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 Technology and Reliability**
1.4 Domain of studies	Computers and Information Technology
1.5 Cycle of studies	Bachelor/Undergraduate
1.6 Programme of studies	Information Engineering

2. Date despre disciplină

2.1 Course name (ro) (en)				Sisteme de operare 1 Operating Systems 1			
2.2 Course Lecturer				S.l./Lect. Dr. Valentin Voiculescu			
2.3 Instructor for practical activities				S.l./Lect. Dr. Valentin Voiculescu			
2.4 Year of studies	1	2.5 Semester	II	2.6. Evaluation type	V	2.7 Course regime	Ob
2.8 Course type		D	2.9 Course code	04.D.02.O.714		2.10 Tipul de notare	Nota

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

3.1 Number of hours per week	3.5	Out of which: 3.2 course	2.00	3.3 seminary/laboratory	1.5
3.4 Total hours in the curricula	49.00	Out of which: 3.5 course	28	3.6 seminary/laboratory	21
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.					43
Tutoring					3
Examinations					5
Other activities (if any):					0

3.7 Total hours of individual study	51.00
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 and Programming Languages 1 Algorithms and Data Structures
4.2 Results of learning	Basic requirements of installing programs, using editors to write source code and making computer programs.

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

5.1 Course	The course will be held in a room equipped, preferably, with a video projector and internet access (to allow simultaneous use in the form of a Teams video conference). Students work on their own computers.
5.2 Seminary/ Laboratory/Project	Attendance of laboratory sessions is mandatory (as stated by the regulations for undergraduate university studies). Students may work on their own computers during sessions in rooms with internet access.

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 has the general objective of familiarizing students with the facilities of the operating systems in general, the concept of virtual machines, and in particular with the Linux operating system and the Bash scripting language, modern technologies often used in the IT&C domain.

In class and lab, students will become familiar with the basics and advanced elements of the the operating systems, with the concept of using virtual machines, practiced by running the Linux operating system on the student's computer.

The study will combine the teaching of concepts with the completion of exercises by the student, individually, on the personal computer. Students will become familiar with the Linux operating system, its features, and working in the command line. The activities are aimed at developing software abstraction skills, identifying and practicing the skills necessary for computer modeling of some situations from reality.

Based on the knowledge gained from this course, the future electrical engineer will be able to implement or modify programs or command-line scripts specific to modern, scriptable and automatable software development and testing activities, being able to create programs from requirements specification to execution, debugging and interpretation of results.

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	**C3.** Application of basic knowledge, concepts and methods regarding the architecture of computer systems, microprocessors, microcontrollers, programming languages and techniques.
Transversal (General) Competences	**CT1.** Methodical analysis of the problems encountered in the activity, identifying the elements for which there are established solutions, thus ensuring the fulfillment of professional tasks. **CT3.** Adapting to new technologies, professional and personal development, through continuous training using printed documentation sources, specialized software and electronic resources in Romanian and, at least, in one language of international circulation.

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.* Is able to answer a series of questions based on notions accumulated and activities carried out during the semester, as part of an examination. Describes the command, action or succession of commands and actions neecssary to perform an operation, at the operating system level. Describes the command (or set of commands) and accompanying arguments required to be used to solve a specific command-line problem. Describes, explains, highlights the consequences of running one (or more) commands in the command line. Describes concepts from the Bash environment needed to solve a given problem under Linux on the command line.
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).* Installing Linux under a virtual machine, starting, interacting with, shuting down the Linux operating system in the command line. Listing the major variants of Linux distributions and identifying the types of packages used by their included package managers. Designing scripts and programs starting from a requirement, set of commands. Checking and debugging the validity of a given script. Identifying the type of privilege found on the command line, the current location in the file structure, editing and manipulates files, running and chaining commands. Performing searches. Viewing details for active processes and sending signals.
Responsability and autonomy	*The student's capacity to autonomously and responsably apply their knowledge and skills.* Demonstrating responsiveness to new learning contexts. Conspecting in advance the course materials, laboratory, to the extent that they are made available. In case of absence, going through the material taught, made available, by oneself. Solving homework individually, autonomously, respecting academic ethics. Respecting the principles of academic ethics, individually carrying out the activities marked in this sense, also correctly citing the bibliographic sources used, if the situation requires it. Applying principles of professional ethics/deontology in the analysis of the technological impact of the solutions proposed in the specialized field on the environment.

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.)*

The didactic materials used are the course notes and presentations, also available in electronic format.
Starting from the analysis of the students' learning characteristics and their specific needs, the teaching process will explore both expository (lecture, exposition), problem-solving and conversational-interactive teaching methods, based on action-based learning models, such as exercise, practical activities and problem solving. Interactivity with students through the associated applied activities. Intervals are reserved for presentation, analysis and solving of some practical problems (reality modeling).
Lectures will be used in the teaching activity, based on Power Point presentations, which will be presented in front of the students as far as is technically possible, or/and through a videoconferencing environment such as Teams. These will be made available to students. Each course will start with a short recap of the previous lesson to ensure continuity of the concepts covered.
The presentations use, as far as possible, examples of real-life application of the concepts taught, so that the information presented is easy to understand and assimilate.

In the applied section, teaching is based on the use of the expository method (covering the communication and demonstrative function). The dialogue during the course is also extended during the application sessions. These are necessary to prepare students for homework and verification tests along the way.
Feedback will also be used, as a way of adapting the pedagogical approach to the students' learning needs.

10. Contents

COURSE		
Chapter	**Content**	**No. hours**
1	Introductory information. The role of the Operating System. History. Operating system examples (e.g.: Windows, Linux, BSD, Android). Operating system interfaces. Installing a Linux virtual machine on the student's computer. Partitioning. Formatting. Checking the installation. Distributions. Package managers. Basic notions of virtualization. Usage scenarios.	4
2	Working in command line. Shell. Prompt. Commands and arguments. Facilities. Chaining command.	2
3	The architecture of the computing system. Processors. Memory. Input/output devices.	4
4	The architecture of the operating system. Monolithic kernel. Microkernel. Drivers. Powering up the system.	4
5	The Linux file system. Using the file system. I/O interaction. Command line editors.	2
6	Users. Permissions. Basic notions related to connecting to the internet, connecting to the machine. Processes – view, interact with Linux processes. Signals. Automating tasks via scripting.	3
7	Advanced aspects of operating systems. Management. Virtual file system. Mounting. Scheduling. Governance. Energy efficiency. Security. Virtualization. Containerization.	3
8	Developing applications. Compiling. Linking. Interpreting. Integrated development editors. Source code management systems. Open vs free. Software licenses. Recap session.	4
9	Final verification	2
	Total:	28

Bibliography:
V.G.Voiculescu, Sisteme de Operare, electronic course materials, Moodle, 2024
R.Deaconescu, R.Rughiniș, M.Carabaș, A.Radovici, Utilizarea sistemelor de operare, 2024
J. Cannon, Linux for Beginners, ACM, 2014
TLDP, https://tldp.org/LDP/intro-linux/html/intro-linux.html, 2024
TLDP, https://tldp.org/LDP/Bash-Beginners-Guide/html/, 2024
W.R. Stevens, S.A. Rago, Advanced programming in the UNIX environment, 3rd edition, 2013

LABORATORY		
Crt. no.	**Content**	**No. hours**

1	Installing under virtual machine. Validating the install. Command line interfaces. Windows vs Linux. Elements of a shell prompt. Simple commands.	3
2	Users. Permissions. Using the file system in Linux. Interesting portions of the file system hierarchy.	3
3	System management. Virtual file systems. Data about the system. Affinity. Energy efficiency (DVFS, idle). Mounting. Debugging. Activity planning.	3
4	Connecting to the internet. Elements of network configuration. Connecting to the machine. SSH. Passwords. Keys. Processes – view, interact with Linux processes. Process paralelism. Signals.	3
5	Scripting using the Linux shell. Verifying the successful execution of a command. Functions. Loops. Regular expressions.	3
6	Elements of an embedded platform. Stages of boot-up for an operating system. Image flashing. Virtualization vs containerization. Recap.	3
7	Colloquium	3
	Total:	21

Bibliography:

V.G.Voiculescu, Sisteme de Operare, electronic course materials, Moodle, 2024
V.G.Voiculescu, Sisteme de Operare, electronic lab materials, Moodle, 2024
R.Deaconescu, R.Rughiniș, M.Carabaș, A.Radovici, Utilizarea sistemelor de operare, 2024
J. Cannon, Linux for Beginners, ACM, 2014
TLDP, https://tldp.org/LDP/intro-linux/html/intro-linux.html, 2024
TLDP, https://tldp.org/LDP/Bash-Beginners-Guide/html/, 2024
W.R. Stevens, S.A. Rago, Advanced programming in the UNIX environment, 3rd edition, 2013

11. Evaluation

Activity type	11.1 Evaluation criteria	11.2 Evaluation methods	11.3 Percentage of final grade
11.4 Course	Knowledge of the means to apply the theory to specific situations	Ongoing tests, homework across the semester	40%
	Knowledge of fundamental notions, final evaluation	Final evaluation	20%
11.5 Seminary/laboratory/project	Knowledge of the Linux operating system, with its features and working in the command line. Solving various Linux command line issues.	The practical activity is constantly checked throughout the semester. Final check.	40%
11.6 Passing conditions			
Installing. Starting up. Interacting with the operating system, in particular with Linux through commands. Creating a script starting from a set of commands. Implementation of a program with specified facilities, modeling, programming, running successfully in order to get the solution. Obtaining 50% of the total score or the minimum score provided by the regulation.			

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 usage and understanding of operating systems has become a requirement in the modern world. Linux, in particular, has become fundamental in many areas: in telecommunications (where we are in the process of transitioning from expensive proprietary equipment to commodity hardware running specific stacks and protocols on top of a generic Linux operating system), similarly in broad and diverse areas in the embedded domain, IoT, including smartphones (Android runs over Linux), in automotive (In vehicle infotainment area), even in outer space (SpaceX/StarLink satellites).
Bash is in turn a language used in some scripting, testing, building frameworks found in the target fields of our graduates, and with the help of this subject the future electrical engineer will be able to implement or modify programs or command line scripts specific to development and testing activities modern, scriptable and automatable software.
Based on the knowledge gained from this course, the future electronics engineer will be able to implement or modify programs or command-line scripts specific to modern, scriptable and automatable software development and testing activities, being able to create programs from requirements specification to execution, debugging and interpretation of results.

Date	Course lecturer	Instructor(s) for practical activities
	S.l./Lect. Dr. Valentin Voiculescu	S.l./Lect. Dr. Valentin Voiculescu
Date of department approval	Head of department	
Date of approval in the Faculty Council	Dean	