Course Name : Introduction to Algorithms and Programming

Course Code	Course Type	Regular Semester	Lecture (hours/week)	Seminar (hours/week)	Lab. (hours/week)	Credits	ECTS
CMP 111	A	Fall	3.00	1.00	0.00	3.50	5.00

Lecturer	Edlir Spaho, MSc
Assistant	
Course language	Albanian
Course level	Bachelor
Description	The course "Introduction to Algorithms and Programming" provides general knowledge on basic algorithms, block diagrams, steps for building and structure a program. Through this course students will learn modular programming techniques with the highest possible efficiency and effectiveness. This course will first address the basic concepts of algorithm and programming, to continue with the presentation and usage of standard data types, conditional and repetitive structures, functions, recursion, arrays and pointers. Also in this course will be treated main searching and sorting algorithms.
Objectives	This course aims to: • Familiarize students with the basic concepts of algorithm and programming. • Familiarize students with the basic algorithms, structure and steps for building a program. • Enable students to build basic object-oriented programs. • Explain the importance of algorithms in building programs as efficiently as possible. • Explain to students the elements, syntax, and basic tools used in a program. • Develop critical thinking in analyzing the algorithm and performance of a program.
Core Concepts	1. Algorithms, Flowcharts and Pseudocodes 2. Constants and Variables 3. Conditional Structures and Loops 4. Arrays 5. Files 6. Functions 7. Recursivity 8. Enumerations

Course Outline

Week	Topic
1	Introduction to Computers and C++ This topic provides an overview of the course, basic concepts of computer literacy and a general introduction to the C ++ programming language. The topic also covers the steps for creating, compiling, executing and saving a project created using a CodeBlocks IDE. (pg. 3-29)
2	Introduction to C++ Programming, Flowcharts, Algorithms and Pseudo codes The topic deals with the steps for building a program, the main structure of a program as well as the analytical and graphical representation of simple algorithms. (pg. 30-50)
3	Variables Declaration, Data Types, Operators and Expressions This topic deals with declaration of C ++ built in data types, rules for using identifiers, variables, constants, mathematical, relational, and logical operators, as well as increment and decrement operators. (pg. 50-72)
4	Data Formatting in C++ This topic treats the instructions used for reading and writing, the ways of formatting different types of data, writing with certain precision, displaying exponential data, etc... (pg. 74-103)
5	Conditional Structures This topic deals with conditional structures if, if-else, conditional operator, nested if structures, branching by using the switch command as well as labeling and jumping in different parts of the program. (pg. 124-174)
6	Repetitive Structures (Loops) This topic deals with what are repetitive structures (loops), their use, elements of a repetitive structures (loops), the main types of loop (for, while and do-while loop), labeling and jumping inside and outside loops as well as nested loops structures. (pg. 175-286)

7	Arrays and Matrices This topic treats the definitions of arrays and matrices, the declaration, reading and writing of their elements, the sum of their elements, the reading and writing of a certain element or elements as well as their displaying in a certain format. (pg. 104-123)
8	Midterm Exam
9	Operations with Vectors and Matrices This topic deals with different operations that can be performed with arrays or matrices such as finding certain elements, finding the element with the largest or smallest value, counting elements according to one or more criteria, sorting of elements and the building up the vector from the matrix and vice versa. (pg. 175- 286)
10	Functions This topic will address the declaration and calling of functions, construction of simple functions such as sum and product, functions with and without result, functions with and without formal parameters, declaration and calling of several functions simultaneously and multiple calling of a function . (pg. 287-310)
11	Functions and Recursion This topic deals with the concept of recursion, its implementation through functions such as factorial, void and inline functions, their implementation with various examples and macro functions. (pg. 344-352, 373-384)
12	Functions with Vectors and Matrices This topic deals with functions for number series, build in mathematical functions and user-generated mathematical functions, functions for string operations such as finding string length, copying partial or complete strings, merging two strings and declaring of local and global variables (pg. 353-373)
13	Enumerations This topic deals with the definition and utilization of enumeration group, the use of loops with numbered values, different operations with enumerations as well as the creation and use of multiple enumerations. (Literature – 2, pg. 4-33)
14	Files in C++ This topic deals with creating and saving header files, calling/including files in the program (with #include) from the current folder and from any folder, saving of functions as header files and using them in other programs. (Literature – 2, pg. 317-330)
15	Review and Projects Presentation
16	Final Exam

Prerequisites	The student must attend the course at a minimum rate of 75%.
Literature	• Agni Dika - Bazat e programimit ne C++, 2005 • Cikël Leksionesh
References	• D. S. Malik C++ Programming, From Problem Analysis to Program Design, 2017 • Deitel H. and Deitel P., “C++ How to Program”, 9th Edition, Prentice Hall, 2014

Course Outcome

1	Students will be able to understand the principles of flowcharts and algorithms.
2	Students will be able to understand and use the key elements of the C ++ language.
3	Students will be able to learn how to program in C ++ using conditional structures, loops, functions, arrays, matrices and files.
4	Students will be able to use various programming techniques to build programs (or program modules) with high efficiency.
5	Students will be equipped with the necessary terminology and techniques to proceed with other subsequent subjects such as Object Oriented Programming, as well as other programming languages.

Course Evaluation

In-term Studies	Quantity	Percentage
Midterms	1	30
Quizzes	0	0
Projects	1	20
Term Projects	0	0
Laboratory	0	0
Class Participation	1	10
Total in-term evaluation percent		**60**
Final exam percent		**40**
Total		**100**

ECTS Workload (Based on Student Workload)

Activities	Quantity	Duration (hours)	Total (hours)
Course duration (Including the exam week: 16x Total hours of the course)	16	4	64
Study hours outside the classroom (Preparation, Practice, etc.)	14	3	42
Duties	1	0	0
Midterms	1	8	8
Final Exam	1	12	12
Other	1	8	8
Total Work Load			**134**
Total Work Load / 25 (hours)			**5.36**
ECTS			**5.00**