



Department of Computer Science

CS-1004 – Object Oriented Programming Spring 2022

Instructor Name: Mr. Farooq Ahmed **TA Name:**
Email address: farooq.ahmad@nu.edu.pk **Email address:** N/A
Office Location:
Office Hours: Friday 9:00 – 11:00 AM

Course Information

Program: BS (CS) **Credit Hours:** 3+1 **Type:** Core **Class Venue:**
Pre-requisites: Programming Fundamentals (CS-1002)
Class Meeting Time: Section D: Mon/Wed 11:30 – 12 :50 pm
Section C: Mon/Wed 01:00 – 02 :30 pm

Course Description/Objectives/Goals:

The core objectives of this course are to introduce,

- Object oriented programming with data abstraction and encapsulation.
- The classes, objects and relationship among different objects and classes in C++
- Generic programming using templates, and template specializations.

Course Learning Outcomes (CLOs):

At the end of the course students will be able to:	Domain	BT* Level
Understand principles of object oriented program	C	2
Identify the objects & their relationships to build object oriented solution	C	3
Model a solution for a given problem using object oriented principles	C	3
Implement and examine an object oriented solution	C	4

* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain

Course Textbooks:

1. C++ Programming: Program Design Including Data Structures, by D. S. Malik (8th Edition)
2. C++: How to Program? by Deitel & Deitel (9th Edition)
3. Problem Solving with C++, by Walter Savitch

(Tentative) Grading Criteria:

Assignments + Project	(25 %)
Quizzes	(5 %)
Midterms	(30 %)
Final Exam	(40 %)

Course Policies:

- Quizzes may be announced or surprise. No Late Submissions and Makeup Quizzes.
- **Plagiarism** in any work (Labs, Quiz, Assignment, Midterms, and Final Exam) from any source, (Internet or a Student) will result in **F** grade or deduction of **absolute marks**.
- **80%** attendance is required for appearing in the Final exams.
- Absolute Grading Scheme will be used.

Course Contents Weekly and Lecture-wise Breakdown

Week	Topic	Lecture-1	Lecture-2
1	Pointers	Pointers Introduction, Pointer variables and Initialization, “Address of” operator, Dereferencing operator. Pointer operations (Relational, Arithmetic)	Use of Constant with Pointers. Difference between a Pointer and a Reference. Passing pointers to functions by value and by reference.
2		Dynamic memory allocation using pointers and accessing dynamic memory. Dynamic variables. new and delete operators.	Dynamic 1- dimensional arrays, Create, Delete, Grow and Shrink. Example of programs using 1D dynamic allocation: e.g., mathematical sets union and intersection.
3		Memory Leak and Dangling Pointers, Dynamic 1- dimensional char arrays for strings, string operations like search, concatenation etc.	Pointers Indirection. Dynamic 2D, allocation, matrices, strings etc.
4	Object-oriented basics	Structured Programming vs Object-oriented Programming, Principles of modularization, abstraction and encapsulation.	Objects vs Class, state vs behavior, access specifiers (Public, Private) , Member functions (accessors, utilities, mutators etc)
5		Constructors (default, overloaded), Function overloading.	Dynamic memory allocation and Object assignment, Parameter passing, Shallow vs Deep copy,
6	Object-oriented basics & Operator overloading	Mid Term 1	
7		Cascaded function calls, static members, inline functions and other miscellaneous issues	Unary operators, Binary operators using member functions
8		Binary operators using non-member functions, concept of friendship,	Unary operators, Pre and post increment, subscript operator.
9	Object and Class relationships	Part-whole relationships, Association/Aggregation/Composition Implementation issues (constructor call sequence, initializer list, etc)	Multiplicity, Memory Management Bi-directional relationships, Forward-class declarations issues
10		Inheritance basics, Type of Inheritance, public, protected, private.	Function Overriding and sub-typing details Polymorphism introduction
11		Static vs dynamic binding details, virtual tables and virtual pointers, Polymorphism vs down casting, run-time type identification, dynamic cast	Pure-virtual functions, Abstract classes, Interfaces (optional)
12		Multiple Inheritance and Diamond Problem	Mid Term 2
13	Generic Programming & Exception Handling.	Template functions	Template classes
14		Template Specializations, Generic Programming using Polymorphism	Exception Handling.
15	Advance Topics (Optional)	Introduction to STL, Iterators and Collections	Object Serialization