COURSE CODE	COURSE TITLE	UNITS / TYPE
CCS0007	COMPUTER PROGRAMMING 2 (LEC)	2 UNITS LEC
PREREQUISITE	COMPUTER PROGRAMMING 1	
CO-REQUISITE/S	COMPUTER PROGRAMMING 2 (LAB)	
COURSE DESCRIPTION	This course is designed to expand the knowledge of learners by teaching some of the more advanced features of the C++ language. The study focuses in deeper understanding of the topics of this course that is the basic need of every programmer who wants to excel in C++ programming. The course includes many opportunities for hands-on experiences. This study covers modular programming with Functions, Arrays, C-String Handling and Manipulations, Pointers, Linked List, File Handling, Structures and Introduction to OOP using C++ programming. .	

INSTITUTION VISION STATEMENT

FEU Institute of Technology aims to be one of the top five technology educational institutions in the Philippines.

INSTITUTION MISSION STATEMENT

FEU Institute of Technology is dedicated to provide quality, relevant, innovative and industry-based education producing competent and principled professionals with greater sense of responsibility, social awareness and high competitiveness contributing significantly to the betterment of the society.

DEPARTMENT VISION STATEMENT

The Information Technology Department aims its program specializations to be a catalyst on the delivery of industry-based standards solutions and internationally recognized IT education.

DEPARTMENT MISSION STATEMENT

The Information Technology Department is committed to provide industry- based information technology solutions, international academic linkages, researches and IT certified professionals.

PROGRAM EDUCATIONAL OUTCOMES

The graduates of the Bachelor of Science in Information Technology program are:

1. engaged in further professional development and have interest in or aptitude for advanced studies or trainings in computing.
2. entrepreneurs or are employed in computing industries, organizing and managing team-based projects leading to successful and sustainable computing systems solutions.
3. responsible computing professionals actively participating in community groups that make a significant impact in addressing current and future societal challenges.

PROGRAM OUTCOMES

a. Apply knowledge of computing appropriate to the discipline.
b. Analyze a complex problem and identify and define the computing requirements appropriate to its solution.
c. Design, implement and evaluate computer-based systems or applications to meet desired needs and requirements.
d. Function effectively as a member or leader of a development team recognizing the different roles within a team to accomplish a common goal.
e. Understand professional, ethical, legal, security and social issues and responsibilities in the utilization of information technology
f. Communicate effectively with the computing community and with society at large about complex computing activities through logical writing, presentations, and clear
g. Analyze the local and global impact of computing information technology on individuals, organizations and society.
h. Recognize the need for and engage in planning self-learning and improving performance as a foundation for continuing professional development.
i. Apply knowledge through the use of current techniques and tools necessary for the IT profession

j. Ability to use and apply current technical concepts and practices in the core information technologies; human computer interaction, information management, programming, networking and web systems and technologies.
k. Identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems.
l. Integrate IT-based solutions into the user environment effectively.
m. Understand best practices and standards and their applications.
n. Assist in the creation of an effective IT project plan.
o. Ability to demonstrate understanding and proficiency of IT specialization

COURSE OUTCOMES AND RELATIONSHIP TO PROGRAM OUTCOMES															
	PROGRAM OUTCOMES														
COURSE LEARNING OUTCOMES (CLO)	**a**	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**	**n**	**o**
1. Design, implement, test and debug a program, based on a given specification, that uses (1) data structure arrays, strings, structures, linked list and files. (2) Conditional iterative, and recursive constructs, and (3) standard libraries in the assigned programming language		I													
2. Assess and recommend revisions to another programmer's code (1) regarding appropriateness of chosen data structure, (2) regarding appropriateness of chosen conditional and iterative constructs given a programming task, and (3) regarding thoroughness in applying procedural abstraction			E												
3. Adapt and apply appropriate techniques, resources, and modern computing tools to create computing applications									D						

WEEK	INTENDED LEARNING OUTCOME (ILO)	DETAILED COURSE CONTENT	TEACHING AND LEARNING ACTIVITY		ASSESSMENT TASK (AT)	CLO
			Mode of Delivery & Activities	**Hours Allotted**		
1	**COURSE ORIENTATION**					
	• Familiarize the student to the course outline • Conduct class regulation and policies	**Orientation of the Course** • Introduction of students and professor • Discussion of the Syllabus • Discussion of Student Outcomes and Program Educational Objectives • Course Requirements • Classroom Policies	*Synchronous Activity:* Class Discussion -Syllabus -Student Outcomes -Program Educational Objective - Course Requirements - Policies	2 hours		
			Asynchronous Activity: -Submission of Undertaking	40 minutes		
	MODULE 1. FUNCTIONS AND PARAMETERS					

1	• Identify User-Defined Functions • Identify Types of User-defined Functions in C++ • Create User defined functions in a C++ program • Understand Parameters and Arguments • Understand Function Overloading	**SUBTOPIC 1** **(User Defined Functions)** • Built-in Functions • Understanding User defined functions • Creating user defined functions • Void functions **SUBTOPIC 2** **(Function Parameters)** • Creating Function with Parameters • Return Statement • Default Arguments • Function Overloading	*Synchronous Activity:* Class Discussion *Asynchronous Activity:* View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	2 hours 40 minutes		**1**
2	**MODULE 2. Arrays**					
	• Create single dimensional arrays. • Understand how single array holds value • Create two dimensional arrays. • Understand how two dimensional array holds value	**SUBTOPIC 1** **(Single Dimensional Array)** • Creation of single dimension array • Passing one dimensional array to a function **SUBTOPIC 2** **(Two-Dimensional Array)** • Creation of two-dimension array • Passing two-dimensional array to a function	*Synchronous Activity:* Class Discussion Guided Formative *Asynchronous Activity:* View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	2 hours 40 minutes	Formative Assessment	**1**
3	**SUMMATIVE ASSESSMENT 1**					
4	**MODULE 3. Character and String Manipulation**					
	• Understand the CSTRING library and its functions • Create programs using CSTRING functions	**SUBTOPIC 1** **(CString Character Manipulation Functions)** • Create programs using CString functions • Create programs using cctype library and its functions	*Synchronous Activity:* Class Discussion	2 hours		**1**

	• Apply CSTRING function in solving problems related to string manipulation • Understand the STRING library and its functions • Create programs using STRING functions • Apply STRING function in solving problems related to string manipulation	**SUBTOPIC 2 (String Class Member)** • Creating Function with Parameters • Function Overloading	*Asynchronous Activity:* View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	40 minutes		
MODULE 4. Structures						
5	• Understand what is a • structure • Learn how to declare and • use structures in a program • Learn how to set structures as a function argument	**SUBTOPIC 1 (Structures)** • Declaring a structure • Accessing structure • members **SUBTOPIC 2 (Structures as an Argument)** • Setting structures as a function argument in a program	*Synchronous Activity:* Class Discussion Guided Formative *Asynchronous Activity:* View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	2 hours 40 minutes	Formative Assessment	**2**
6	**SUMMATIVE ASSESSMENT 2**					
7	**MODULE 5. Pointers**					
	• Learn about the pointers and its syntax definition. • Demonstrate how to declare a pointer just like declaring an ordinary variable of a certain type • Understand the difference between Dynamic Arrays and Pointers • Demonstrate how to declare and use dynamic arrays and pointers.	**SUBTOPIC 1 (Pointers)** • Declaration, assignment and dereferencing pointer • Pointers Arithmetic • Relationship between array and pointers **SUBTOPIC 2 (Dynamic Arrays)** • Defining Dynamic Array and Pointers • Create programs using dynamic arrays and pointers	*Synchronous Activity:* Class Discussion *Asynchronous Activity:* View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation)	2 hours 40 minutes		**2**

			View Guide			
8	**MODULE 6. Linked List**					
	• Understand what is a Linked List • Learn different operations of a Linked List • Create a basic Linked List program using C++	**SUBTOPIC 1** **(Linked List and its Operations)** • Linked List Structures • Traversing through • linked list • Dynamic array using • linked list **SUBTOPIC 2** **(Building a Linked List)** • Application of linked • list with structures • Singly linked list • Doubly linked list	*Synchronous Activity:* Class Discussion Guided Formative	2 hours	Formative Assessment	**2**
			Asynchronous Activity: View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	40 minutes		
9	**SUMMATIVE ASSESSMENT 3**					
10	**MODULE 7. File Handling**					
	• To understand and use different file handling functions. • To open, create and close a file using FILE Class libraries • To open, create and close a file using the fstream • library	**SUBTOPIC 1** **(The FILE Class)** • File class Library • Manipulating text file • File functions (create/append/read) • File access mode **SUBTOPIC 2** **(The fstream Library)** • FSTREAM Library • File functions • (create/append/read) • File access mode	*Synchronous Activity:* Class Discussion	2 hours		**3**
			Asynchronous Activity: View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	40 minutes		
	MODULE 8. Introduction to OOP					

11	• Learn the behavior and concepts of object and classes. • Learn how to create classes and instantiate objects • Understand Access modifiers • Understand member functions used in OOP • Understand what is Inheritance, • Understand the relationship between a base and a derived class	**SUBTOPIC 1** **(Basic OOP Concepts)** • Basic OOP terminologies. • Classes and member function. • Creating instances of an object from a class **SUBTOPIC 2** **(Member Functions and Inheritance)** • Accessor and Mutator • Destructor and Constructor • Inheritance • Base and derived class	*Synchronous Activity:* Class Discussion Guided Formative	2 hours	Formative Assessment	**3**
			Asynchronous Activity: View Video Courseware View Supplementary Materials Consultation with Faculty Members (on top of the assigned consultation) View Guide	40 minutes		
12	**SUMMATIVE ASSESSMENT 4**					
13	**FINAL EXAMINATION**					

CLO	SUMMATIVE ASSESSMENT
1	Summative Assessments: Module 1, 2, 3, Final Exam
2	Summative Assessments: Module 4 5, 6, Final Exam
3	Summative Assessments: Module 7, 8, Final Exam

GRADING SYSTEM

FINAL GRADE = 75% Weighted Average of Summative Assessments + 25% Final Exam

INSTRUCTIONAL MATERIALS

- Courseware Materials. This can be downloaded from the Learning Management System (LMS) via the URL https://onefeu.instructure.com
- Lecture Materials (PPT)/ Supplementary and Reinforcement Materials
- Video Materials
- Study Guide
- Project Specification Document

TEXTBOOK/REFERENCES/ONLINE REFERENCES

Textbooks:

- Malik, D. (2023). C++ Programming: From Problem Analysis to Program Design (10th Edition). Cengage Learning.
- Gregoire, M. (2021). Professional C++ (5th Edition). Wrox Press.
- McGrath, M. (2022). C++ Programming in Easy Steps (6th Edition). In Easy Steps Limited.
- Stroustrup, B. (2022). Programming: Principles and Practice Using C++ (2nd Edition). Addison-Wesley.
- Horton, I. (2020). Beginning C++ Game Programming (2nd Edition). Apress.
- Zak, D. (2021). An Introduction to Programming with C++ (9th Edition). Cengage Publishing.

Online Sources:

- https://www.w3schools.com/cpp/
- https://www.tutorialspoint.com/cplusplus/index.htm

- https://www.javatpoint.com/cpp-tutorial
- https://www.cprogramming.com/tutorial/c++-tutorial.html
- http://feu.instructure.com
- Courseware Materials available at http://feu.instructure.com
- FEUTECH e-book library collection at
- https://drive.google.com/drive/folders/15eccuUWwMrcnbHSNMvqn6HWMMyQB3d2?usp=sharing

Journals:
MIS Quarterly
Security Management
Tech & Learning
EBSCO
Journal of Educational Multimedia & Hypermedia

Prepared by:	**Approved by:**	**Version:**
Ronel F. Ramos Faculty-In-Charge	Dr. Roman M. De Angel Program Director, IT	Version: 6.0 Date Modified: August 2025
Evaluated by: Dr. Kirk Alvin S. Awat Coordinator, AGD		