

Department of Data Science (PU CIT)
University of the Punjab
Object Oriented Programming – Spring 2025
(BSDS F24 M&A Sections)

Course code	CC-211
Credit hours	3
Prerequisite	CC-112 Programming Fundamentals (and Enthusiasm, Consistency and Honesty too)
Follow up	CC-213 Data Structures
Course Instructor	Muhammad Idrees Email: idrees@pucit.edu.pk, sir.idrees@gmail.com Office hours: Friday (10:00am to 12:30pm) Any time (working day) with permission through email

Course Objectives

- To equip the learner with the philosophy and necessary skills to formulate solutions of real world problems using object-oriented paradigm.
- Justify the philosophy of object-oriented design and the concepts of encapsulation, abstraction, inheritance, and polymorphism.
- Strong concepts of object manipulation and dynamic memory allocation within classes

Textbook

- John R. Hubbard, *Schaum's Programming with C++*, 2nd Edition, McGRAW HILL.
- Tony Gaddis, *Starting out with C++: from control structures through objects*, 7th Ed., Addison-Wesley.

Reference Books/Websites

- D.S. Malik, *C++ Programming, From Problem Analysis to Program, Design*, 5th Ed.
- <http://www.learncpp.com/>.
- <https://www.youtube.com/playlist?list=PLAE85DE8440AA6B83>.

Grading Instruments (Sessional decomposition may vary at end of course)

5 to 8 marks for	Quizzes, planned/announced or sudden
10 to 15 marks for	Programming assignments
2 to 3 marks for	Written assignments
Up-to 5 marks for	Term project / technical report
35 marks for	Midterm exam/pre-mid lecture notes
40 marks for	Final exam/post-mid lecture notes

Passing Criteria

- As per college rules, minimum requirement to pass this course is to get overall 50% marks.

Tentative Course Outline and Lecture Plan

Topics	No. of Lectures
Revision of Structured Programming, Pitfalls of Structured Programming	2
Revision of Arrays and File handling in C/C++	4
Introduction/Motivation to Object-Oriented Programming: Terminology; Real world examples	1
From Struct to Class, Access Specifiers: private and public only; Member functions (methods); Data Abstraction through Encapsulation; Setter/Mutator and Getter/Accessor methods; How the function knows which object invoked him; Two Special Method: Constructor & Destructor; Constructor with No argument (Default Constructor); Constructor with Default arguments; Constructor with one argument (Type-cast Constructor) Calling sequence of Constructor & Destructor for multiple objects; Destructors	3
<p>Pointer as data member: Role of destructors; Pointer/Reference to objects; Passing objects to functions by reference; Pointer this.</p> <p>Passing objects to functions by value; Default Member-wise copy (Assignment & Initialization); Problem of Member-wise copy & its solution: Copy Constructor; Calling sequence of Constructor & Destructor: when objects are passed by value/reference</p>	2
<p>Object as data member (Composition), Aggregation; Cascading calls with and without this pointer.</p> <p>Nameless objects; Array of objects; Constant data members; Static functions; Static data members; constant and static objects; Calling sequence of Constructor & Destructor for constant and static objects</p> <p>Friend functions (efficient but shake the concept of encapsulation); Declaring a global function as friend of a class; Declaring member function of a class as friend of another class; Friend class (an easy but more un-secure way);</p>	2
<p>Operator Overloading: Binary Operator receiving Instance of class as 1st operand; Overloading Unary Operator: as member, as non-member; Unusual Operators: ++, --, [] (set & get, both versions), type-cast, Parentheses;</p> <p>Operator NOT receiving Instance of class as 1st operand Overloading Binary Operators (Stream Insertion/Extraction) for I/O stream; Other unusual operators: comma, arrow, new, delete, delete[] etc.</p>	2
Missed/Misc topic; Review of the course for Mid Term Exam	

MIDTERM EXAM	
<p>is-A Relationship: Public Inheritance; Protected data member; protected, private inheritance; Multilevel Inheritance: Direct and Indirect Base Class; Calling of Constructor and Destructor for Derived Class Objects; Explicit call to the constructor of Base class from Derived class; Review of Simple Inheritance, Multilevel Inheritance, Multiple Inheritance</p> <p>A review of OO relationships terminologies: Aggregation, composition, generalization (is-A), knows-a.</p>	3
<p>Polymorphism: Overriding base-class members in derived class; Virtual functions and Dynamic binding; Pure virtual functions and abstract class; Defining a pure virtual destructor</p> <p>Concept of v-Pointer and v-Table; Detail discussion on v-table structure, virtual constructor, object cloning</p> <p>Some tricky things in pointer type casting and applying them on different topics studied so far.</p>	3
Multiple inheritance: Scope problem and Diamond problem; Virtual inheritance	1
CLI; GUI; Other OOP languages; Streams; Database; Graphics; Data structures	4
<p>Function Template: Overloading of Function Template; Specialized of Template Function</p> <p>Class Template: Specialized method of Template Class; Complete Specialized Template Class</p>	2
Standard template library (STL)	2
Missed/Misc. topics; Revision	1
FINAL EXAM	

Important Notes

- Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties.
- You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible.
- The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.
- Introductory knowledge of programming is assumed for this course. All code written in quizzes, assignments, homework's, and exams must be in C++. Code must be intelligently documented (commented). Undocumented code may not be given any credit.

- You **may** use CodeBlocks or Microsoft Visual Studio.NET or eclipse IDE with g++ compiler for writing your code OR choose one of your own choice.
- There is no makeup for missed sessional grading instruments like quizzes, assignments, and home works.