COURSE DESCRIPTION CARD - SYLLABUS

Course name
Operating Systems [N1Inf1>SOP]

Course

Field of study
Computing

Year/Semester
1/2

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
part-time

Requirements
compulsory

Number of hours

Lecture
20

Laboratory classes
20

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

5,00

Coordinators

dr inż. Dariusz Wawrzyniak
dariusz.wawrzyniak@put.poznan.pl

Lecturers

Prerequisites

A student starting this module should have basic knowledge of functioning computer and imperative programming. Should have solving skills basic problems in the field of implementation and assessment of the cost of operation of simple algorithms and ability to obtain information from indicated sources. He should also understand the necessity expand their competences and demonstrate readiness to cooperate within a team. Moreover, in terms of social competences, the student must demonstrate attitudes such as honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

Course objective

1. Providing students with basic knowledge about the functioning of operating systems in the field of computer system resource management. 2. To introduce students to the concept of CPU allocation planning, memory management and input-output devices and file system organization. 3. Developing students' ability to solve simple management problems computer system, including the protection of system resources and information. 4. Developing students' skills in organizing processing, taking into account efficiency and optimal use of system resources.

Course-related learning outcomes

Knowledge:
1. Knows selected issues related to operating systems.
2. Knows issues in the field of algorithms and data structures and the basics of complexity theory computational.
3. Knows issues related to the applicability of combinatorial optimization algorithms.
4. Knows the principles of structured and object-oriented programming.

Skills:
1. Is able to design and create software using the services of the operating system in accordance with the given specifications, using appropriate methods, techniques and tools.
2. Is able to analyze the functionality and requirements of IT systems in context use of operating system services.
3. Is able to obtain information from literature, databases and other properly selected sources, as well in English language.

Social competence:
1. Understands the need for lifelong learning and improving competences.
2. Is able to cooperate and work in a group, taking on various roles in it.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment
a) In the scope of lectures, verification of the assumed learning outcomes is carried out by:
• answers to questions about the material covered in previous lectures.
b) In the field of laboratories, verification of the assumed learning outcomes is carried out by:
• assessment of skills related to the implementation of laboratory exercises,
• assessment of knowledge and skills related to the implementation of laboratory tasks through two colloquia in
semester.
Summative assessment
a) In the scope of lectures, verification of the assumed learning outcomes is carried out by:
• assessment of knowledge and skills demonstrated in a written test of a problem nature consisting of 4 - 5 open questions, with the possibility of obtaining 20 - 30 points for each of them and in total 100 points; To obtain a positive grade, you must score at least 50 points;
• discussion of the test results.
b) In the field of laboratories, verification of the assumed learning outcomes is carried out by:
• assessment of knowledge and skills related to the content provided in laboratories through final colloquium;
• a summary of grades awarded during the semester in the form of an average.
Activity during classes is rewarded with additional points, in particular for:
• discussion of additional aspects of the issue,
• effectiveness of applying the acquired knowledge when solving a given problem,
• comments leading to the improvement of teaching materials or the teaching process.

Programme content

The lecture program covers the following topics:
1. Definition and functions of the operating system, classification of operating systems, structure system software and its relationship with hardware, the principle of operation of the system kernel operational.
2. General concept of computer system resource management.
3. CPU management: CPU time allocation planning, ranking criteria; algorithms CPU allocation planning.
4. Working memory management: evolution of memory organization, memory allocation, image creation process in memory, paging and segmentation.
5. Virtual memory: page missing error, operational efficiency, page replacement problem, algorithms page exchanges.
6. Management of input/output devices: classification of input/output devices, structure

I/O mechanism, interaction of the central unit with input/output devices, buffering and spooling.
7. File system - logical organization: file definition and its attributes, file access methods, file operations interface, logical directory structure.
8. File system - physical organization: allocation of disk blocks (contiguous, chain and index), free space management (bit vector, linked list, grouping, counting), implementation directory (linear list, hash table, index structure); implementation of file operations (buffer cache, integrity problem, concurrent file access).
9. File system - implementation examples: CP/M, DOS, ISO 9660, Unix, NTFS.
Laboratory classes are conducted in the form of fifteen two-hour classes in the laboratory computer. The exercises are carried out individually by students. The topic concerns using a Unix-like operating system and covers the following topics:
1. Basic concepts: logging in, command interpreter, terminal as a device, users, groups, hierarchical directory structure, use of help.
2. File system support: basic operations on files and directories (copying, relocating, linking), generalization patterns.
3. Access rights: interpretation of access rights to regular files and directories, ways to change rights access.
4. Process handling: creating a list of processes, annihilation, changing priority, methods starting processes (sequential, concurrent, background, conditional, with stream redirection, streams).
5. Filters: head, tail, more, grep, cut, tr, sort, uniq, sed, etc.
6. Utility programs: cat, cmp, comm, wc, vi editor, etc.
7. Shell: local and environment variables, aliases
8. Scripts: conditional structures, loops, selection statement, support for special variables (including positional parameters).

Course topics

The lecture program covers the following topics:
1. Definition and functions of the operating system, classification of operating systems, structure system software and its relationship with hardware, the principle of operation of the system kernel operational.
2. General concept of computer system resource management.
3. CPU management: CPU time allocation planning, ranking criteria; algorithms CPU allocation planning.
4. Working memory management: evolution of memory organization, memory allocation, image creation process in memory, paging and segmentation.
5. Virtual memory: page missing error, operational efficiency, page replacement problem, algorithms page exchanges.
6. Management of input/output devices: classification of input/output devices, structure I/O mechanism, interaction of the central unit with input/output devices, buffering and spooling.
7. File system - logical organization: file definition and its attributes, file access methods, file operations interface, logical directory structure.
8. File system - physical organization: allocation of disk blocks (contiguous, chain and index), free space management (bit vector, linked list, grouping, counting), implementation directory (linear list, hash table, index structure); implementation of file operations (buffer cache, integrity problem, concurrent file access).
9. File system - implementation examples: CP/M, DOS, ISO 9660, Unix, NTFS.
Laboratory classes are conducted in the form of fifteen two-hour classes in the laboratory computer. The exercises are carried out individually by students. The topic concerns using a Unix-like operating system and covers the following topics:
1. Basic concepts: logging in, command interpreter, terminal as a device, users, groups, hierarchical directory structure, use of help.
2. File system support: basic operations on files and directories (copying, relocating, linking), generalization patterns.
3. Access rights: interpretation of access rights to regular files and directories, ways to change rights access.
4. Process handling: creating a list of processes, annihilation, changing priority, methods

starting processes (sequential, concurrent, background, conditional, with stream redirection, streams).
5. Filters: head, tail, more, grep, cut, tr, sort, uniq, sed, etc.
6. Utility programs: cat, cmp, comm, wc, vi editor, etc.
7. Shell: local and environment variables, aliases
8. Scripts: conditional structures, loops, selection statement, support for special variables (including positional parameters).

Teaching methods

1. Lecture: multimedia presentation, presentation illustrated with examples given on the board, solving tasks.
2. Laboratory exercises: performing experiments, solving tasks, discussion, working in team.

Bibliography

Basic
1. A. Silberschatz, J.L. Peterson, G. Gagne, Podstawy systemów operacyjnych, WNT, W-wa, 2006.
2. W. Stallings, Systemy operacyjne. Architektura, funkcjonowanie i projektowanie. wyd. 9, Helion, 2018.
3. A.S. Tanenbaum, H. Bos, Systemy operacyjne. wyd. 4, Helion, 2015.
4. C. Sobaniec, System operacyjny Linux — przewodnik użytkownika, Nakom, Poznań, 2002.
Additional
1. G. Nutt, Operating Systems. A Modern Perspective, Addison Wesley Longman, Inc., 2002.
2. B. Goodheart, J. Cox, Sekrety magicznego ogrodu. UNIX System V Wersja 4 od środka, WNT, W-wa, 2001.
3. U. Vahalia, Jądro systemu UNIX, WNT, W-wa, 2001.
4. M.E. Russinovich, D.A. Solomon, Windows Internals, Microsoft Press, Redmond, Washington, 2005.
5. J. Marczyński, UNIX użytkowanie i administrowanie, wyd. 2, Helion, Gliwice, 2000.
6. A.S. Tanenbaum, Strukturalna organizacja systemów komputerowych. Wydanie V, Helion, Gliwice, 2006.
7. D. Wawrzyniak, Systemy operacyjne i sprzęt informatyczny, W: Informatyka gospodarcza, A. Gąsiorkiewicz, K. Rostek, J. Zawiła-Niedźwiecki (red.), Wydawnictwo C.H. Beck, Warszawa, 2010.

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5,00
Classes requiring direct contact with the teacher	42	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	83	3,00