COURSE OUTLINE

(1) GENERAL

SCHOOL	ENGINEERING		
ACADEMIC UNIT	Department of Computer Engineering and Informatics		
LEVEL OF STUDIES	Undergraduate		
COURSE CODE	**CEID_NE5057**	**SEMESTER**	**fall**
COURSE TITLE	Algorithms and Combinatorial Optimization		

INDEPENDENT TEACHING ACTIVITIES *if credits are awarded for separate components of the course, e.g. lectures, laboratory exercises, etc. If the credits are awarded for the whole of the course, give the weekly teaching hours and the total credits*	**WEEKLY TEACHING HOURS**	**CREDITS**
Lectures, Tutorials, Laboratory	2(L), 2(T), 1(Lab)	5
Add rows if necessary. The organisation of teaching and the teaching methods used are described in detail at (d).	TOTAL	5

COURSE TYPE *general background, special background, specialised general knowledge, skills development*	skills development
PREREQUISITE COURSES:	Recommended prerequisite knowledge: "Discrete Mathematics" (NY109), or equivalent.
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	Greek (English if there are Erasmus students)
IS THE COURSE OFFERED TO ERASMUS STUDENTS	Yes
COURSE WEBSITE (URL)	https://www.ceid.upatras.gr/webpages/faculty/zaro/teaching/alg-and-comb-opt/index.html

(2) LEARNING OUTCOMES

Learning outcomes

The course learning outcomes, specific knowledge, skills and competences of an appropriate level, which the students will acquire with the successful completion of the course are described.

Consult Appendix A

- *Description of the level of learning outcomes for each qualifications cycle, according to the Qualifications Framework of the European Higher Education Area*
- *Descriptors for Levels 6, 7 & 8 of the European Qualifications Framework for Lifelong Learning and Appendix B*
- *Guidelines for writing Learning Outcomes*

Upon conclusion of the course the students ought to be able to:

- Understand fundamental concepts of combinatorial and network optimization.
- Apply classic and advanced algorithmic techniques for the solution of fundamental combinatorial and network optimization problems.
- Apply generic techniques (eg linear programming) for the solution of fundamental combinatorial and network optimization problems.

Upon conclusion of the course the students are expected to have the following skills/competences:

- Develop efficient models for the solution of combinatorial and network optimization problems.
- Use classic and advanced combinatorial techniques to develop algorithms for tackling fundamental and complex issues of combinatorial and network optimization problems.
- Use generic methods (eg linear programming) for solving fundamental and complex combinatorial and network optimization problems.

General Competences

Taking into consideration the general competences that the degree-holder must acquire (as these appear in the Diploma Supplement and appear below), at which of the following does the course aim?

Search for, analysis and synthesis of data and information, with the use of the necessary technology *Adapting to new situations* *Decision-making* *Working independently* *Team work* *Working in an international environment* *Working in an interdisciplinary environment* *Production of new research ideas*	*Project planning and management* *Respect for difference and multiculturalism* *Respect for the natural environment* *Showing social, professional and ethical responsibility and sensitivity to gender issues* *Criticism and self-criticism* *Production of free, creative and inductive thinking* *……* *Others…* *…….*

- Search for, analysis and synthesis of data and information, with the use of the necessary technology
- Adapting to new situations
- Decision-making
- Working independently
- Criticism and self-criticism
- Production of free, creative and inductive thinking

(3) SYLLABUS

1. ***Basic Concepts of Combinatorial and Network Optimization***
   Basic models of optimization problems. Representation of optimization problems and their relation to algorithmic efficiency and time complexity.
2. ***Advanced Algorithmic Techniques for Solving Fundamental Optimization Problems***
   Shortest paths: features and properties. Theorems for computing and verifying an optimal solution. Shortest path algorithms (Dijkstra, Bellman-Ford-Moore, Dial, Radix-Heap) and other efficient implementations using priority queues. Methods for detecting and computing negative cycles.
   Maximum flow: features and properties. Theorems for computing and verifying an optimal solution. Maximum flow algorithms: augmenting path, shortest augmenting path, preflow-push.
   Minimum cost flow: features and properties. Theorems for computing and verifying an optimal solution. Algorithms: negative cycle cancelling, successive shortest path.
3. ***Generic Techniques for Solving Optimization Problems***
   Introduction to generic optimization techniques. Global and local optimum. Convex programming. Linear programming. Basic feasible solutions and the Simplex method. Duality. The Ellipsoid method. Interior-point methods. Integer programming.

(4) TEACHING and LEARNING METHODS - EVALUATION

DELIVERY *Face-to-face, Distance learning, etc.*	Face-to-face. Tutorials and laboratory sessions with exemplary solutions of exercises.
USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY *Use of ICT in teaching, laboratory education, communication with students*	ICT methods are used in both teaching and communication with the students. Lecture slides and supplementary material are uploaded in the course's web site.
TEACHING METHODS *The manner and methods of teaching are described in detail.* *Lectures, seminars, laboratory practice, fieldwork, study and analysis of bibliography, tutorials, placements, clinical practice, art workshop, interactive teaching, educational visits, project, essay writing, artistic creativity, etc.* *The student's study hours for each learning activity are given as well as the hours of non-directed study according to the principles of the ECTS*	see table below

Activity	***Semester workload***
Lectures	2*13=26
Tutorials (exercises)	2*13=26
Laboratory practice	1*13=13
Individual study, preparation and problem solving	3*13=39
Weekend study	2*13=26
Study during the 3 “empty weeks” (2 weeks of vacation and 1 week of exam preparation)	5*3=15
Course total (25-30 hours per ECTS unit)	**145**

STUDENT PERFORMANCE EVALUATION	
Description of the evaluation procedure *Language of evaluation, methods of evaluation, summative or conclusive, multiple choice questionnaires, short-answer questions, open-ended questions, problem solving, written work, essay/report, oral examination, public presentation, laboratory work, clinical examination of patient, art interpretation, other* *Specifically-defined evaluation criteria are given, and if and where they are accessible to students.*	The language of instruction and examination is Greek. Special provisions (lecture notes and examinations in English) can be made for foreign students. Evaluation (criteria can be found in the web site of the course): • Exercises (30% of final mark). • Final written examination (70% of final mark). Written final examination: graded difficulty, including short-answer questions, algorithm design for problem solving, proofs of algorithm correctness and complexity, problem modelling, problem solving by applying generic optimization techniques. Series of theoretical exercises aiming at familiarizing students with: • The use of the algorithmic techniques taught in the course. • Solving optimization problems through advanced algorithmic and generic optimization techniques.

(5) ATTACHED BIBLIOGRAPHY

- *Suggested bibliography:*

- J. Kleinberg and E. Tardos, *Algorithm Design*, Pearson Addison-Wesley, 2006.
- T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd Edition, MIT Press, 2009.
- I. Κολέτσος και Γ. Στογιάννης, *Εισαγωγή στην Επιχειρησιακή Έρευνα*. 3η έκδοση. Εκδόσεις Συμεών, 2017. In Greek.
- R. Ahuja, T. Magnanti, and J. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, 1993.
- C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, 1982.
- W. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver, *Combinatorial Optimization*, John Wiley & Sons, 1998.
- Lecture notes and slides uploaded in the web site of the course.

- *Related academic journals:*

- ACM Transactions on Algorithms.
- Algorithmica, Springer.
- European Journal of Operational Research, Elsevier.
- Journal of Discrete Algorithms, Elsevier.
- Journal of the ACM.
- Journal of Optimization Theory and Applications, Springer.
- Operations Research, INFORMS.
- Operational Research, Springer.
- SIAM Journal on Optimization.
- Theoretical Computer Science, Elsevier.