



Code: ING-INF/05

Credits: 9

Matter: Information Processing Systems

Main language of instruction: Italian

Other language of instruction: English

Teaching Staff

Head instructor

Dr. Salvatore MONTELEONE - salvatore.monteleone@unicusano.it

Introduction

1. Objective of the course:

The Information Processing Systems Course aims to provide the student with a good knowledge of the principles that govern structured programming. The course introduces the basic concepts of binary information representation and processing. Furthermore, the educational objective of the course is to provide the student with detailed knowledge on the construction of algorithms through the management of flow control, and on data structures. The activities associated with the Course develop the skills necessary to implement algorithms and data structures in complete programs through the use of the C programming language.

Objectives

2. Course Structure:

- Basic concepts of information representation and processing from the bit level.
- Basic concepts, and related models, necessary for understanding the structure of programmable digital electronic calculators.
- Basic concepts in the programming of electronic computers.
- Program development techniques according to the structured programming paradigm through the use of the C language.
- Approach, development, construction of algorithms and data structuring and management.

Competencies:

A. Knowledge and understanding

At the end of the course, the student will have both theoretical and design knowledge regarding structured programming in the C language and the basic

architectures of modern electronic computers. Furthermore, the student will acquire the knowledge of elements of information representation in computers. Through the e-tivities, students will acquire the ability to use the most common programming tools.

B. Application of knowledge

The student will be able to evaluate possible software solutions to moderately complex problems, and tackle their implementation; The student will also be able to apply the principles learned to tackle simple problems of the binary representation of information. The e-tivities provide for the application of theoretical knowledge to the implementation of simple C language programs, using the appropriate tools and development environments.

C. Making judgments

The student will be able to deal independently with the analysis, design, and implementation of software based on structured programming and will also be able to evaluate the quality of its solutions, in terms of simplicity, readability, modularity, efficiency, and suitability for reuse.

D. Communication skills

The student will be able to expose, effectively and with language properties, analysis and solutions of problems that can be faced with structured programming, as well as problems of information representation, using adequate terminology.

E. Learning skills

At the end of the course, the student will have knowledge of the fundamental notions necessary to independently address structured programming problems by identifying and integrating partial solutions already available, both formalized and implemented. This will allow the student to independently deepen the knowledge of software modules and programming interfaces, providing useful bases to learn what will be proposed in the specialized courses of computer engineering, with particular reference to topics related to the knowledge of programming languages and paradigms, as well as computer architectures.

Syllabus

3. Programme of the course:

Subject 1. Numeral systems:

Binary numeral system;

Bits, bytes and multiples;



Binary-to-decimal conversions;
Arithmetic operations in the binary numeral system;
Octal numeral system;
Hexadecimal numeral system;
Codes;
BCD codes;
Representation of signed integers.

Subject 2. Information representation and computer architecture:

Complement representation;
Representation of real numbers: fixed-point, floating-point;
Representation of alphanumeric characters;
ASCII;
Representation of images;
Structure of a programmable digital electronic computer;
Von Neumann model: CPU, memory, input/output, bus;
Microprocessors.

Subject 3. Introduction to C programming:

Introduction to programming;
Introduction to programming in C - Arithmetic and decision making;
Development of a structured program in C.

Subject 4. Flow control and functions:

Flow control statements in C;
Functions and recursion;

Subject 5. Arrays:

Arrays in C, searching and sorting;
Manipulation of vectors and arrays.

Subject 6. Pointers:

Pointers in C;
Pointers' manipulation.

Subject 7. Structured data types and files:

Structures, unions, bit manipulation and enumerations in C;
File processing in C.

Subject 8. Dynamic data structures:

Dynamic data structures in C;
Handling dynamic data structures.

Evaluation system and criteria

The examination consists of a written test divided in two parts:

1. 3 Exercises on information representation and processing (5 marks each, for a total of 15 out of 30 marks)
2. Development of a structured program in C (for a total of 10 out of 30 marks)

In addition, there are six e-tivities, consisting of practical and theoretical problems. These need to be sent to the instructor in advance of the examination. Each e-tivity counts 1 mark for a total of 6 out of 30 marks.

Bibliography and resources

4. Materials to consult:

Notes written by the instructor, to cover the course contents and examination programme, will be available in English.

5. Recommended bibliography:

B. W. Kernighan; D. M. Ritchie, "The C Programming Language (Second Edition)", Pearson Prentice Hall.