Universitatea Națională de Știință și Tehnologie Politehnica București
Facultatea de Electronică, Telecomunicații și
Tehnologia Informației

COURSE DESCRIPTION

1. Program identification information

1.1 Higher education institution	**National University of Science and Technology Politehnica Bucharest**
1.2 Faculty	**Electronics, Telecommunications and Information Technology**
1.3 Department	**Electronic Devices, Circuits and Architectures**
1.4 Domain of studies	Electronic Engineering, Telecommunications and Information Technology
1.5 Cycle of studies	Masters
1.6 Programme of studies	Advanced Computing in Embedded Systems

2. Date despre disciplină

2.1 Course name (ro) (en)				Sisteme de operare Operating Systems			
2.2 Course Lecturer				Prof. Dr. Marius-Constantin Vochin			
2.3 Instructor for practical activities				Prof. Dr. Marius-Constantin Vochin			
2.4 Year of studies	1	2.5 Semester	I	2.6. Evaluation type	E	2.7 Course regime	Ob
2.8 Course type		DS	2.9 Course code	UPB.04.M1.O.22-01		2.10 Tipul de notare	Nota

3. Total estimated time (hours per semester for academic activities)

3.1 Number of hours per week	3	Out of which: 3.2 course	1.00	3.3 seminary/laboratory	2
3.4 Total hours in the curricula	42.00	Out of which: 3.5 course	14	3.6 seminary/laboratory	28
Distribution of time:					hours
Study according to the manual, course support, bibliography and hand notes Supplemental documentation (library, electronic access resources, in the field, etc) Preparation for practical activities, homework, essays, portfolios, etc.					40
Tutoring					0
Examinations					4
Other activities (if any):					0

3.7 Total hours of individual study	58.00
3.8 Total hours per semester	100
3.9 Number of ECTS credit points	4

4. Prerequisites (if applicable) (where applicable)

4.1 Curriculum	Computer Programming, Data Structures and Algorithms, Computing Systems Architecture, Microprocessor Architecture

4.2 Results of learning	Programming knowledge in C/C++

5. Necessary conditions for the optimal development of teaching activities (where applicable)

5.1 Course	Internet
5.2 Seminary/ Laboratory/Project	Mandatory attendance at the laboratories (according to the regulations for master's degree studies in UPB).

6. General objective *(Reffering to the teachers' intentions for students and to what the students will be thought during the course. It offers an idea on the position of course in the scientific domain, as well as the role it has for the study programme. The course topics, the justification of including the course in the currcula of the study programme, etc. will be described in a general manner)*

The discipline aims to study the internal structure of the Linux operating system and the methods to design and implement system software in the Linux infrastructure. Knowledge of the resource management mechanisms of computer systems (process management and communication mechanisms between processes, memory, input / output equipment and file management) and the structure and functioning of system utilities as tools for developing applications in computer systems.

7. Competences *(Proven capacity to use knowledge, aptitudes and personal, social and/or methodological abilities in work or study situations and for personal and proffesional growth. They refflect the empolyers requirements.)*

Specific Competences	C1.1 Using the Linux operating system and the Bash scripting library C1.2 Implementation of system software for interaction with peripherals C1.3. Using Linux kernel facilities to measure and optimize program performance
Transversal (General) Competences	CT1 CT3 Learning how to use, analyze and optimize a Linux-based computing system, to increase productivity and fulfill professional tasks.

8. Learning outcomes *(Synthetic descriptions for what a student will be capable of doing or showing at the completion of a course. The learning outcomes reflect the student's acomplishments and to a lesser extent the teachers' intentions. The learning outcomes inform the students of what is expected from them with respect to performance and to obtain the desired grades and ECTS points. They are defined in concise terms, using verbs similar to the examples below and indicate what will be required for evaluation. The learning outcomes will be formulated so that the correlation with the competences defined in section 7 is highlighted.)*

Knowledge	*The result of knowledge aquisition through learning. The knowledge represents the totality of facts, priciples, theories and practices for a given work or study field. They can be theoretical and/or factual.* Knowledge of operating systems, especially for the embedded environment, principles of operation and implementation.
Skills	*The capacity to apply the knowledge and use the know-how for completing tasks and solving problems. The skills are described as being cognitive (requiring the use of logical, intuitive and creative thinking) or practical (implying manual dexterity and the use of methods, materials, tools and intrumentation).* Skills related to the use of operating systems to create applications and integrated solutions.

Responsability and autonomy	*The student's capacity to autonomously and responsably apply their knowledge and skills.* Diagnostics of operating systems and implementation of software applications.

9. Teaching techniques *(Student centric techniques will be considered. The means for students to participate in defining their own study path, the identification of eventual fallbacks and the remedial measures that will be adopted in those cases will be described.)*

Teaching is done through a projector and at the blackboard; Course materials will be published through the Moodle platform. Students gain practical skills in using the Linux operating system and system programming. The didactic materials are the laboratory platforms and course materials.

10. Contents

COURSE

Chapter	Content	No. hours
1	Definitions, concepts, general structure of an operating system in a computer system, characteristics of system software, functions, classifications, history.	1
2	Communications between processes, mutual exclusion, semaphores, monitors, equivalences.	1
3	Process management, tasks, execution threads, interlocking, planning algorithms.	1
4	Management of inputs and outputs: management of interruptions, device drivers, resource planning, treatment of blockages on resources. Indexed search.	1
5	Memory management, memory sharing, memory space management, allocation algorithms, virtual memory, paging, page allocation and deallocation algorithms, working set.	2
6	External memory management, space management on external media, file organization and management, optimization algorithms.	1
7	The networking stack of the Linux operating system.	2
8	Network stack performance and latency optimizations.	2
9	Implementing memory and input-output (IO) space-mapped device drivers	2
10	Real-time processes.	1
	Total:	14

Bibliography:

1. Andrew Tanenbaum „Structured Computer Organization” 6th edition, Pearson, 2012
2. Andrew Tanenbaum „Operating systems” 2nd edition, Prentice Hall, 1997

LABORATORY

Crt. no.	Content	No. hours
1	Introduction to the Linux operating system	2

2	The Bash command interpreter	4
3	C Programming in Linux (Userspace)	4
4	Linux-specific profiling and optimization	4
5	Network infrastructure configuration	4
6	Network infrastructure testing	4
7	Kernel programming (Kernelspace)	4
8	Final laboratory colloquium	2
	Total:	28
Bibliography:		

11. Evaluation

Activity type	11.1 Evaluation criteria	11.2 Evaluation methods	11.3 Percentage of final grade
11.4 Course	- knowledge of fundamental theoretical notions; - knowledge of how to apply the theory to specific problems;	The written verification test and the subjects cover the entire subject, making a synthesis between the comparative theoretical study of the subject and the explanation of the application models through exercises and problems.	50%
	- knowing the details of applying the theory to specific problems;	Homework, customized according to each student's interest for each chapter, during the semester.	20%
11.5 Seminary/laboratory/project	- knowledge of basic data structures and types of algorithms for their use in solving a given problem; - knowing how to use some tools for developing, simulating and debugging programs - demonstration of the functioning of the completed programs	Final laboratory colloquium, comprising a theoretical component and a practical component. The theoretical component is verified by an oral test; the practical component is evaluated by checking the solution method (implementation, testing, operation) by the student of a practical problem and oral discussion.	30%
11.6 Passing conditions			
achievement of the course objective, proven by obtaining at least 50% of each score			

12. Corroborate the content of the course with the expectations of representatives of employers and representative professional associations in the field of the program, as well as with the current state of knowledge in the scientific field approached and practices in higher education institutions in the European Higher Education Area (EHEA)

The industry of designing, implementing and testing electronic systems requires university graduates to be proficient in the use of the Linux operating system. Being the nerve center of any electronic system, the operating system makes the logical connection between all its components. A good understanding of the subsystems of the Linux kernel allows an operating system designer to integrate new components into the system, optimize the operation of already integrated ones, and implement an efficient and accurate system performance testing mechanism.

The course curriculum provides the graduates with adequate competences with the expressed needs of employers, allowing them to quickly integrate into a professional team in Romania or abroad without the need for additional training. Also, the knowledge presented to students allows them to approach some advanced research topics in the field of operating systems, within the doctoral programs of UPB or other universities.

Date	Course lecturer	Instructor(s) for practical activities
09.09.2022	Prof. Dr. Marius-Constantin Vochin	Prof. Dr. Marius-Constantin Vochin

Date of department approval	Head of department
31.10.2024	Prof. Dr. Claudius DAN

Date of approval in the Faculty Council	Dean
01.11.2024	Prof. Dr. Mihnea Udrea