

Course Title:	Programming Workshop Python I
Course Code:	CSE132
Program:	Master Degree In Computer Engineering
Department:	Computer Engineering
Course coordinator:	Dr. Amina GHARSALLAH
Institution:	Private Higher School of Engineers of Gafsa (ESIP)

A. Course identification

1. Credit hours: 1.5(0-0-1.5)
2. Course type
a. College Department Others
b. Fundamental Transversal Optional
3. Level/year at which this course is offered: 1.1/3
4. Pre-requisites for this course (if any): Basic programming (variables, loops, conditions and
functions), Basic OOP

1. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Self- study	Total workload
1	Traditional classroom			
2	Blended	22.5		
3	E-learning		23	45.5
4	Distance learning			
5	Other ()			

2. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	Lecture Design of the second s	-
2	Laboratory/Studio	22.5
3	Tutorial	-
4	Others (specify)	
	Total	22.5



B. Course Objectives and Learning Outcomes

Course Description

This course offers a hands-on introduction to C++ programming, focusing on writing, testing, and debugging code. Students will learn variables, control structures, functions, and basic data structures like arrays and pointers.

The workshop helps students apply algorithmic concepts from CSE131 (Algorithm and Data Structures) through practical coding exercises. Topics include loops, recursion, file handling, object-oriented programming, and algorithm efficiency.

By the end of the course, students will be able to write C++ programs, use fundamental data structures, and apply algorithms to solve real-world problems.

Course Main Objectives

This course aims to:

- \checkmark Teach the basics of C++, including variables, data types, and control structures.
- ✓ Build problem-solving skills using loops, functions, and recursion.
- ✓ Introduce data structures, such as arrays and pointers, for managing data.
- ✓ Apply sorting and searching techniques to make programs more efficient.
- ✓ Improve coding practices, including debugging and file handling.

1. Course Learning Outcomes

CLOs		Aligned PLOs
1	Knowledge and understanding	
1.1	Understand C++ programming basics, including variables, data types, and control structures.	
1.2	Know the syntactic, semantic aspect and the concepts necessary for C++ (identifiers, declaration, operators, instructions, modules, structures, etc.).	PLOK.1
1.3	Learn the concept of object-oriented programming in C++ and work with arrays, strings, pointers, and STL containers.	
2	Skills	
	Work effectively in teams, communicate programming concepts, and present coding solutions.	PLOS.2
2.3	Develop problem-solving skills using loops, functions, and recursion in C++.	
	Implement and manage data structures such as	PLO.S5
	as arrays and pointers.	
2.4	Apply sorting and searching techniques to improve program efficiency.	

C. Course Content

No	List of Topics	Contact Hours
1	Introduction to C++: Variables, Data Types, Operators	4.5



No	List of Topics	Contact Hours
2	Control Structures: Conditional Statements & Loops	4.5
3	Functions & Recursion	4.5
4	Arrays, Pointers & Searching Algorithms	4.5
5	File Handling & Algorithm Optimization	4.5
Total		22.5

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding		
1.1	 ✓ Understand C++ programming basics, including variables, data types, and control structures. ✓ Know the syntactic, semantic aspect and the concepts necessary for C++ (identifiers, declaration, operators, instructions, modules, structures, etc.). ✓ Learn the concept of object-oriented programming in C++ and work with arrays, strings, pointers, and STL containers. 	- Lecturing - Practical work	- Assignments - Quizzes - Exams
2.0	Skills		
2.1	✓ Work effectively in teams, communicate programming concepts, and present coding solutions.	 Practical work projects 	- Assignments, - Quizzes - Exams
	✓ Develop problem-solving skills using loops, functions, and recursion in C++.	- Class discussions	- Report
2.2 Eco	 ✓ Implement and manage data structures such as arrays and pointers. ✓ Apply sorting and searching techniques to improve program efficiency. 	- Assignments - projects	- Quizzes - Exams

2. Assessment Tasks for Students Ce Galsa

	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Practical Work (written or oral)	Weekly	%
2	Quizzes, Homework assignments	Random	%
3	First mid Term	8	%
4	Final Exam	16	100%



E. Student Academic Counselling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice:

1- Office hours

2- Blackboard interface

F. Learning Resources and Facilities

1. Learning Resources

Required Textbooks	 Bjarne Stroustrup, <i>The C++ Programming Language</i>, 4th Edition, Addison-Wesley, 2013. Stephen Prata, <i>C++ Primer Plus</i>, 6th Edition, Pearson, 2011.
Essential References Materials	PC Data projector
Electronic Materials	 Cplusplus.com – www.cplusplus.com (Comprehensive C++ documentation and tutorials) GeeksforGeeks – www.geeksforgeeks.org (Concept explanations, code examples, and problem-solving exercises)
Other Learning Materials	NA
2 Facilities Required	

2. Facilities Required

Item	Resources
	Classroom board
Accommodation	Computer lab with the necessary software
	Internet access
Technology Resources	Data projector

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of teaching and assessment.	Students, Faculty, Program Leaders, Peer Reviewer	Direct/Indirect
Extent of achievement of course learning outcomes.	Faculty, Program Leaders, Peer Reviewer	Direct, Indirect
Quality of Learning resources	Faculty, Program Leaders, Peer Reviewer	Direct, Indirect
Teaching and learning quality and effectiveness.	Students, Faculty Program Leaders, Peer Reviewer	Direct, Indirect

H. Specification Approval Data

Council / Committee	Computer Engineering Council
Date	11/09/2023