SYLLABUS

Computer Networks

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	Bachelor
1.6. Study programme/Qualification	Computer Science
1.7. Form of education	Full-Time

2. Information regarding the discipline

2.1. Name of the discipline			Computer Networks				Discipline code	**MLE5022**
2.2. Course coordinator			PhD. Prof. Adrian Sergiu DARABANT					
2.3. Seminar coordinator			PhD. Prof. Adrian Sergiu DARABANT					
2.4. Year of study	**2**	2.5. Semester	3	2.6. Type of evaluation	**E**	2.7. Discipline regime		**Compulsory**

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

3.1. Hours per week		**2**	of which: 3.2 course	**2**	3.3 seminar/laboratory/ project	**2**
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)						20
Additional documentation (in libraries, on electronic platforms, field documentation)						20
Preparation for seminars/labs, homework, papers, portfolios and essays						30
Tutorship						11
Evaluations						13
Other activities:						-
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	• Operating Systems, Computer System Architecture
4.2. competencies	• Basic knowledge on C/C++ and Python programming, basic network knowledge, data encryption algorithms, basic skills on using virtualization technologies

5. Conditions (if necessary)

5.1. for the course	• Classroom with network and Internet access and to laboratory equipment.
5.2. for the seminar /lab activities	• Laboratory with Internet connected computers; Linux and Windows;

6.1. Specific competencies acquired [1]

Professional/ essential competencies	• design and administration of computer networks • advanced programming skills in high-level programming languages
Transversal competencies	• application of organized and efficient work rules, of responsible attitudes towards the didactic-scientific field, to bring creative value to own potential, with respect for professional ethics principles and norms • use of efficient methods and techniques to learn, inform, research and develop the abilities to bring value to knowledge, to adapt at the requirements of a dynamical society and to communicate efficiently in Romanian language and in an international language

6.2. Learning outcomes

Knowledge	• The graduate has adequate knowledge of the protocols by which the Internet works and has the necessary skills to design and test his/her own protocols. • The graduate has the basic knowledge required to install, configure and maintain a server system on the Internet. • The graduate possesses the basic knowledge of operating system specific programming and is familiar with scripting languages.
Skills	• The graduate is able to design and maintain a computer network of medium complexity. • The graduate is able to present and explain methods, algorithms, paradigms and techniques used in various branches of computer science.
Responsibility and autonomy:	• The graduate has the ability to observe and obtain information from various sources. • The graduate has the ability to understand and communicate information effectively.

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

7.1 General objective of the discipline	• Be able to understand the fundamental principles and inner workings of a computer network and of Internet
7.2 Specific objective of the discipline	• Learning the underlying concepts and principles of modern computer networks with emphasis on protocols, architectures, and implementation issues; • Learning to program networking applications using TCP/IP • Learning and understand the layered Internet protocols architecture • Have all the basis knowledge about TCP/IP – theoretical aspects and programming communicating applications

8. Content

8.1 Course	Teaching methods	Remarks
1. Computer Networks Introduction. Definition. Examples. Network Topologies.	Exposure, conversation, explanation, didactical demonstration	
2. The socket programming API. Network programming using TCP and UDP.	Exposure, conversation, explanation, didactical demonstration	
3. Protocols: definition. Protocol layers. The OSI reference model. The TCP/IP layered model.	Exposure, conversation, explanation, didactical	

---

[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.

		demonstration	
4.	The functions and services of the IP layer. Structure of an IP datagram. IP addressing (classfull). Datagram check summing. The ARP protocol.	Exposure, conversation, explanation, didactical demonstration	
5.	The concept of Subnetworks and Supernetworks. CIDR. Network masks.	Exposure, conversation, explanation, didactical demonstration	
6.	The UDP protocol and services. The structure of an UDP datagram UDP ports and processes.	Exposure, conversation, explanation, didactical demonstration	
7.	The TCP protocol. Structure of a TCP segment. Principles of TCP data transmission.	Exposure, conversation, explanation, didactical demonstration	
8.	The TCP Sliding Window mechanism. Flow Control. Congestion avoidance.	Exposure, conversation, explanation, didactical demonstration	
9.	Broadcast and multicast communication. The ICMP protocol. Error and network state signaling.	Exposure, conversation, explanation, didactical demonstration	
10.	The application layer. HTTP, SMTP, FTP	Exposure, conversation, explanation, didactical demonstration	
11.	The Internet Domain Name System. The DNS protocol.	Exposure, conversation, explanation, didactical demonstration	
12.	Network routing. Distance based and link state based routing algorithms. Routing protocols: RIP, BGP, OSPF.	Exposure, conversation, explanation, didactical demonstration	
13.	The physical layer. Transmission media. Characteristics, fiber networks, wireless networks. Error detection and correction.	Exposure, conversation, explanation, didactical demonstration	
14.	Network Security; Netiquette and computer network behavior norms.	Exposure, conversation, explanation, didactical demonstration	

Bibliography

1. J. Kurose, K. Ross, Computer Networking: A Top Down Approach, Addison-Wesley, rev2,3,4 2002-2007.
2. Douglas E. Comer, Internetworking with TCP/IP
     a. Vol 1- Principles, Protocols, and Architecture
     b. Vol 3- Client-Server Programming and Applications
3. G.R.Wright, R. Stevens, TCP/IP Illustrated – vol 1,2, Addison Wesley.
4. Matt Naugle, Illustrated TCP/IP – A Graphic Guide to protocol suite, John Willey & Sons, 1999.
5. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, UNIX® Network Programming Volume 1, Third Edition: The Sockets Networking API
6. Peterson, Larry - Davie, Bruce: Computer Networks: A Systems Approach. Morgan Kaufman, (3rd ed.), 2003.
7. Stallings, William: Data and Computer Communications. Prentice Hall, (6th ed.), 2000.
8. Tanenbaum, Andrew S.: Computer Networks. Prentice Hall, (4th ed.), 2003.
9. Dr. Nasrine Abushakra: Netiquette: Modern Manners For A Modern World: The Ultimate Guide To Online Etiquette, ISBN 1523817569, 2016

8.2 Seminar / laboratory	Teaching methods	Remarks
1. Laboratory Configuration. Necessary tools, Virtual machines and build systems;	Explanation, dialogue, case studies, examples, proofs	
2. A simple client-server TCP application;	Explanation, dialogue, case studies, examples, proofs	
3. Concurrent TCP client-server applications;	Explanation, dialogue, case studies, examples, proofs	
4. Concurrent Multiplexed TCP- Servers. The select call. Network debugging – wireshark	Explanation, dialogue, case studies, examples, proofs	
5. Simple UDP client-server; Security Handling;	Explanation, dialogue, case studies, examples, proofs	
6. Complex/Concurrent UDP applications. Ping. Traceroute.	Explanation, dialogue, case studies,	

Ipconfig/ifconfig.	examples, proofs	
7. TCP/IP programming -Mid term evaluation;	Practical tests	
8. Packet Tracer - simple network simulation	Explanation, dialogue, case studies, examples, proofs	
9. Packet Tracer - Physical/logical network design	Explanation, dialogue, case studies, examples, proofs	
10. Packet Tracer - Static Routing, NAT	Explanation, dialogue, case studies, examples, proofs	
11. Packet Tracer – RIP Routing	Explanation, dialogue, case studies, examples, proofs	
12. Packet Tracer – Packet filtering and VLANs	Explanation, dialogue, case studies, examples, proofs	
13. Packet Tracer – Complex design	Explanation, dialogue, case studies, examples, proofs	
14. Lab Evaluation.	Practical tests.	

Bibliography:
1. Douglas E. Comer, Internetworking with TCP/IP - Vol 3- Client-Server Programming and Applications
2. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, UNIX® Network Programming Volume 1, Third Edition: The Sockets Networking API
3. Cisco Networking Academy Classes, http://cisco.netacad.net

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

- The course respects the IEEE and ACM Curriculla Recommendations for Computer Science studies;
- The course exists in the studying program of all major universities in Romania and abroad;
- The content of the course covers the most important aspects necessary for a network engineer/architect in a network specialized company.

10. Evaluation

Activity type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course	- know the basic principle of computer networks theory; - apply the course concepts - problem solving	Written exam-Moodle Test	50%
10.5 Seminar/laboratory	- TCP/IP programming skills and network simulation knowledge	Mid-term and final term lab tests or overall lab grade (whichever apply)	50%
10.6 Minimum standard of performance			
At least a grading of 5 (from a scale of 1 to 10) at both theoretical exam and laboratory assessments.			

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

Not applicable.

Date:
15/04/2025

Signature of course coordinator
PhD Prof. Adrian Sergiu DARABANT

Signature of seminar coordinator
PhD Prof. Adrian Sergiu DARABANT

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.*".