**ISYE 4133 ADVANCED OPTIMIZATION**

**Elective**

**Credit:** 3-0-3

**Prepared by:** Prof. Joel Sokol, Summer 2009

**Prerequisite(s):** ISyE 3133, good programming skill

**Catalog Description:**
Topics include mathematical foundations of optimization, modeling and solution techniques for very large and/or complex problems, algorithmic solution methods and heuristics, and optimization software.

**Text:**

*Introduction to Mathematical Programming: Applications and Algorithms* by Wayne L. Winston, Duxbury Press, 2002 (advanced chapters) or *Optimization in Operations Research* by Ronald L. Rardin, Prentice Hall, 1997 (advanced chapters), or equivalent

User manual for optimization software

Handouts (if/when necessary)

**Objective:**
The purpose of this course is to provide students with

- deeper insight into the fundamentals of optimization, and
- practical techniques for modeling and solving (exactly or approximately) very large and/or complex optimization problems (i.e., problems that can't be solved by giving a straightforward formulation to the solver).

**Topical Outline**

1. Using optimization software within a solution algorithm
2. Solving problems where memory is the limiting factor
    a. Column generation
    b. Constraint generation
    c. Optimization-within-optimization ideas
    d. Methods of data storage and problem creation (sparse representations)
3. Solving problems where complexity or time is the limiting factor
    a. Basic cutting planes
    b. Eliminating symmetry
4. Finding good solutions to intractable problems
    a. Heuristic techniques (construction/improvement, buildup/takeaway, LP-based methods, greedy ideas, decomposition, basic neighborhood search, large-scale neighborhood search, repetition with randomization)
    b. Metaheuristic ideas (genetic algorithms, simulated annealing, tabu search)
5. Linearizing nonlinear problems

6. Understanding computational complexity

**Course Outcomes and their relationships to ISyE Program Outcomes**

At the end of this course, students will be able to:

- Identify whether an optimization problem is hard or easy,
- Better understand the theory of how optimization problems are solved,
- Implement helpful techniques for optimization problems that can't be solved in a straightforward way,
- Handle very large data sets in optimization settings, and
- Understand how to use optimization to solve real-world industrial and systems engineering problems.

| **Course outcome \ Program Outcomes** | **a. apply math** | **b. data** | **c. IE method** | **d. team** | **e. problem solving** | **f. prof/ and ethical responsibilities** | **g. communication** | **h. global, eco, envi and soc context** | **i. continue to improve** | **j. current issues** | **k. participate in an organization** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Identify … hard or easy | H | | M | | L | | | | L | | M |
| Better understand the theory … | H | | H | | | | | | | | |
| Implement helpful techniques … | H P | M P | H P | LP | H P | | LP | | L | | M |
| Handle very large data sets… | M P | H P | M P | LP | M P | | | | L | | L |
| Understand how to use … | | | H P | LP | H P | LP | LP | | L | LP | M |
| | | | | | | | | | | | |

- H, M and L denote high, moderate and low relationships.
- P denotes use in a team project

**Evaluation of the important course outcomes**

Four or more important outcomes will be evaluated from direct questions on the final exam:

1. Students are able to create heuristics tailored to a specific optimization problem.
2. Students are able to identify the proper optimization technique(s) to attempt when problems can't be solved in a straightforward way.
3. Using optimization software, students are able to code solution algorithms for optimization problems that can't be solved in a straightforward way.
4. Students are able to successfully handle large data sets that accompany real-world optimization problems.