

Information Technology - basic - syllabus

Course period: 4th module (April - May 2018)

Course description: This course will give practical knowledge about computers and computer networks. It will start with a general description of computers, discussing both hardware and software general concepts. It will then give an introduction to programming, with some conceptual and theoretical notions for “sound programming”, and a working knowledge of the python programming language. It will also give some knowledge to effectively use the online resources, such as academic databases and advanced search engines techniques. The classes will be in part standard lectures, and in part “computer lab” sessions, done in the computer rooms.

The aim of this course: This basic IT course has the goal to give to the students a set of basic tools, both conceptuals and practicals, to make them able to use Information Technology for applications in fields such as Social Sciences, Humanities, etc.

Objectives:

- Basic computer hardware understanding and troubleshooting.
- Basic understanding of concepts and technical definitions of Computer science and computer networks.
- Working knowledge of programming techniques, with emphasis on applications on Social Sciences and Humanities.
- Python language programming, theory and practice.

The course structure (number of lectures, seminars): The course will develop over 8 weeks, with **32 classes** of 90', of which **13** will be in the form of lectures (frontal teaching) and **19** will be in the form of Computer Lab Practice.

Author of the syllabus: Fabio Grazioso

Lecturer (name and surname): Fabio Grazioso

Seminar's professor (name and surname): Fabio Grazioso

Course language: English

Course requirements: No knowledge of computer will be assumed. Some basic mathematics (algebra) knowledge is required.

Ground rules: Attendance to 85% of the lectures is required. 1 intermediate assignment (homework) and 1 final assignment will be required.

Sample exam's questions: What is a general definition of an Operative System? What is the meaning of “data packet” talking of computer network protocols? What is the name of the graphic library used for data plotting in Python? Write a sorting algorithm using pseudo-code.

Reference Textbooks and other material:

- 1) Fox - Information Technology: An Introduction for Today's Digital World (2013).
- 2) Vermaat, Sebok, Freund - Discovering Computers 2017 (2016).
- 3) Tanenbaum, Bos - Modern Operating Systems (2014).
- 4) Harris, Harris - Digital Design and Computer Architecture (2nd Ed) (2012).
- 5) Mueller - Windows command line administration instant reference (2010).
- 6) Gillam - Unicode Demystified (2002).
- 7) Hunt - TCP-IP network administration (2002).
- 8) Martin - Clean Code- A Handbook of Agile Software Craftsmanship (2008)
- 9) Cormen - Algorithms unlocked (2013).
- 10) Lutz - Learning Python (2013).

- 11) Friedl - Mastering Regular Expressions (2006).
- 12) Baecker - Readings in human-computer interaction (1995).
- 13) Norman - The Design of Everyday Things (2002).

(note: the books list is very long, but from most of the books only small parts, sometimes just few pages, will be considered.)

In the following you can find a summary table of the main topics and the detailed schedule.

Information Technology basic - course summary

<p>Hardware architecture. General architecture of a computer. The Von Neumann machine. The representation of information in a computer (hardware). The machine representation of numbers and errors (logical).</p>
<p>Software essentials. The operating system. MS-DOS, Linux/Unix. Graphical Operating Systems: Microsoft Windows, Apple MacOS.</p>
<p>Application software and Utility software analysis. Word processors. Spreadsheets. Presentation. Graphic editors. Audio editors. Text editors. Utility software, examples.</p>
<p>Networks. Networking protocols. TCP/IP protocol. World Wide Web, HTML, browsers. Search engines.</p>
<p>Introduction to programming. Algorithms. Main programming statements and pseudo-code. Procedural programming, structured programming. Elements of object-oriented programming.</p>
<p>Discrete Mathematics. Examples of algorithms: sort, search.</p>
<p>Programming languages. Main programming language definitions. Syntax and semantics.</p>
<p>Elements of the Python language. Keywords and syntax. Control flow statements. Data types. Libraries: NumPy, SciPy, Matplotlib. Examples of data processing. Examples of data plotting and representation: 2D plots, 3D plots, annotations. Datafiles input and output. Strings manipulation.</p>

Course classes

(type: L= lecture, P = practice)

#	type	topics
1	L	BL1 General architecture of a computer. The Von Neumann machine. The representation of information in a computer (hardware).
2	L	BL2 The machine representation of numbers and errors (logical). The binary numbers, the hexadecimal numbers. Encodings.
3	L	BL3 The operating system in general. MS-DOS, Linux/Unix. Graphical Operating Systems: MS Windows.
4	L	BL4 Word processors. Spreadsheets. Presentation. Graphic editors. Audio editors. Text editors. Utility software, examples.
5	P	BP1 Computer Lab practice on word processing, spreadsheets, presentation software.
6	P	BP2 Computer Lab practice on graphic editing (both vector and raster) for academic applications.
7	L	BL5 Networking protocols, layers, packets. TCP/IP protocol. World Wide Web, HTML, browsers. Search engines. Other protocols: email, FTP, servers, clients.
8	P	BP3 Computer Lab practice on internet resources: academic resources, advanced search engines practice.
9	L	BL6 Main programming language types: machine code, low-level, high-level, interpreted, compiled. Syntax and semantics.
10	L	BL7 Programming environments: editor, debugger, compiler, interpreter. Software design techniques. Algorithms.
11	L	BL8 The Python language. Keywords and syntax. Control flow statements. Data types.
12	P	BP4 Computer Lab practice on Python programming: first simple algorithms.
13	L	BL9 Python Libraries: NumPy, SciPy. Examples of mathematical functions, mathematical data generation and manipulation. Examples of simple scientific plots.
14	P	BP5 Computer Lab practice on Python programming: first scientific applications.
15	P	BP6 Computer Lab practice on Python programming: more scientific applications.
16	L	BL10 Python data processing. Examples of data plotting and representation: 2D plots, 3D plots, annotations.
17	P	BP7 Computer Lab practice on Python programming: scientific data plotting.

18	P	Computer Lab practice on Python programming: scientific data plotting.
19	L	Python datafiles input and output. Text files and binary files. Strings manipulation.
20	P	Computer Lab practice on Python programming: data files input and output.
21	P	Computer Lab practice on Python programming: data files input and output.
22	L	More strings manipulation with Python. Regular Expressions.
23	L	Discrete Mathematics. Algorithms: sort, search.
24	P	Computer Lab practice on Python programming: Examples from the course.
25	P	Computer Lab practice on Python programming: Examples from the course.
26	P	Computer Lab practice on Python programming: Examples from the course.
27	P	Computer Lab practice on Python programming: Examples from the course.
28	P	Computer Lab practice on Python programming: Examples from the course.
29	P	Computer Lab practice on Python programming: Examples from the course.
30	P	Computer Lab practice on Python programming: Examples from the course.
31	P	Computer Lab practice on Python programming: Examples from the course.
32	P	Computer Lab practice on Python programming: Examples from the course.