



National Computing Education Accreditation Council NCEAC



NCEAC.DOC.001-E

COURSE DESCRIPTION

NAME OF INSTITUTION	Lahore Garrison University
PROGRAM (S) TO BE EVALUATED	Computer Science , Fall 2022

Course Description : This course is designed to provide the students familiar with programming language C++. The course will cover Basics Functions, structures, classes, inheritance, polymorphism, file handling.

Course Code	CSC321								
Course Title	Object Oriented Programming								
Credit Hours	3+1								
Prerequisites by Course(s) and Topics	Programming Fundamentals								
Assessment Instruments with Weights (homework, quizzes, midterms, final, programming assignments, lab work, etc.)	SESSIONAL (Quizzes, Assignments, Presentations) =25 % Midterm Exam =25 % Final Exam = 50%								
Course Coordinator	Ayesha Nasir								
URL (if any)									
Current Catalog Description	This course is designed to provide the students familiar with programming language C++. The course will cover Basics Functions, structures, classes, inheritance, polymorphism, file handling.								
Textbook (or Laboratory Manual for Laboratory Courses)	Object Oriented Programming in C++ by Robert Lafore, 4th Edition , C++ Programming from Problem Analysis to Program Design by D.S Malik, 5th Edition								
Reference Material	Starting Out with C++ from Control Structures to Objects, 9th Edition, Tony Gaddis , C++ How to Program, 10th Edition, Deitel & Deitel.								
Course Goals	In this course the student will gain a broad understanding of modern computer programming. The student will acquire introductory skills in problem analysis, solution design, and program construction . Through practical programming activities, the student will gain an appreciation of the nature and history of computer programming. The topics includes Functions, structures, classes, operating overloading and overriding, inheritance, polymorphism, file handling, templates								
Course Learning Outcomes (CLOs):	<table><tr><td>At the end of the course the students will be able to:</td><td>Domain</td><td>BT Level*</td></tr><tr><td>* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain</td><td></td><td></td></tr></table>			At the end of the course the students will be able to:	Domain	BT Level*	* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain		
At the end of the course the students will be able to:	Domain	BT Level*							
* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain									

Topics Covered in the Course, with Number of Lectures on Each Topic (assume 15-week instruction and one-hour lectures)	Week	Lecture	Topics Covered
	Week 1	1	C++ Basics Review (I/O, Data Types, Control Structures, Loops, Arrays, Functions)
		2	Functions in C++ (Introduction, advantages, Uses of functions, Classification of Functions)
	Week 2	3	User defined and Built in Functions
		4	User defined functions
	Week 3	5	Structures , Structures Vs Arrays
		6	Nested Structures Introduction to OOP, Advantages of OOP and its History,
	Week 4	7	Structures with functions
		8	File Handling (Introduction, advantages, disadvantages, file handling classes)
	Week 5	9	Const vs non constant functions, static data members and functions
		10	Introduction to Fundamentals of OOP (objects, classes, inheritance, polymorphism, abstraction, encapsulation)
	Week 6	11	Classes and Objects
		12	Class member functions and data members, Access specifiers
	Week 7	13	Access Specifiers , classes vs structures
		14	Classes with public, private and protected Access Specifiers
	Week 8	1 hours	Mid Term
	Week 9	15	Constructors (Introduction, purpose, difference between constructor and functions)
		16	Types of constructor (default and parameterized)
	Week 10	17	Types of constructor (copy constructor)
		18	Destructor
	Week 11	19	Friend functions
		20	Inheritance (Introduction, purpose and types)
	Week 12	21	Inheritance (Single level, multiple inheritance syntax and implementation and difference)
		22	Inheritance (Multilevel level, hierarchical inheritance syntax and implementation and difference)
	Week 13	23	Inheritance (hybrid inheritance syntax and implementation)
		24	Function overloading (Simple)
	Week 14	25	Polymorphism (Introduction, purpose, categories)
		26	Polymorphism (Compile time polymorphism, function overloading)
	Week 15	27	Polymorphism (Compile time polymorphism, operator overloading)
		28	Polymorphism (Run Time polymorphism, operator overriding)
	Week 16	29	Polymorphism (Run Time polymorphism, operator overriding)
		30	Polymorphism (Virtual Function)
	Week 17	2 hours	Final Term
Laboratory Projects/Experiments Done in the Course			
Programming Assignments Done in the Course			

Instructor Name Ayesha Nasir

Instructor Signature

Date