GÖTEBORGS UNIVERSITET

The Board of the IT Faculty

DIT370, Discrete Optimization, 7.5 higher education credits

Second Cycle/A1F

The syllabus in English is the binding document.

1. Confirmation

The Board of IT Faculty/the Dean established the course plan at 2006-11-17. It has been revised 2009-09-18 to be valid from the spring term 2010.

Field of education: Science

Main field: Computer Science

Department: Computer Science and Engineering

2. Position in the educational system

The course is a part of the Computer Science Master´s programme and an elective course at the University of Gothenburg.

The level for the course in relation to degree requirements is Master´s degree, code A1F. The course has course/courses at second cycle level as entry requirements.

3. General prerequisites

The requirement for the course is to have successfully completed a first year studies within the subject Computer Science or equivalent. Specifically the course DIT600 Algorithms (or equivalent) is required.

4. Course content

You learn in this course specific methods to model and solve problems where some objective function shall be maximized or minimized under side constraints, especially for discrete problems, i.e., such with countable objects and integer variables.

5. Learning outcomes

After the course the student is expected be able to:

- identify optimization problems in various fields (industrial production, infrastructure, planning and scheduling, economics, data mining, bioinformatics, computer and network design, etc., and even in artificial intelligence)
- formulate them in exact mathematical models that capture the essentials of the real problems but are still manageable by computational methods
- assess which problem class a given problem belongs to (linear, integer, mixed, nonlinear programming, polynomial or NP-hard, approximable or not), find and how to use more information about problem classes and complexity
- apply linear programming and the theory of network flows to suitable applications, understand these topics both formally and geometrically
- distinguish approximation algorithms from heuristics, apply several heuristic approaches (e.g., branch-and-bound) as well as design techniques for approximation algorithms, to concrete problems
- dualize optimization problems (LP dual, Lagrange dual) and use the dual forms, e.g., to obtain bounds
- apply techniques for the design of exact algorithms (dynamic programming, cutting planes, column generation, parameterized algorithms) to concrete problems
- find more information about available software tools (modelling languages, solvers) and use them

6. Required reading

See separate literature list

7. Assessment

The course is examined by an exam and assignments.

A student who has failed a test twice has the right to change examiner, unless weighty argument can be adduced. A written application should be sent to the Department.

8. Grading scale

The grades are Fail (U), Pass (G) and Pass with Distinction (VG).

9. Course evaluation

The course is evaluated through meetings both during and after the course between teachers and student representatives. Further, an anonymous questionnaire can be used to ensure written information. The outcome of the evaluations serves to improve the course by indicating which parts could be added, improved, changed or removed.

10. Additional information

The course is held in English.