



AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH (AIUB)

Faculty of Science and Technology
Department of Computer Science
Undergraduate Program

COURSE OUTLINE

I - Course Code and Title: CSC 1102 & 1103: Introduction to Programming

II - Credit: 3/1 credit hours (3 hours of Theory & 3 hours of Lab per week)

III - Nature: Major Course for BSc. in CSc

IV- Prerequisite: Nil

V - Course Description:

The goal of this course is to teach the key programming and problem-solving skills where the students will be able to design, implement, debug, and test structured and Object-Oriented Programs (OOP). This course teaches the fundamentals of Structured and OOP programming including – Data types, Operators & Expressions; Standard input/output (I/O); Conditional statements (if...else), Switch; Loops (do...while, while, for); Arrays (single, multidimensional); Pointers, Functions; Structure, Unions & Enumerations; Strings and The Object Oriented Programming (OOP) Principles such as Classes and Objects; Constructors; Operator Overloading; Inheritance and Polymorphism; Encapsulation, Abstract Class; Stream Input/Output, File Processing, Templates, Exception Handling, String Processing, Standard Template Library (STL) etc.

VI – Objectives:

At the end of the course, the students will be able to:

1. Understand the role of designing and the development of programming solutions to solve different computational problems.
2. Define aspects of Object-Oriented programming language.
3. Develop programs and demonstrate programming skills in Object-oriented Programming.
4. Make good use of compilers, debuggers, and related tools.
5. Develop the pre-requisite programming skills for Data Structure & Algorithm courses.
6. Apply gained knowledge and Skills to the IT industry.

VII – Topics to be covered

Time Frame		TOPICS	Specific Objective(s)	Suggested Activities	Teaching Strategy(s)
Week 1	L1	<ul style="list-style-type: none"> Mission & Vision of AIUB Rules and Regulations of the University, Department and the Course 	a. Mission & Vision of AIUB b. To understand how human being interacts with computer and machines c. How computer languages perform as the gateway to bridge up human and computer	Lecture, Lab: Exercise	Lecture notes, question-answer session.
	L2	<ul style="list-style-type: none"> Variables & Data types Operator Expression Comments 	a. Developing simple programs, compiling, and running b. Understand the concept of variables in programming. c. Explain different data types. d. Using different types of variables to store data e. Use of different types of operators and expressions.		
Week 2	L3	<ul style="list-style-type: none"> Advance uses of Operators (Conditional, Relational etc.) Type Casting 	a) Demonstrate the use of operators. b) Explain expressions in structural programming c) Illustrate type casting	Lecture, LAB: Exercise	Lecture notes, question-answer session.
	L4	<ul style="list-style-type: none"> Steps of Computation 	a) Apply knowledge to develop programs from Pseudo code		
Week 3	L5	<ul style="list-style-type: none"> If-else statement switch statement 	a) How to branch program control flow using conditional statements? b) To introduce the control structures of if-else and switch c) Using conditional operators through if, switch statement d) Introducing break statement to break a condition	Lecture, LAB: Exercise	Lecture notes, question-answer session.

	L6	<ul style="list-style-type: none">Loopswhile, do-while, for	a) How to perform iterative jobs using loops (while, do-while, for) b) How to repeat the execution of a block of code for a specific number of times c) Infinite loops d) How to convert a loop? (for example, writing a loop written in while to convert in for or do while)		
Week 4	L7	<ul style="list-style-type: none">ArrayArray Declaration and InitializationSimple Storing and Accessing Array Elements	a) How to declare, initialize, store array elements? b) Importance of array for storing and loop through similar type of data.	Lecture, Quiz – 1 (L1-L6), LAB Exercise	Lecture notes, question-answer session.
	L8	<ul style="list-style-type: none">Character Array, StringsString Handling Library Functions (self-study)	a) What are Strings? Relate Strings and Characters. b) Explain how Strings can be represented using array of characters.		
Week 5	L9	<ul style="list-style-type: none">PointersDynamic Memory Allocation	a) Introducing pointers to the students. b) How pointers point to a value of a variable? c) What are the advantages of using pointers? d) How to access pointers pointing to a variable and the address of a pointer itself?	Lecture, LAB: Exercise	Lecture notes, question-answer session.
	L10	<ul style="list-style-type: none">FunctionsFunction call	a) What is a function? b) How to declare, write and call a function?		
Week 6	L11	<ul style="list-style-type: none">Passing parameter/sReturning value	a) How to send arguments (actual parameters) and return values by calling a function?	Lecture, Quiz – 2 (L7-L10), LAB: Exercise	Lecture notes, question-answer session. Lab Mid Project Demonstration
	L12	Revision Class			
Midterm Week Week 7					

Week 8	<i>L13</i>	<ul style="list-style-type: none"> • Introduction to Object Oriented Programming 	a) Introducing Object Oriented Programming Principles to students b) Pillars of OOP development (Class, Objects, Inheritance, Encapsulation, Abstraction and Polymorphism)	Lecture, LAB: Exercise	Lecture, Student reporting and Board work
	<i>L14</i>	<ul style="list-style-type: none"> • Classes and Objects • Properties, Methods • Access Modifiers 	a) What are Classes and Objects in Object-Oriented programming? b) Real-life example of conceptualizing classes and objects c) Writing Classes with properties and functions d) Access modifiers (private, public, protected) and their purpose. e) Application of access modifiers for class, and class members (variables and function). f) Accessing class data member and member function from inside and outside of the class.		
Week 9	<i>L15</i>	<ul style="list-style-type: none"> • Constructor and Destructor • Static class member in C++ 	a) Constructor and Destructor: what and why? b) Default constructor, Parameterized constructor. c) Constructor and destructor calling order. d) Static class member e) Arguments /parameter for methods.	Lecture, LAB: Exercise	Lecture notes, question-answer session.
	<i>L16</i>	<ul style="list-style-type: none"> • Operator Overloading 	a) Passing arguments to a function (call-by-value, call-by-reference).		
Week 10	<i>L17</i>	<ul style="list-style-type: none"> • Inheritance 	a) Base class and derived class in relation to inheritance (real life example) b) What is inherited in inheritance /Access control in Inheritance (public, private and protected)	Lecture, Quiz – 3 (L 13-L16) , LAB: Exercise	Lecture notes, question-answer session.
	<i>L18</i>	<ul style="list-style-type: none"> • Inheritance 	a) Single, Multiple, and multi-level inheritance		

Week 11	L19	<ul style="list-style-type: none">Polymorphism	a) Run-time and Compile time polymorphism b) Virtual function c) Pure virtual function	Lecture, LAB: Exercise	Lecture notes, question-answer session.
	L20	<ul style="list-style-type: none">Overloading and OverridingEncapsulation and Abstraction	a) Data Abstraction and benefits of data abstraction, Interface in data abstraction.		
Week 12	L21	<ul style="list-style-type: none">Exception HandlingUser-defined Exceptions	a) What are Exceptions? How to handle Exceptions? b) Try and catch block in exception handling, throwing an exception, catching an exception, catching an exception, c) default and multiple catch blocks, nested try blocks d) How to utilize standard Exceptions? e) How to create user-defined exception handler?	Lecture, Quiz – 4 (L 17-L20), LAB: Exercise	Lecture notes, question-answer session.
	L22	<ul style="list-style-type: none">Templates	a) Function templates, class templates		
Week 13	L23	<ul style="list-style-type: none">Stream I/O (in depth)File Processing	a) Read from a file, writing to a file b) Standard library for reading and writing from/to a file, opening and closing a file.	Lecture, LAB: Exercise	Lecture notes, question-answer session. Lab Final Project Demonstration
	L24	Review for Final Exam			
Final Exam Week					
Week 14					
Lab Exam					
Week 15 (Exact Date will be Announced Later)					

VIII- Course Policies:

1. Student Attendance

All students are expected to attend all scheduled classes, and to read all assigned chapters / materials before coming to class. Attendance will be taken at the beginning of each class period.

Students with less than **80%** attendance will not receive any marks for attendance.

2. Class Participation & Peer Evaluation

You Students are expected to participate actively in the class. Your contribution towards your team will be counted too. Performance will be recorded during the theory lectures and labs. You must carry the text book with you during the class and lab. If you do not bring the book, then 0 will be given in the performance.

3. Quiz & Exam

For both terms, there will be 2 quizzes, best one quiz will be counted for grading. There will be Lab exam in addition to Lab Performances based on Lab Task / Assignment given by the course teacher. The details will be provided during Labs. During the quiz and exam, if you practice any unfair means, I will give you UW for the rest of the semester.

No Make-up Quiz will be taken!!!

4. Poor grades in Midterm Exam:

Students with “F” or “T” in the midterm, must drop (as per policy) the course right after the mid-term exam (within a week). If you wish to continue with “F” at your own risk and if you receive “F” in the final, you won’t be allowed to drop the course. You must drop the theory and Lab both.

There is no provision for any bonus marking. If such requests are made, penalty will be applied.

IX– Evaluation (Theory & Lab)

Mid	Theory	(3 credit)	Contribution
	Attendance:	10%	40 %
	Performance:	10%	
	Quiz:	30%	
	Mid-term Exam:	50%	
Final	Attendance:	10%	60%
	Performance:	10%	
	Quiz:	30%	
	Final Exam:	50%	
	Total:		100%

Mid	Lab	(1 credit)	40 %
	Attendance:	10%	
	Lab Performance:	5 x 10% = 50%	
	Project (Part-1):	40% (Deadline: 6th week)	
Final	Attendance:	10%	60%
	Lab Performance:	5 x 6% = 30%	
	Project (Part-2 in OOP)	40% (Deadline: 13th week)	
	Lab Exam	20%	
	Total:		100%

X – Textbook/ References

1. Course Manual for Introduction to Programming