SYLLABUS

Operating Systems and Computer Architectures

University year 2025-2026

1. Information regarding the programme

1.1. Higher education institution	Babeş Bolyai University
1.2. Faculty	Faculty of Mathematics and Computer Science
1.3. Department	Department of Computer Science
1.4. Field of study	Computer Science
1.5. Study cycle	Master
1.6. Study programme/Qualification	Artificial Intelligence for Connected Industries
1.7. Form of education	Full time

2. Information regarding the discipline

2.1. Name of the discipline	**Operating Systems and Computer Architectures**					Discipline code	**MME8093**
2.2. Course coordinator					Assoc. prof. phd. Darius-Vasile BUFNEA		
2.3. Seminar coordinator					Assoc. prof. phd. Darius-Vasile BUFNEA		
2.4. Year of study	1	2.5. Semester	1	2.6. Type of evaluation	E	2.7. Discipline regime	Mandatory

3. Total estimated time (hours/semester of didactic activities)

3.1. Hours per week	**4**	of which: 3.2 course	2	3.3 seminar/laboratory/project	1/0/1
3.4. Total hours in the curriculum	56	of which: 3.5 course	28	3.6 seminar/laboratory/project	28
Time allotment for individual study (ID) and self-study activities (SA)					**hours**
Learning using manual, course support, bibliography, course notes (SA)					25
Additional documentation (in libraries, on electronic platforms, field documentation)					25
Preparation for seminars/labs, homework, papers, portfolios and essays					25
Tutorship					12
Evaluations					7
Other activities:					0
3.7. Total individual study hours	**94**				
3.8. Total hours per semester	**150**				
3.9. Number of ECTS credits	**6**				

4. Prerequisites (if necessary)

4.1. curriculum	-
4.2. competencies	Average administration and programming skills

5. Conditions (if necessary)

5.1. for the course	Video projector
5.2. for the seminar /lab activities	Computers, Windows and Linux computers, virtual machines, Network infrastructure

6.1. Specific competencies acquired [1]

[1] One can choose either competences or learning outcomes, or both. If only one option is chosen, the row related to the other option will be deleted, and the kept one will be numbered 6.

Professional/ essential competencies	• C4. Align software with system architectures • C11. Design information system
Transversal competencies	• CT1. think analytically • CT4. solve problems

6.2. Learning outcomes

Knowledge	• Analyse software specifications • Interpret technical texts • Define software architecture • Create data models
Skills	• Design information system • Develop with cloud services • Implement cloud resources • Use software libraries • Use software design patterns
Responsibility and autonomy:	• Manage engineering projects • Oversee development of software • Provide technical documentation • Identify customer requirements

7. Objectives of the discipline (outcome of the acquired competencies)

7.1 General objective of the discipline	• Knowing the key concepts related to the architecture of an operating system and operating systems for parallel clusters
7.2 Specific objective of the discipline	• Understanding computer architectural models, processor functioning, and the use of information representation systems in computers. • Knowledge of advanced operational aspects within Windows and Linux operating systems. • Awareness of the interactions between computer architecture, operating systems, and programming languages. • Knowledge of the Unix operating system, Shell programming, Unix file system structure, I/O, processes, signals, and inter-process communication in Unix. • Acquiring skills to build, install, configure, maintain, monitor, and troubleshoot a parallel HPC cluster based on the Linux operating system.

8. Content

8.1 Course	Teaching methods	Remarks
1. Computer System Architecture	Exposure, description, explanation, debate and dialogue, discussion of case studies	
2. Operating System Architecture	Exposure, description, explanation, case studies	

3. File Systems, Distributed File Systems	Exposure, description, explanation, debate and dialogue, discussion of case studies	
4. Static and Dynamic Libraries in Linux (and Windows)	Exposure, description, explanation, case studies	
5. Distributed Operating System Architecture	Exposure, description, explanation, case studies	
6. Unix/Linux: Introduction and Shell Programming	Exposure, description, explanation, debate and dialogue, discussion of case studies	
7-8. Processes. Threads. Concurrency and Parallelism	Exposure, description, explanation, case studies	
9. Memory Management. Distributed Memory	Exposure, description, explanation, debate and dialogue, discussion of case studies	
10. Operating Systems for Parallel and Distributed Architectures	Exposure, description, explanation, case studies	
11. Architecture of Parallel Clusters	Exposure, description, explanation, case studies	
12-13. Basics of Network, Operating System, and Distributed Operating System Administration	Exposure, description, explanation, case studies	
14. Final review	Exposure, description, explanation, case studies	

Bibliography

1. Albing C., Vossen J.P., Newham C. bash Cookbook. O'Reilly, 2007
2. Boian F., Vancea A., Boian R., Bufnea D., Sterca A., Cobarzan C., Cojocar D., Sisteme de operare, Ed. Risoprint, 2006.
3. Stallings W., Operating Systems: Internal and Design Principles. 6th edition, Prentice Hall,
4. Tanenbaum A.S., Modern Operating Systems. 3rd edition, Prentice Hall, 2009
5. Al. Vancea, F. Boian, D. Bufnea, A. Gog, A. Darabant, A. Sabau – Arhitectura calculatoarelor. Limbajul de asamblare 80x86., Editura Risoprint, Cluj-Napoca, 2005.
6. A. Gog, A. Sabau, D. Bufnea, A. Sterca, A. Darabant, Al. Vancea – Programarea în limbaj de asamblare 80x86. Exemple şi aplicaţii., Editura Risoprint, Cluj-Napoca, 2005.
7. Randal Hyde – The Art of Assembly Programming, No Starch Press, 2003. (http://homepage.mac.com/randyhyde/webster.cs.ucr.edu/www.artofasm.com/DOS/index.html)
8. George F. Coulouris, Jean Dollimore, Tim Kindberg: Distributed Systems: Concepts and Design, Addison-Wesley; 5 edition (May 7, 2011), ISBN-10: 0132143011, ISBN-13: 978-0132143011
9. Joseph D. Sloan: High Performance Linux Clusters with OSCAR, Rocks, OpenMosix, and MPI, O'Reilly Media (November 23, 2004), ISBN-10: 0596005709, ISBN-13: 978-0596005702
10. Charles Bookman: Linux Clustering: Building and Maintaining Linux Clusters, Sams Publishing; 1 edition (June 29, 2002), ISBN-10: 1578702747, ISBN-13: 978-1578702749

8.2 Seminar / laboratory	Teaching methods	Remarks
1. Project presentation	Conversation, debate, case studies	The Seminar/lab is organized as a total of 7 classes - 2 hours every other week
2. Cluster requirements	Conversation, debate, case studies	
3. Cluster building and deployment	Conversation, debate, case studies	
4. Cluster configuration	Conversation, debate, case studies	
5. Cluster maintenance	Conversation, debate, case studies	
6. Cluster debugging and monitoring	Conversation, debate, case studies	
7. Final evaluation of seminar/lab activities	Conversation, debate	

Bibliography

Students will have to build, deploy, configure, maintain, monitor and debug a Linux parallel cluster that runs in a virtual environment using virtual machines running on a physical Windows operating system. The key concepts to accomplish these goals are presented during the course hours and are also available in the course' bibliography (see above).

9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

- Courses with similar content are taught to students at other major universities, such as the partner universities within the European-funded program under which the master's program in Artificial Intelligence for Connected Industries is implemented.
- The course content is considered highly important in the current context of the need for specialists in operating systems, parallel operating systems, clusters, DevOps, and the increasing demand for computing power for computational science, artificial intelligence, interdisciplinary applications, or commercial applications.

10. Evaluation

Activity type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course	Know the key concepts related to computer architectures, operating systems, distributed operating systems and parallel cluster architectures	Written exam	30%
10.5 Seminar/laboratory	Know how to deploy, maintain, debug and monitor a parallel cluster in a virtual environment	Presentation on an operating systems or computer architecture related topic	30%
		Homework assignments	30%
		Default	10%
10.6 Minimum standard of performance			
At least grade 5 (from a scale of 1 to 10) at written exam and seminar/lab activities.			

11. Labels ODD (Sustainable Development Goals)[2]

Not applicable.

Date:
14.04.2025

Signature of course coordinator

Assoc. prof. phd. Darius-Vasile BUFNEA

Signature of seminar coordinator

Assoc. prof. phd. Darius-Vasile BUFNEA

Date of approval:
...

Signature of the head of department

Assoc. prof. phd. Adrian STERCA

[2] Keep only the labels that, according to the *Procedure for applying ODD labels in the academic process*, suit the discipline and delete the others, including the general one for *Sustainable Development* – if not applicable. If no label describes the discipline, delete them all and write „*Not applicable*.".