming Language, Bjarne Stroustrup;
6. Data Structures, Algorithms and Applications in C++ by Sartaj Sahni
7. Object-oriented Analysis And Design Using Umlan Introduction To Unified Process And Design Patterns 1st Edition by Mahesh P. Matha / PHI
8. C++ Primer Plus by Stephen Prata / Pearson
9. Fundamentals of Data Structures in C++ by Horowitz, Sahani & Mehata / Orient Longman

Session 1:

- Introduction to C, History, Standards
- Overview of C Basics: Variables, Data Types, Constants, Qualifiers, Operators, Control Structures etc.
- Pointers:
  Concept of pointers
  Pointer arithmetic
  Chain of pointers
  Pointer to const, const pointers
  Void pointer, NULL pointer

Assignment:

1. If a five-digit number is input through the keyboard, write a program to calculate the sum of its digits.

2. A cashier has currency notes of denominations 10, 50 and 100. If the amount to be withdrawn is input through the keyboard in hundreds, find the total number of currency notes of each denomination the cashier will have to give to the withdrawer.
3. If the three sides of a triangle are entered through the keyboard, write a program to check whether the triangle is isosceles, equilateral, scalene or right angled triangle.
4. Write a program to find the greatest of the three numbers entered through the keyboard using conditional operators.

Session 2:

- Arrays : 1D, 2D arrays, Pointers & Arrays

Assignment:

1. Write a recursive function to obtain the running sum of first 25 natural numbers.
2. Write a program to find GCD of numbers.
3. Write a program for number format conversions (Decimal, binary and octal)
4. Write a program to pick up the largest number from any 5 row by 5 column matrix.
5. Write a program that interchanges the odd and even components of an array.

Session 3:

- Functions
    - Overview of functions
    - Scope & Lifetime of variables
    - Recursion
    - Function pointers
    - External linkage
- C Preprocessor

Assignment:

1. A 5-digit positive integer is entered through the keyboard, write a function to calculate sum of digits of the 5-digit number:
   (1) Without using recursion
   (2) Using recursion
2. A positive integer is entered through the keyboard, write a program to obtain the prime factors of the number. Modify the function suitably to obtain the prime factors recursively.
3. Given three variables **x**, **y**, **z** write a function to circularly shift their values to right. In other words if x = 5, y = 8, z = 10 after circular shift y = 5, z = 8, x =10 after circular shift y = 5, z = 8 and x = 10. Call the function with variables **a**, **b**, **c** to circularly shift values.
4. Write a program to test whether a character entered is a small case letter or not.

Session 4:

- Strings
- Structures & unions
- Dynamic Memory management

Assignment:

1. Write a program that extracts part of the given string from the specified position. For example, if the sting is "Working with strings is fun", then if from position 4, 4 characters are to be extracted then the program should return string as "king". Moreover, if the position from where the string is to be extracted is given and the number of characters to be extracted is 0 then the program should extract entire string from the specified position.

2. Write a program that replaces two or more consecutive blanks in a string by a single blank. For example, if the input is
   Sample: “Grim return to the planet of apes!!”
   Expected Output: “Grim return to the planet of apes!!” (There should be only one space between two words.)
3. Write a program to create an employee structure with the members as employee ID, employee name, salary, year of joining. Accept the data for a certain number of employees and find their total, average, minimum and maximum salary. Also find the employees with maximum, minimum years of service.
4. Write a program that will read a line and delete from it all occurrences of the word ‘the’.

Session 5:

- Introduction to C++
- New features in C++
- Reference variables, new & delete operators
- C++ functions (Overloading, Default arguments, Inline functions, extern "C" block)

Session 6:

- Variables & Functions
- Public, Private access specifiers
- Constructors & destructors
- Copy constructor
- namespace

Assignment – Lab:

- Write a Student class and use it in your program. Store the data of ten students and display the sorted data according to their roll numbers, date of births, and total marks.
- Print the average of three numbers entered by the user by creating a class named 'Average' having a function to calculate and print the average without creating any object of the Average class.
- C++ program to Display Student Details using constructor and destructor.
- C++ program to Display Student Details using constructor and destructor.

Session 7:

- this keyword
- Static members
- Friend functions & classes
- Const functions, mutable variables
- Explicit constructors

Session 8:

- overloading unary, binary operators
- overloading as friend functions
- overloading <<, >>
- Type Conversions

Assignment – Lab:

- Write Date and Time classes that allows you to add, subtract, read and print simple dates in dd/mm/yyyy and time in hh:mm:ss forms. Use function overloading in your program.
- Assignments to overload =, ==, +, ++, --, <<, >> and [ ] operators.
- Write a C++ program to Compare Two Strings using overloading.

- Write a C++ Program to check Palindrome using function overloading.

Session 9:

- Inheritance
- Types of inheritance
- Overriding
- Protected members
- Constructors in inheritance

Assignment – Lab:

- Design a hierarchy of computer printers. Use multiple inheritance in your hierarchy. Also use friend functions and classes in your program.

Session 10:

- Virtual functions
- Virtual destructors
- Pure Virtual functions
- Virtual Inheritance

Assignment – Lab:

- Using virtual and pure virtual functions implement hierarchy of computer printers.

Session 11:

- Class Template
- Function Template

Assignment – Lab:

- Write a program using class template.

Session 12 & 13:

- Definition of file
- File handling in C++
- Doing read, write operation in files

Assignment – Lab:

- Assignments on files doing different operations.

Session 14 & 15:

- STL
- RTTI
- Advanced Typecasting

Assignment – Lab:

- Find the number of students who are passes or failed using MAP.
- Find the prime numbers from 2 to n using sieves algorithm, use SET
- Implement the Run time polymorphism using RTTI