**SYLLABUS**

**Academic year 2023-2024**

**Year of study I / Semester II**

## 1. Information on academic programme

| | |
|---|---|
| 1.1. University | **„1 Decembrie 1918" from Alba Iulia** |
| 1.2. Faculty | **Faculty of Computer Sience and Engineering** |
| 1.3. Department | **Informatics, Mathematics and Electronics Department** |
| 1.4. Field of Study | **Computer Science** |
| 1.5. Cycle of Study | **Undergraduate** |
| 1.6. Academic programme / Qualification | **Computer Science / 251201, 251203, 251204** |

## 2. Information of Course Matter

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2.1. Course | ***Graph algorithms*** | | | 2.2. Code | | **INFO111** | |
| 2.3. Course Leader | Dr. Dorin Wainberg | | | | | | |
| 2.4. Seminar Tutor | Dr. Dorin Wainberg | | | | | | |
| 2.5. Academic Year | **I** | 2.6. Semester | **II** | 2.7. Type of Evaluation (E – final exam/ CE - colloquy examination / CA - continuous assessment) | **E** | 2.8. Type of course (C– Compulsory, Op – optional, F - Facultative) | **C** |

## 3. Course Structure (Weekly number of hours)

| | | | | | |
|---|---|---|---|---|---|
| 3.1. Weekly number of hours | **4** | 3.2. course | **2** | 3.3. seminar, laboratory | **2** |
| 3.4. Total number of hours in the curriculum | **56** | 3.5. course | **28** | 3.6. seminar, laboratory | **28** |

| Allocation of time: | hours |
|---|---|
| Individual study of readers | **30** |
| Documentation (library) | **20** |
| Home assignments, Essays, Portfolios | **42** |
| Tutorials | **-** |
| Assessment (examinations) | **2** |
| Other activities……. | **-** |

| | |
|---|---|
| 3.7 Total number of hours for individual study | 94 |
| 3.9 Total number of hours per semester | 94+56=150 |
| 3.10 Number of ECTS | 6 |

## 4. Prerequisites (where applicable)

| | |
|---|---|
| 4.1. curriculum-based | Linear algebra |
| 4.2. competence-based | C4.1 Defining the basic concepts and principles of the professional field, as well as mathematical theories and models.<br>C4.2 Interpretation of mathematical and computer model. |

## 5. Requisites (where applicable)

| | |
|---|---|
| 5.1. course-related | Room equipped with video projector / board |
| 5.2. seminar/laboratory-based | Room equipped with board |

## 6. Specific competences to be acquired

| | |
|---|---|
| Professional competences | *C1.1 The appropriate description of programming paradigms and of specific language mechanisms, as well as the identification of differences between semantic and syntactic aspects.*<br>*C2.1 The identification of appropriate methodologies for software systems development.*<br>*C2.2 The identification and explanation of appropriate mechanisms for software systems specification.*<br>*C2.3 The use of methodologies, specification mechanisms and development environments for the development of computer applications.*<br>*C6.1. The identification of base concepts and models for computer systems and computer networks.*<br>*C6.2. The identification and explanation of base architectures for organizing and managing systems and networks.*<br>*C6.3. The use of various techniques for installing, configuring and managing systems and networks.* |
| Transversal competences | - |

**7. Course objectives (**as per the programme specific competences grid)

| | |
|---|---|
| 7.1 General objectives of the course | Learning the fundamental concepts in graph theory, with a sense of some of its modern applications. |
| 7.2 Specific objectives of the course | Our aims in this course are twofold. First, to discuss some of the major results of graph theory, and to introduce the language, methods and terminology of the subject. Second, to emphasize various approaches (algorithmic, probabilistic, etc) that have proved fruitful in modern graph theory: these modes of thinking about the subject have also proved successful in areas of informatics, and we hope that students will find the techniques learnt in this course to be useful in their future works. |

**8. Course contents**

| **8.1 Course (learning units)** | **Teaching methods** | **Remarks** |
|---|---|---|
| Preliminaries.<br>General notions.<br>Ways for representing a graph | ***Lecture, conversation*** | |
| Basic concepts in Graph Theory<br>Cyclomatic number | ***Lecture, conversation*** | |
| Graph traversal<br>Breadth First Traversal<br>Depth First Traversal | ***Lecture, conversation*** | |
| Minimum distances in graphs | ***Lecture, conversation*** | |
| Connected components | ***Lecture, conversation*** | |
| Bipartite graphs<br>Maximum matching problem in a bipartite graph | ***Lecture, conversation*** | |
| Hamiltonian paths and circuits<br>Chen algorithm<br>Foulkes algorithm<br>Kaufmann algorithm | ***Lecture, conversation*** | |
| Flow networks<br>Bellman-Kalaba algorithm<br>Ford algorithm<br>Dijkstra algorithm | ***Lecture, conversation*** | |
| Maximum flow in transport networks | ***Lecture, conversation*** | |
| Trees. Deffinitions and theorems. | ***Lecture, conversation*** | |
| Traversal of a dirrected tree | ***Lecture, conversation*** | |
| Trees of minimum values<br>Kruskal algorithm<br>Sollin algorithm | ***Lecture, conversation*** | |
| Binary trees | ***Lecture, conversation*** | |
| Structural trees | ***Lecture, conversation*** | |

**References**


1. Behzad, M., Chartrand, G., Lesniak-Foster, L., *Graphs and digraphs*, Prindle, Weber and Schmidt, Boston, Massachusetts, 2014.
2. Bollobas, B., *Graph theory. An introductory course*, Springer-Verlag, New York, Heidelberg, Berlin, 2012.
3. Christo¯des, N., *Graph theory. An algorithmic approach*, Academic Press, 2011.
4. Ford, L., Fulkerson, D. R., *Flows in networks*, Princeton Univ. Press, 1992.
5. Wainberg, D., Breaz, D., Alb Lupaş, A., *Elemente de Algoritmica grafurilor*, Ed. Aeternitas, 2010.


| **Seminars-laboratories** | **Teaching methods** | |
|---|---|---|
| Preliminaries.<br>General notions.<br>Ways for representing a graph | ***Exercises and problems*** | |
| Basic concepts in Graph Theory<br>Cyclomatic number | ***Exercises and problems*** | |
| Graph traversal<br>Breadth First Traversal<br>Depth First Traversal | ***Exercises and problems*** | |
| Minimum distances in graphs | ***Exercises and problems*** | |
| Connected components | ***Exercises and problems*** | |

| | | |
|---|---|---|
| Bipartite graphs<br>Maximum matching problem in a bipartite graph | ***Exercises and problems*** | |
| Hamiltonian paths and circuits<br>Chen algorithm<br>Foulkes algorithm<br>Kaufmann algorithm | ***Exercises and problems*** | |
| Flow networks<br>Bellman-Kalaba algorithm<br>Ford algorithm<br>Dijkstra algorithm | ***Exercises and problems*** | |
| Maximum flow in transport networks | ***Exercises and problems*** | |
| Trees. Deffinitions and theorems. | ***Exercises and problems*** | |
| Traversal of a dirrected tree | ***Exercises and problems*** | |
| Trees of minimum values<br>Kruskal algorithm<br>Sollin algorithm | ***Exercises and problems*** | |
| Binary trees | ***Exercises and problems*** | |
| Structural trees | ***Exercises and problems*** | |

**References**


1. Behzad, M., Chartrand, G., Lesniak-Foster, L., *Graphs and digraphs*, Prindle, Weber and Schmidt, Boston, Massachusetts, 2014.
2. Bollobas, B., *Graph theory. An introductory course*, Springer-Verlag, New York, Heidelberg, Berlin, 2012.
3. Christo¯des, N., *Graph theory. An algorithmic approach*, Academic Press, 2011.
4. Ford, L., Fulkerson, D. R., *Flows in networks*, Princeton Univ. Press, 1992.
5. Wainberg, D., Breaz, D., Alb Lupaş, A., *Elemente de Algoritmica grafurilor*, Ed. Aeternitas, 2010.


**9. Corroboration of course contents with the expectations of the epistemic community's significant representatives, professional associations and employers in the field**

*Applying the discipline Graph algorithms in building and developing of a computer network is essential. Any company or institution that owns a computer network would need graduates who have successfully completed this subject. Also, a lot of programming techniques are based on the algorithms presented here. Therefore, we can conclude that Graph algorithms is a fundamental course of computer science.*

**10. Assessment**

| Activity | 10.1 Evaluation criteria | 10.2 Evaluation methods | 10.3 Percentage of final grade |
|---|---|---|---|
| 10.4 Course | *Final evaluation* | *Written paper* | 50% |
| 10.5 Seminar/laboratory | *Continuous assessment* | *Tests during the semester* | 50% |
| 10.6 Minimum performance standard: Modelling and solving some medium complexity level problems, using the mathematical and computer sciences knowledge. | | | |

Submission date | Course leader signature | Seminar tutor signature

Date of approval by Department members | Department director signature

Date of approval by Faculty councile | Dean signature