



**Meru University of Science and Technology**  
**School of Computing and Informatics**  
**Department of Computer Science**

---

**Course Title:** OBJECT ORIENTED PROGRAMMING I

**Course code:** CIT 3153

**Year of study:** ONE **Semester:** TWO

**Pre-requisite** Fundamentals of Computer Programming

**Course Purpose**

The purpose of this course is to teach programming concepts using the C++ programming language which prepares a learner for advanced programming courses.

**Expected Learning Outcomes**

At the end of the course, the learner should be able to:

- Describe the object-oriented programming approach using C++ programming language
- Apply the concept of encapsulation, inheritance and polymorphism when solving an Object oriented problem using C++ programming language
- Apply techniques of structured (functional) decomposition to decompose problem and a program solution into smaller pieces
- Design and implement code that includes the reuse of both existing code and calling functions in the C++ libraries
- Apply fundamental syntax rules for declaring and using constructors and destructors in a program
- Develop C++ programs by writing small programs from specifications given in class

## Course Content

Week	Main Topic	Sub-topic
1	<b>Student semester registration</b>	
2	<b>Introduction to Object Oriented Programming (OOP)</b>	<ul style="list-style-type: none"> <li>• What is OOP?</li> <li>• How is Object Oriented Programming (OOP) different from Procedural Oriented Programming (POP)?</li> <li>• What are the features of OOP?</li> <li>• What are the advantages and disadvantages of using OOP approach?</li> <li>• What is the structure/format of a C++ program?</li> </ul>
3	<b>Variables, Constants, Data types and structure of C++ program</b>	<ul style="list-style-type: none"> <li>• What is a variable?</li> <li>• What rules govern variable naming?</li> <li>• What are the different types of variables?</li> <li>• What is a constant?</li> <li>• How are constants defined in C++?</li> <li>• What is a data type?</li> <li>• How are data types defined in C++?</li> <li>• What are the different types of data types?</li> </ul>
4	<b>Control Structures</b>	<ul style="list-style-type: none"> <li>• How are selection control structures implemented in C++?</li> <li>• How are iteration control structures implemented in C++?</li> <li>• How are jump control structures implemented in C++?</li> </ul>
5	<b>Arrays and Functions</b>	<ul style="list-style-type: none"> <li>• How are arrays implemented in C++?</li> <li>• How are functions used in C++?</li> <li>• How is function <b>overloading</b> implemented in C++?</li> </ul>
6	<b>Classes and Objects</b>	<ul style="list-style-type: none"> <li>• What is a class?</li> <li>• How is a class defined in C++?</li> <li>• How is an object declared in C++?</li> <li>• What is the structure of a C++ program using classes? (OOP)</li> </ul>
7	<b>CAT</b>	
8	<b>Principle of <b>Encapsulation</b> in C++</b>	<ul style="list-style-type: none"> <li>• What is encapsulation?</li> <li>• How is encapsulation implemented in C++?</li> </ul>

9	<b>Principle of Inheritance in C++</b>	<ul style="list-style-type: none"> <li>• What is inheritance?</li> <li>• How are the different types of inheritance implemented in C++?</li> <li>• How is function overriding implemented in C++?</li> </ul>
10	<b>Principle of Polymorphism in C++</b>	<ul style="list-style-type: none"> <li>• What is polymorphism?</li> <li>• How is polymorphism implemented in C++?</li> </ul>
11	<b>Constructors and Destructors in C++</b>	<ul style="list-style-type: none"> <li>• What are constructors and destructors?</li> <li>• What are the features of constructors and destructors?</li> <li>• What are some of the restrictions that apply to constructors and destructors?</li> <li>• How is a constructor implemented in a C++ program?</li> <li>• How is a destructor implemented in a C++ program?</li> </ul>
12	<b>Virtual Functions</b>	<ul style="list-style-type: none"> <li>• What is a virtual functions?</li> <li>• How are virtual functions implemented in C++?</li> </ul>
13	<b>Files in C++</b>	<ul style="list-style-type: none"> <li>• How is the concept of files implemented in C++?</li> </ul>
14 & 15	<b>Examination</b>	

#### Mode of Delivery

- Lectures
- Class exercises
- Class group discussion
- Practical lab sessions

#### Course Assessment

Type	Weighting (%)
Examination	70
Continuous Assessment	30
Total	100

#### Core Reading Material

1. Object Oriented Programming with C++ by M.P. Bhawe, S.A. Patekar
2. Object oriented programming with C++ by Sourav Sahay
3. Object Oriented Programming Using C++, Sanjeev Sofat, Cyber Tech. Publication, New Delhi

#### Recommended Reference Material

1. Object Oriented Programming in C++ by E. Balaguruswamy, TMH Publishing Co. Ltd., New Delhi

Name of Lecturer **Mr. Anthony Irungu** ..... Signature ..... Date .....