Programming 1	Code & No:	CS 110
	Credits:	4 (3,2,0)
	Pre-requisite:	PCOM 113
	Co-requisite:	None
	Level:	3

Course Description:

This course introduces the students to the fundamentals of logic formulation together with their implementation in the C++ programming language. It introduces students to structured, top-down programming design and implementation. This course should serve as a foundation for students in the Computer Science and information technology program.

Course Aims:

1. Construct error-free C++ programs.
2. Divide a problem into its logical components.
3. Understand the basic structured programming concepts.
4. Design and code small to medium sized problems from the start using C/C++ constructs, such as input/output statements, if-then-else statements, while and for loops, functions
5. Employ some of the available data structures in C++ such as built-in data types, arrays and pointers to solve programming problems.
6. Use the required developments tools to write, compile, trace and debug C++ programs.

Student Outcomes (SOs):

☒(a) An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline

☒(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

☒(c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

☐(d) An ability to function effectively on teams to accomplish a common goal

☐(e) An understanding of professional, ethical, legal, security and social issues and responsibilities

☐(f) An ability to communicate effectively with a range of audiences

☐(g) An ability to analyze the local and global impact of computing on individuals, organizations, and society

☐(h) Recognition of the need for and an ability to engage in continuing professional development

☐(i) An ability to use current techniques, skills, and tools necessary for computing practice.

☐(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices. [CS]

☐(k) An ability to apply design and development principles in the construction of software systems of varying complexity. [CS]

☐(j) An ability to use and apply current technical concepts and practices in the core information technologies of human computer interaction, information management, programming, networking, and web systems and technologies. [IT]

☐(k) An ability to identify and analyze user needs and take them into account in the selection, creation, evaluation, and administration of computer-based systems. [IT]

☐(l) An ability to effectively integrate IT-based solutions into the user environment. [IT]

☐(m) An understanding of best practices and standards and their application. [IT]

☐(n) An ability to assist in the creation of an effective project plan. [IT]

Course Learning Outcomes (CLOs):

1. Identify the basic components of a computer system.
2. Design an algorithm to solve a given problem using the top-down design approach.
3. Know the difference between call-by-value and call-by-reference parameters.
4. Understand the notion of procedural abstraction.
5. Understand and use the three basic programming structures: sequence, selection, repetition.
6. Use arrays, strings and pointers to manipulate data.

SOs and CLOs Mapping:

CLO/SO	a	b	c	d	e	f	g	h	i	j	k	l	m	n
CLO1		√												
CLO2		√												
CLO3	√													
CLO4			√											
CLO5			√											

CLO6			√										

No.	Topics	Weeks	Teaching hours
1	Introduction to Computers	1	3
2	Problem solving techniques	1	3
3	Introduction to C++ , Variables , Data types, Operators	2	6
4	Conditional statements	2	6
5	Loops concepts	2	6
6	Functions, call by value, Call by reference	2	6
7	Arrays, types of arrays, Arrays to functions	2	6
8	Pointers, strings	1	3
9	Introduction to classes	1	3
	Total	**14**	**42**

Textbook:

- C++: How To Program, Deitel and Deitel, 9th edition, Prentice Hall, 2013.

Essential references:

- The C++ Programming Language: Special Edition, BjarneStroustrup, 4th edition, Addison-Wesley Professional, 2013.
- C++ Programming: From Problem Analysis to Program Design, De D. S. Malik, Cengage Learning, 2012.
- C++ Programming for the Absolute Beginner, De Dirk Henkemans and Mark Lee, Thomson Course, Technology, 2009.
- C++: The Complete Reference, 4th Edition, by Herbert Schildt (Author) , Publisher: McGraw Hill Education (India) Private Limited; 4 edition (2 April 2003), ISBN-13: 978-0070532465