

Course Title:	Introduction to C++ Programming		
Course Code:	<b>22PLC152/22PLC252</b>	CIE Marks	50
Course Type (Theory/Practical /Integrated )	Integrated	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P: S)	2:0:2	Exam Hours	03
Total Hours of Pedagogy	40 hours	Credits	03
<b>Course objectives</b> <ul style="list-style-type: none"> <li>● Understanding about object oriented programming and Gain knowledge about the capability to store information together in an object.</li> <li>● Understand the capability of a class to rely upon another class and functions.</li> <li>● Understand about constructors which are special type of functions.</li> <li>● Create and process data in files using file I/O functions</li> <li>● Use the generic programming features of C++ including Exception handling</li> </ul>			
<b>Teaching-Learning Process</b> These are sample Strategies, which teacher can use to accelerate the attainment of the various course outcomes and make Teaching -Learning more effective <ol style="list-style-type: none"> <li>1. Chalk and talk</li> <li>2. Onine demonstration</li> <li>3. Hands on problem solving</li> </ol>			
<b>Module-1 (8 hours)</b>			
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.			
<b>Textbook 1: Chapter 1(1.1 to 1.8)</b>			
<b>Module-2 (8 hours)</b>			
Functions in C++: Tokens – Keywords – Identifiers and constants – Operators in C++ – Scope resolution operator – Expressions and their types – Special assignment expressions – Function prototyping – Call by reference – Return by reference – Inline functions -Default arguments – Function overloading.			
<b>Textbook 2: Chapter 3(3.2,3.3,3.4,3.13,3.14,3.19, 3.20) , chapter 4(4.3,4.4,4.5,4.6,4.7,4.9)</b>			
<b>Module-3 (8 hours)</b>			
Inheritance & Polymorphism: Derived class Constructors, destructors-Types of Inheritance-Defining Derived classes, Single Inheritance, Multiple, Hierarchical Inheritance, Hybrid Inheritance.			
<b>Textbook 2: Chapter 6 (6.2,6.11) chapter 8 (8.1 to,8.8)</b>			
<b>Module-4 (8 hours)</b>			
I/O Streams: C++ Class Hierarchy- File Stream-Text File Handling- Binary File Handling during file operations.			
<b>Textbook 1: Chapter 12(12.5) , Chapter 13 (13.6,13.7)</b>			

**Module-5 (8 hours)**

Exception Handling: Introduction to Exception - Benefits of Exception handling- Try and catch blockThrow statement- Pre-defined exceptions in C++

**Textbook 2: Chapter 13 (13.2 to13.6)**

**Course outcome (Course Skill Set)**

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

CO1	Able to understand and design the solution to a problem using object-oriented programming concepts.
CO2	Able to reuse the code with extensible Class types, User-defined operators and function Overloading.
CO3	Achieve code reusability and extensibility by means of Inheritance and Polymorphism
CO4	Implement the features of C++ including templates, exceptions and file handling for providing programmed solutions to complex problems.

**Programming Assignments:**

1. Write a C++ program to sort the elements in ascending and descending order.
2. Write a C++ program to find the sum of all the natural numbers from 1 to n.
3. Write a C++ program to swap 2 values by writing a function that uses call by reference technique.
4. Write a C++ program to demonstrate function overloading for the following prototypes.

*add(int a, int b)*

*add(double a, double b)*

5. Create a class named Shape with a function that prints "This is a shape". Create another class named Polygon inheriting the Shape class with the same function that prints "Polygon is a shape". Create two other classes named Rectangle and Triangle having the same function which prints "Rectangle is a polygon" and "Triangle is a polygon" respectively. Again, make another class named Square having the same function which prints "Square is a rectangle".Now, try calling the function by the object of each of these classes.

6. 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

Write a C++ program to demonstrate multilevel inheritance using this.

7. 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.

8. Write a C++ program to write and read time in/from binary file using fstream

9. Write a function which throws a division by zero exception and catch it in catch block. Write a C++ program to demonstrate usage of try, catch and throw to handle exception.

10. Write a C++ program function which handles array of bounds exception using C++.

### **Assessment Details (both CIE and SEE)**

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50). The minimum passing mark for the SEE is 35% of the maximum marks (18 marks out of 50). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/ course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination(SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

#### **Continuous Internal Evaluation(CIE):**

##### **Two Unit Tests each of 20 Marks (duration 01 hour)**

- First test after the completion of 30-40 % of the syllabus
- Second test after completion of 80-90% of the syllabus

One Improvement test before the closing of the academic term may be conducted if necessary. However best two tests out of three shall be taken into consideration.

##### **Two assignments each of 10 Marks**

The teacher has to plan the assignments and get them completed by the students well before the closing of the term so that marks entry in the examination portal shall be done in time. Formative (Successive) Assessments include Assignments/Quizzes/Seminars/ Course projects/Field surveys/ Case studies/ Hands-on practice (experiments)/Group Discussions/ others. . The Teachers shall choose the types of assignments depending on the requirement of the course and plan to attain the Cos and POs. (to have a less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course). CIE methods /test question paper is designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

**The sum of two tests, two assignments, will be out of 60 marks and will be scaled down to 30 marks**

##### **CIE for the practical component of the Integrated Course**

- On completion of every experiment/program in the laboratory, the students shall be evaluated and marks shall be awarded on the same day. The **15 marks** are for conducting the experiment and preparation of the laboratory record, the other **05 marks shall be for the test** conducted at the end of the semester.
- The CIE marks awarded in the case of the Practical component shall be based on the continuous evaluation of the laboratory report. Each experiment report can be evaluated for 10 marks. Marks of all experiments' write-ups are added and **scaled down to 15 marks**.
- The laboratory test (**duration 02/03 hours**) at the end of the 14<sup>th</sup> /15<sup>th</sup> week of the semester /after

completion of all the experiments (whichever is early) shall be conducted for 50 marks and **scaled down to 05 marks.**

Scaled-down marks of write-up evaluations and tests added will be CIE marks for the laboratory component of IPCC for **20 marks.**

#### **Semester End Examination(SEE):**

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

- The question paper shall be set for 100 marks. The medium of the question paper shall be English/Kannada). The duration of SEE is 03 hours.

**The question paper will have 10 questions. Two questions per module. Each question is set for 20 marks. The students have to answer 5 full questions, selecting one full question from each module. The student has to answer for 100 marks and marks scored out of 100 shall be proportionally reduced to 30 marks. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), should have a mix of topics under that module.**

**Lab SEE will be conducted based on the Lab assignments with both internal and external examiners as per prevailing practice. The exam will be conducted for 50 marks and minimum passing is 20 marks. The marks obtained will be proportionally reduced to 20 marks (max) and will be summed with theory SEE to get the total SEE marks.**

**Passing in the subject: The student will pass the subject only if he obtained minimum passing marks both in theory SEE and Lab SEE. If a student fails in either theory/lab he has to clear the corresponding component only. Grading will be assigned by combining the performance in Lab and theory.**

#### **Suggested Learning Resources:**

**Books (Title of the Book/Name of the author/Name of the publisher/Edition and Year)**

Textbooks

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.

#### **Web links and Video Lectures (e-Resources):**

Weblinks and Video Lectures (e-Resources):

1. Basics of C++ - <https://www.youtube.com/watch?v=BCIS40yzsA>
2. Functions of C++ - <https://www.youtube.com/watch?v=p8ehAjZWjPw>

Tutorial Link:

1. [https://www.w3schools.com/cpp/cpp\\_intro.asp](https://www.w3schools.com/cpp/cpp_intro.asp)
2. <https://www.edx.org/course/introduction-to-c-3>

#### **Activity Based Learning (Suggested Activities in Class)/ Practical Based learning**

- Assign small tasks to Develop and demonstrate using C++

**COs and POs Mapping (Individual teacher has to fill up)**

COs	POs						
	1	2	3	4	5	6	7
C01							
C02							
C03							
C04							

**Level 3- Highly Mapped, Level 2-Moderately Mapped, Level 1-Low Mapped, Level 0- Not Mapped**