

# Computer Science 1490: Introduction to Combinatorial Optimization

**Lecture location:** CIT 368

**Lecture hours:** T/Th 10:30 - 11:50 AM

**Instructor:** Meinolf Sellmann (sello@cs.brown.edu), CIT333

**Head teaching assistant:** Ben Simon (bwsimon@cs.brown.edu)

**Graduate teaching assistant:** Yuri Malitsky (ynm@cs.brown.edu)

**Undergraduate Teaching assistant:** Karthik Pattabiraman (karthikp@cs.brown.edu)

**Texts:** *Combinatorial Optimization, Algorithms and Complexity* by Papadimitriou and Steiglitz and *Linear Programming* by Chvatal

## Course Material

Introduction to Combinatorial Optimization provides a comprehensive overview of basic optimization technology from Operations Research and Constraint Programming. Concepts covered include: Linear Programming, Duality Theory, Total Unimodularity, Backtracking and Branch-and-Bound, Finite Domain Constraint Programming, and Local Search. Students are exposed to the highly industrially relevant standard optimization software packages Cplex and ILOG Solver. Additionally, problems such as SAT, transportation problem, maximum flow, SEND MORE MONEY, and mastermind<sup>1</sup> will be explored.

CSCI1490 fulfills the AI or theory requirement of the Sc.B. and the A.B. degrees. There will be an opportunity for graduate students to earn 2000-level credit for additional course work.

## What You Should Know

- Linear Algebra and how to solve systems of equations
- Modeling (program specification, termination, verification etc.)
- Algorithms and Data Structures (shortest paths, minimum spanning trees, stacks, queues, etc.)
- Basic Complexity Theory (P, NP etc.)

---

<sup>1</sup><http://www.javaonthebrain.com/java/mastermind/>

- C++ (or another OO language with the willingness to write some C++)

## Announcements

The primary source for announcements relating to the course is class, which you are expected to attend. While we strive to post them on the course website as promptly as possible, we do not guarantee that all announcements will be posted.

## Homework

There will be weekly homework assignments which will be a mix of both problem sets and programming assignments. Please strive to make your solutions clear and concise. A set of homework guidelines has been provided, and you are expected to follow them.

Typeset solutions (especially using L<sup>A</sup>T<sub>E</sub>X) are strongly encouraged. Hand-written solutions are acceptable, but if the TAs cannot read a solution, they cannot give it any credit. Homework will generally be assigned every Thursday and due the following Wednesday at 4:00 pm.

**Late homework and programming projects will not be accepted** without explicit prior permission from the instructor. *There are no “late days.”* Extensions may be granted by the instructor, but must be requested prior to 24 hours before the deadline. Homework must be submitted into the correct bin (on the second floor near the TA rooms).

Graded homework will be returned to the hand-back bin in CIT 271.

You must hand in all but two homeworks to pass. However, any homework not handed in will receive a grade of zero. The lowest homework grade will be dropped. This does not apply to the five programming projects however — all of these must be handed in and none of these grades will be dropped.

Corrections to errors in homework will be posted on the course web page <http://www.cs.brown.edu/courses/csci1490/>, which you are expected to check daily for corrections and other announcements. Solutions to homework will also be posted on the course web page.

Please ensure that your homework follows these guidelines. Handins that don't will be penalized accordingly.

Some guidelines for the content of your work:

1. Show all of your work. It is not sufficient to give the answer to a problem — you must show all the work that leads to that answer, including a proof if appropriate.

2. For all questions requiring an algorithm, provide not only pseudocode but written comments explaining your algorithm.
3. Be as concise as possible.

The following are guidelines for the format of your work. **You must follow these guidelines.** They make it much easier for the TAs to grade your work. They may seem strict or ridiculous, but they make our lives much easier.

1. Your name must be at the top of every sheet of paper.
2. Homework problems must be stapled together.
3. Each problem must be on a separate piece of paper. If a problem requires several pages, you may use both sides of the paper, but do not put the solutions for two problems on one sheet of paper.
4. If a problem requires several sheets of paper, please staple those sheets together in addition to stapling that problem to the rest of your homework.

These rules make it easy for the TAs to separate your homeworks into problems, grade them, then reassemble them. Please follow them.

### Collaboration policy

Collaboration, in groups of 2, is permissible and encouraged on all homeworks and projects. You **must** collaborate on at least 4 **homeworks** and/or **projects** during the semester with 4 **different partners**. On joint homeworks, each student is expected to contribute to the final write up, even though only one hand in is required. On all homeworks, students are encouraged to discuss solutions together. Students are not explicitly forbidden from working out problems together. Students may choose which homework assignments or projects to do jointly.

There is no collaboration on exams. The honor code will be strictly enforced. If you need help with a specific homework problem, or any concept in general, please see one of the TAs or the professor.

### Exams

There will be a midterm and a final exam.

## Grading

The final grade will *approximately* be determined as follows:

- 60% homework and projects
- 15% midterm
- 25% final

Please note that your final grade is not strictly computed from these percentages, and will be determined by the staff.

## Tentative course schedule

Week 1 Introduction and Standard Model of Linear Programming

Week 2 Simplex Intuition and Mathematical Foundations

Week 3 Simplex Algorithm and Duality

Week 4 Primal-Dual Algorithm and Elementary Graph Algorithms

Week 5 Transportation Problem

Week 6 Slack, Review, and Midterm

Week 7/8 Flows/Matrix Games, and Total Unimodularity

Week 9/10 Integer Programming

Week 11 Constraint Programming

Week 12 Local Search

Week 13 Slack and Review

Week 14 Review and Final Examination