Course Specification

— (Bachelor)

Course Title: Computer Programming-1	
Course Code : 501220-3	
Program: Bachelor of Computer Science	
Department: Department of Computer Science	
College: College of Computers and Information Technology	
Institution: TU	
Version: 1	
Last Revision Date: 01-02-2024	

Table of Contents

A. General information about the course:

1. Course Identification

1. Credit hours: (3)

2. Course type

A.	☐ University	☒ College	☐ Department	☐ Track	☐ Others
B.	☒ Required			☐ Elective	

3. Level/year at which this course is offered: (3rd Level/2nd Year)

4. Course general Description:

Introduce the fundamental concepts of programming and problem-solving techniques. Topics include data types, control structures, arrays and the mechanics of running, testing, and debugging.

5. Pre-requirements for this course (if any):

None

6. Pre-requirements for this course (if any):

None

7. Course Main Objective(s):

Students at the end of this course are able :
• Develop problem solving and algorithm development skills
• Develop understanding of fundamental concepts in computer programming
• Develop skills to write programs using control structures and loops

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	3	60%
2	E-learning	-	-
3	Hybrid ● Traditional classroom ● E-learning	-	-
4	Distance learning	-	-
5	Other (Lab)	2	40%

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	**Lectures**	45
2.	**Laboratory/Studio**	30
3.	**Field**	-
4.	**Tutorial**	-
5.	**Others (specify)**	-
Total		75

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	**Knowledge and understanding**			
1.1	Understand the basic terminology used in computer programming	**K1**	Lectures Labs Project	**Direct Assessment Tool** Quizzes / Homework/Project/ Exams **Indirect Assessment Tool** Course Exit Survey
2.0	**Skills**			
2.1	Write, compile and debug programs written in C++ language	**S1**	Lectures Labs	**Direct Assessment Tool** Quizzes /Homework/ Exams **Indirect Assessment Tool** Course Exit Survey
2.2	Use different data types in a computer program	**S1**	Lectures Labs	**Direct Assessment Tool** Quizzes /Homework/ Exams **Indirect Assessment Tool** Course Exit Survey

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
2.3	Design programs involving decision structures and loops	**S2**	Lectures Labs	**Direct Assessment Tool** Quizzes /Homework/ Exams **Indirect Assessment Tool** Course Exit Survey
3.0	**Values, autonomy, and responsibility**			

C. Course Content

No	List of Topics	Contact Hours
1.	Overview of Computers and Programming Languages	5
2.	Introduce basic components of a C++ program, including special symbols, and identifiers	5
3.	Explore simple data types	5
4.	Introduce arithmetic operators and examine how a program evaluates arithmetic expressions	10
5.	Introduce input and output statements	10
6.	Write complete C++ programs that use the introduced topics	10
7.	Control Structures 1 : Decisions	10
8.	Control Structures 2 : Loops	10
9.	Arrays and Strings	10
Total		**75**

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	HomeWorks/StudentParticipation-Attendance	Every Week	10%
2.	Quizzes	Week 4 & 12	10%
3.	Final Labs Exam	Week 7	10%
4.	Mid-Term	Week 15	20%
5.	Final Examination	Week 16	50%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	C++ Programming: From Problem Analysis to Program, D S Malik, CEGAGE Learning, 2011
Supportive References	C++ how to Program, Harvey M. Deitel and Paul J. Deitel, Prentice Hall, 2008
Electronic Materials	
Other Learning Materials	

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	• Classroom with 20-30 chairs • Lab with 15 PCs and required software tools installed (Dev c++)
Technology equipment (projector, smart board, software)	• Video projector / data show • White board
Other equipment (depending on the nature of the specialty)	

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	● Students ● Faculty members ● Coordinator ● Council ● Curriculum Committees	● Course exit survey ● Feedback from Faculty members ● Feedback from Course Coordinator ● Feedback from council ● Feedback from Curriculum Committees
Effectiveness of Students assessment	● Students ● Faculty members ● Coordinator ● Council ● Curriculum Committees	● Course exit survey ● Feedback from Faculty members ● Feedback from Course Coordinator ● Feedback from council ● Feedback from Curriculum Committees
Quality of learning resources	● Students ● Faculty members ● Coordinator	● Course exit survey ● Feedback from Faculty members

Assessment Areas/Issues	Assessor	Assessment Methods
	• Council • Curriculum Committees	• Feedback from Course Coordinator • Feedback from council • Feedback from Curriculum Committees
The extent to which CLOs have been achieved	• Students • Faculty members • Coordinator • Council • Curriculum Committees	• Course exit survey • Feedback from Faculty members • Feedback from Course Coordinator • Feedback from council • Feedback from Curriculum Committees
Quality of learning resources	• Students • Faculty members • Coordinator • Council • Curriculum Committees	• Course exit survey • Feedback from Faculty members • Feedback from Course Coordinator • Feedback from council • Feedback from Curriculum Committees

Assessors (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)

Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	CS council
REFERENCE NO.	Meeting #11
DATE	07/03/2024