



University of Nicosia, Cyprus

Course Code COMP-113	Course Title Programming Principles II	ECTS Credits 6
Department Computer Science	Semester Fall, Spring	Prerequisites COMP-111
Type of Course Required	Field Computer Science	Language of Instruction English
Level of Course 1 st Cycle	Year of Study 1 st	Lecturer(s) Dr Athena Stassopoulou
Mode of Delivery Face-to-face	Work Placement N/A	Co-requisites None

Objectives of the Course:

The main objectives of the course are to:

- build on the knowledge already acquired in Programming I, by focusing on the more advanced concepts of procedural programming.
- enable the students to develop algorithmic thinking and problem solving.
- introduce dynamic memory allocation.
- provide practical experience in manipulating data strings, arrays, pointers and structures.
- provide the fundamentals of recursion.
- introduce students to I/O file stream and data files.
- introduce Object-Oriented Programming.

Learning Outcomes:

After completion of the course students are expected to be able to:

1. use built-in data structures to organize and process information.
2. develop algorithms and choose suitable data structures to produce solutions for complex problems using procedural programming.
3. develop working programs using the more advanced concepts such as pointers and arrays of structures.
4. develop programs using dynamic memory allocation.
5. describe and use recursion.
6. develop programs that use I/O file streams and data files.
7. apply the fundamentals of Object-Oriented Programming.
8. develop simple object-oriented programs.
9. develop applications consisting of multiple source files.

Course Contents:

1. User defined data types, Enumeration.
2. Arrays (one dimensional, two dimensional), C-Strings.
3. Addresses and Pointers. Dynamic memory allocation, array names as pointers, pointer arithmetic, passing addresses, passing arrays, pointer arrays.

4. Data Structures: Structures, arrays of structures, structures as function arguments, passing a pointer, returning structures, dynamic data structure allocation, searching an array of structures.
5. Recursion. How the computation is performed by use of the stack, recursion versus iteration.
6. I/O File Streams and Data Files: I/O File Stream Objects and Methods, reading and writing character-based files, exceptions and file checking.
7. Introduction to Object Oriented Programming: Introduction to Objects and Classes, class functions.

Learning Activities and Teaching Methods:

Lectures, Lab Presentations, Lab Tutorials, Practical Exercises and Assignments

Assessment Methods:

Homework, Projects, Midterm Exam, Final Exam.

Required Textbooks/Reading:

Authors	Title	Publisher	Year	ISBN
Gary J. Bronson	Program Development and Design using C++	Thomson Course Technology	2006	0-619-21677-8

Recommended Textbooks/Reading:

Authors	Title	Publisher	Year	ISBN
Daniel Y. Liang	Introduction to Programming with C++, 2 nd edition	Pearson	2010	0-13-136587-8
Deitel & Deitel	C++ How to Program, 2nd edition	Prentice Hall Inc.	1998	0-13-528910-6
Bjarne Stroustrup	The C++ Programming Language, 3rd edition,	Addison-Wesley	1997	0-201-88954-4
<u>J. LaJoie, S. B. Lippman & B. E. Moo</u>	C++ Primer	Addison-Wesley	2005	0-201-72148-1