SYLLABUS

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**

2. Information regarding the discipline

2.1 Name of the discipline	**Graph algorithms**						
2.2 Course coordinator	**Lect. PhD. Radu Lupsa**						
2.3 Seminar coordinator	**Lect. PhD. Radu Lupsa**						
2.4. Year of study	**1**	2.5 Semester	**2**	2.6. Type of evaluation	**C**	2.7 Type of discipline	**Compulsory**

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	1 sem + 1 lab
3.4 Total hours in the curriculum	56	Of which: 3.5 course	28	3.6 seminar/laboratory	28
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					20
Additional documentation (in libraries, on electronic platforms, field documentation)					10
Preparation for seminars/labs, homework, papers, portfolios and essays					30
Tutorship					4
Evaluations					5
Other activities:					-

3.7 Total individual study hours	69
3.8 Total hours per semester	125
3.9 Number of	5

ECTS credits	

4. Prerequisites (if necessary)

4.1. curriculum	• Data Structures and Algorithms
4.2. competencies	Average skils in analysis and design of algorithms and data structures, including implementing them in a programming language.
5.1. for the course	•
5.2. for the seminar /lab activities	Laboratory with computers; high level programming language environment (C++, Java, .NET, python)

5. Conditions (if necessary)

6. Specific competencies acquired

Professional competencies	• C3.2 Identificarea si explicarea modelelor informatice de baza adecvate domeniului de aplicare • C3.3 Utilizarea modelelor si instrumentelor informatice si matematice pentru rezolvarea problemelor specifice domeniului de aplicare • C 4.2 Interpretarea de modele matematice si informatice (formale) • C 4.3 Identificarea modelelor si metodelor adecvate pentru rezolvarea unor probleme reale
Transversal competencies	• CT1 Aplicarea regulilor de munca organizata si eficienta, a unor atitudini responsabile fata de domeniul didactic-stiintific, pentru valorificarea creativa a propriului potential, cu respectarea principiilor si a normelor de etica profesionala • CT3 Utilizarea unor metode si tehnici eficiente de învatare, informare, cercetare si dezvoltare a capacitatilor de valorificare a cunostintelor, de adaptare la cerintele unei societati dinamice ?i de comunicare în limba româna ?i într-o limba de circula?ie interna? ionala

7.1 General objective of the discipline	• Knowing the graph theoretical concepts and using these concepts in the problem modeling. • Knowing how to implement the graph algorithms in a programming language.
7.2 Specific objective	• Analyzing the issues around the main topics of graph: connectivity, shortest paths, modeling prerequisites and activity planning, flows, traveling salesman problem, planar graphs.

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

To be able to use graphs as a modeling tool for programming problems. To know the elementary graph problems and their solutions. To develop the general algorithmic skills.

8. Content

8.1 Course	Teaching methods	Remarks
1. Basic graph theory definitions (graph, multigraph, directed graph/multigraph, walk, trail, path), basics on graph representations	Exposure: description, explanation, examples, debate	
2. In-deep study of possible graph representations	Exposure: description, explanation, examples, debate	
3. Connectivity and shortest path problems. Depth-first and breadth-first traversal of a graph.Connected and strongly-connected components.	Exposure: description, explanation, examples, debate	
4. Minimum-cost path in a graph. Dynamic programming approach. Bellman-Ford algorithm.	Exposure: description, explanation, examples, debate	
5. Dijkstra algorithm . A-star algorithm. Floyd-Warshal algorithm.	Exposure: description, explanation, examples, debate	
6. Dependency graphs, partial order and topological sorting. Topological sorting algorithms and strongly connected components algorithms.	Exposure: description, explanation, examples, debate	
7. Activity planning problem and algorithms.	Exposure: description, explanation, examples, debate	
8. Trees and forests. Minimum spanning trees. Kruskal and Prim algorithms.	Exposure: description, explanation, examples, debate	
9. NP-complete problems. Hamiltonian cycle, Traveling Salesman Problem.	Exposure: description, explanation, examples, debate	
10. Other hard problems: clique, vertex cover, coloring.	Exposure: description, explanation, examples, debate	
11. Eulerian cycle. Planar graphs: Euler's relation, K5 and K3,3 graphs, relations between number of edges and vertices.	Exposure: description, explanation, examples, debate	
12. Transport networks. Maximum flow. Ford-Fulkerson algorithm.	Exposure: description, explanation, examples, debate	
13. Maximum flow of minimim cost.	Exposure: description, explanation, examples, debate	
14. Matching problem	Exposure: description, explanation, examples, debate	

Bibliography

1. CORMEN, LEISERSON, RIVEST: Introducere in algoritmi, Editura Computer Libris Agora, 2000.
2. T. TOADERE: Grafe. Teorie, algoritmi si aplicatii , Ed. Albastra, Cluj-N., 2002

3. KÁSA ZOLTÁN: Combinatiroca cu aplicatii, Presa Universitara Clujeana, 2003.

4. BERGE C., Graphes et hypergraphes, Dunod, Paris 1970.

5. BERGE C., Teoria grafurilor si aplicatiile ei, Ed. Tehnica, 1972

6. http://www.cs.ubbcluj.ro/~rlupsa/edu/grafe/

8.2 Seminar	Teaching methods	Remarks
1. Basic definitions. Graph representations.	Dialogue, debate, examples, guided discovery	
2. More on graph representations. Graph traversals. Connected components.	Dialogue, debate, examples, guided discovery	
3. Finding shortest path. Bellman-Ford algorithm.	Dialogue, debate, examples, guided discovery	
4. Shortest path: Dijkstra and Floyd-Warshall.	Dialogue, debate, examples, guided discovery	
5. Tree traversal. Minimum spanning tree: Kruskal and Prim algorithms.	Dialogue, debate, examples, guided discovery	
6. Planar graphs. Hard problems in graph theory.	Dialogue, debate, examples, guided discovery	
7. Maximim flow and matching problems.	Dialogue, debate, examples, guided discovery	

8.3 Laboratory	Teaching methods	Remarks
8. Graph representations	Dialogue, debate, case study, guided discovery	
9. More on graph representations	Dialogue, debate, case study, guided discovery	
10. Connectivity and lowest length paths	Dialogue, debate, case study, guided discovery	
11. Lowest cost paths	Dialogue, debate, case study, guided discovery	
12. Dependency graphs. Minimum spanning trees.	Dialogue, debate, case study, guided discovery	
13. NP-complete problems	Dialogue, debate, case study, guided discovery	
14. Finishing the lab activity.	Dialogue, debate, case study, guided discovery	

Bibliography

1. KÁSA Z., TARTIA C., TAMBULEA L.: Culegere de probleme de teoria grafelor, Lito. Univ. Cluj-Napoca 1979.

2. CATARANCIUC S., IACOB M.E., TOADERE T., Probleme de teoria grafelor, Lito. Univ. Cluj-Napoca, 1994.

3. TOMESCU I., Probleme de combinatorica si teoria grafurilor. Ed. Did. si Pedag. Bucuresti 1981.

4. KÁSA Z., TARTIA C., TAMBULEA L.: Culegere de probleme de teoria grafelor, Lito. Univ. Cluj-Napoca 1979.

5. CATARANCIUC S., IACOB M.E., TOADERE T., Probleme de teoria grafelor, Lito. Univ. Cluj-Napoca, 1994.

6. TOMESCU I., Probleme de combinatorica si teoria grafurilor. Ed. Did. si Pedag. Bucuresti 1981.

7. http://www.cs.ubbcluj.ro/~rlupsa/edu/grafe/

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;

10. Evaluation

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	- know the basic principle of the domain; - apply the course concepts - problem solving	Written exam	66.67%
10.5 Seminar/lab activities	- be able to implement course concepts and algorithms	Verifying the practical works.	33.33%
10.6 Minimum performance standards			
• At least grade 5 (from a scale of 1 to 10) at both written exam and laboratory work.			

Date Signature of course coordinator Signature of seminar coordinator

.... Lect. PhD. Radu Lupsa............. ..Lect. PhD. Radu Lupsa

Date of approval

....

Signature of the head of department

...........….............................