

INSTITUTION	NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS																			
SCHOOL	SCHOOL OF SCIENCE																			
DEPARTMENT	INFORMATICS AND TELECOMMUNICATIONS																			
COURSE LEVEL	UNDERGRADUATE																			
COURSE TITLE	Systems Programming																			
COURSE CODE	K24	Semester	5	ECTS	8															
TEACHING HOURS per week	THEORY	4	SEMINAR.		LABORATORY															
COURSE TYPE	<p>Select one of the following and delete the rest Compulsory</p> <table border="1"> <tr> <td>K</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table> <p><i>Fill the table as in the curriculum: Track (A-Computer Science, B- Computer Engineering) / Specialization Compulsory (Y) / Core Specialization (B)/ Elective Specialization (E)</i></p>						K													
K																				
URL	http://cgi.di.uoa.gr/~mema/courses/k24/k24.html http://cgi.di.uoa.gr/~antoulas/k24/																			
EXPECTED PRIOR KNOWLEDGE/ PREREQUISITES AND PREPARATION:	K08 Data Structure and Programming Techniques, Recommended K14																			
TEACHING AND EXAMINATIONS LANGUAGE:	GREEK																			
THE COURSE IS OFFERED TO ERASMUS STUDENTS																				

COURSE CONTENT
<p>This course examines in depth the Unix environment as a development environment. We will look at the Linux API for the C / C ++ languages as well as the Linux shell. We will cover topics such as: basic Unix commands, shell programming, script languages, programming of system functions in C / C ++ for error handling, creation and termination of processes, sending / receiving signals, low-level input / output system calls, communication between local processes, creation, termination and synchronization of threads, file system management, as well as network programming. This course requires independent and consistent effort from the student.</p>

STUDENT LEARNING OBJECTIVES

Upon successful completion of the course the student will be able to:

- Use basic Unix commands and navigate/manage a Unix file system
- Develop shell programs and program with script languages
- Explain and program Unix inter-process communication mechanisms
- Design, implement, and evaluate multi-threaded programs
- Design, implement and evaluate network programs (Internet applications)

TEACHING AND LEARNING METHODS – ASSESSMENT

TEACHING METHOD

In Class (Face to Face)

USE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

Learning process supported by class web page and Piazza class forum. Posted class materials include: course description and syllabus, slides, announcements, programming assignments, discussions on programming assignments, and pointers to external links.

TEACHING ORGANIZATION

Describe in detail the way and methods of teaching:

*Enhanced Lectures,
Online Lectures,
Seminars,
Tutorial,
Laboratory,
Laboratory Exercise,
Study & analysis of literature,
Practice (Positioning),
Interactive teaching,
Developing a project,
Individual / group work
Telework (reference to tools) etc.*

Details of the student's study hours for each learning activity and hours of non-guided study are shown to ensure that the total workload at the semester corresponds to the ECTS

Activity	Student Workload (hours)
Lectures	52
Programming Assignments	86
Independent Study	12
Total Course (25 hours of workload per unit of credit)	150

ASSESSMENT OF STUDENTS

Description of the assessment process

Assessment Methods, Formative or Concluding, Multiple Choice Test, Quick Response Questions, Test Development Questions, Problem Solving, Written Work, Report / Report, Oral Examination, Public Presentation, Laboratory Work, Other / Other

Fully defined evaluation criteria are mentioned and if and where they are accessible to students.

Students are evaluated via final written exam and programming assignments. Programming assignments are graded via oral exam. Re-grades of exam and assignments are possible.

Assessment methods	Number	Percentage
Written examination	1	50%
Programming assignments	3-4	50%

LITERATURE AND STUDY MATERIALS / READING LIST

Peter S. Pacheco. M.J. Rochkind, *Advanced Unix Programming*, Prentice-Hall Software Series, Englewood Cliffs, NJ, 2004.

Kay Robbins, Steven Robbins, "Unix Systems Programming: Communication, Concurrency, and Threads", Prentice Hall PTR, 2003.