

**Course:**

**OOP**

**Object Oriented**  
**Programming**

**Course Objectives and Contents**

# Lectutre Items

- Overall Aims Of Course:
- Intended Learning Outcomes
- OOP Course Contents
- Student Assessment Methods

# Object Oriented Programming

## Course Objectives (CS2103)

- Overall Aims Of Course:

The course aims to increase the student ability to deal with , understand and operate modern programming systems , It also aims to improve Programming skills, basically using object programming approach , and The file I/O operations are also achieved .

# Intended Learning Outcomes Of Course (ILOs) :

## **a) Knowledge and Understanding:**

- a1 - understand structure , objects and classes in OOP.
- a2 - understand the operations of actual system.
- a3- understand the concepts of encapsulation , inheritance and polymorphism , overloading functions, overriding, abstract class, pointers in OOP programming
- a4- understand the operations of file I/O in programming .

# Course (ILOs) – Cont.

## b) Intellectual Skills:

- **b1- differentiate between structure and Class , Classes and Objects.**
- **b2 -apply the concept of Object Oriented Programming in his ( her ) computer programming operations .**
- **b3- analyze a practical problem and solve it with computer programming methods.**
- **b4- deal with actual computer programs and discuss and enhance its operations.**
- **b5 -design of modern programming system using OOP approach**

## Course (ILOs) – Cont.

### c) Professional and Practical Skills:

- c1- use the actual existing programming module to design or modify current systems
- c2- use of encapsulation, inheritance , and polymorphism, , overloading functions, overriding, abstract class in computer programming systems
- c3- develop actual problem solving systems using OOP.
- c4- modify existing system for getting best and reliable operations from it.

# OOP Course Contents

1. General revision about C++ Programming
2. Structured Programming
3. Basic definition about OOP
4. Structure
5. Classes and Objects
6. Objects as arguments
7. Arrays and Strings
8. Inheritance
- 9. Mid- Term Exam**
10. Functions overloading
11. overriding
12. Polymorphism
13. Abstract class
14. encapsulation
15. Streams and files
16. Disk file I/O with streams
17. File Pointers
- 18. Final- Term Exam**

# Assessment Methods

| Method  | Weight | Week            |
|---|--------|-----------------|
| <u>Term assignments</u><br>- Quizzes<br>- Reports | 10     | During the Term |
| Mid- Term Exam                                    | 10     | 8               |
| Practical Examination                             | 10     | 15              |
| Final-Term exam                                   | 70     | 16              |
| Total   | 100%   |                 |