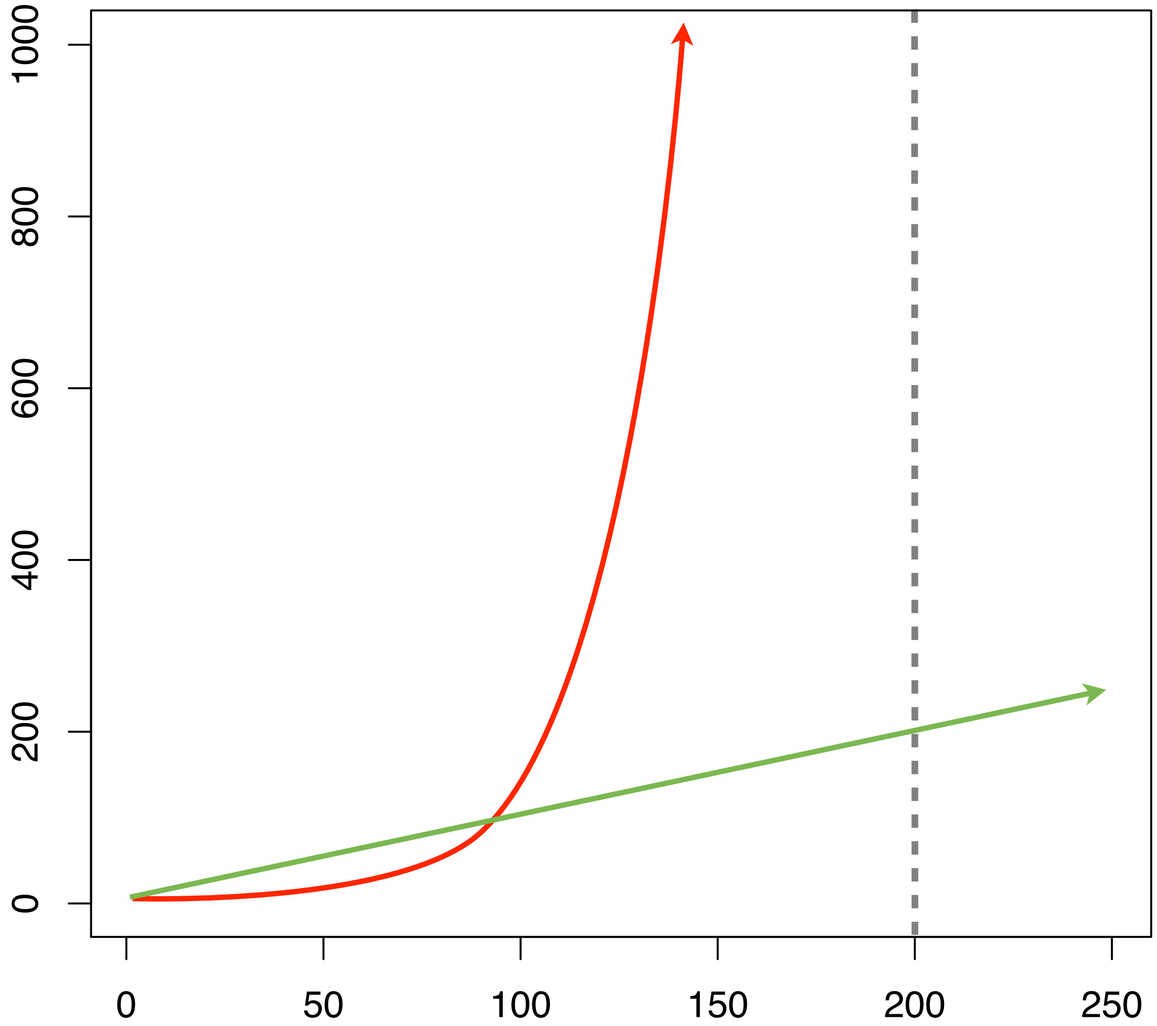
Discrete Optimization

Introduction

Goal of the Lecture

- Course Structure and Philosophy
- Assessment Design

Discrete Optimization is HARD!



Course Structure

- Lectures, introduction to broad topics
  - Constraint Programming
  - Local Search
  - Integer Programming

Assessment Design

- Five NP-Hard problems
- Optimize them
  - by any means!
- Submit some solutions to demonstrate how great your optimization is

Grading Rubric

- Submitting junk or infeasible solutions
  - –0/10
- Submitting solutions of **low** quality
  - –3/10
- Submitting solutions of **good** quality
  - –7/10
- Submitting solutions of **GREAT** quality
  - –10/10
- The best of grade of all submissions
  - –mixing different approaches

Certificates

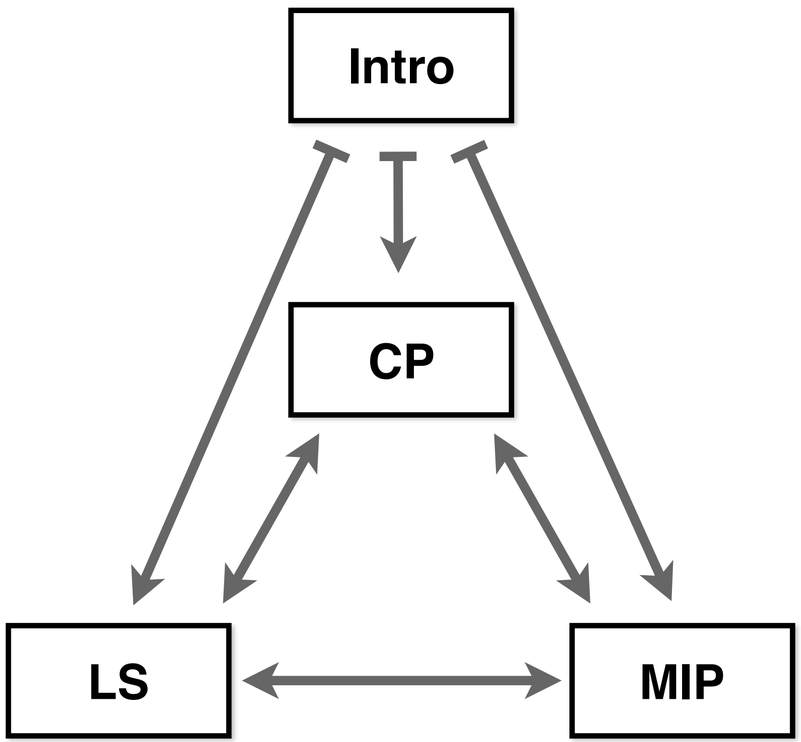
- We want this certificate to really valuable
  - this is a challenging but very rewarding class
- Regular
  - 7 on every problem, on average
- Distinction
  - 9 on every problem, on average
- Focus on submitting **good** solutions

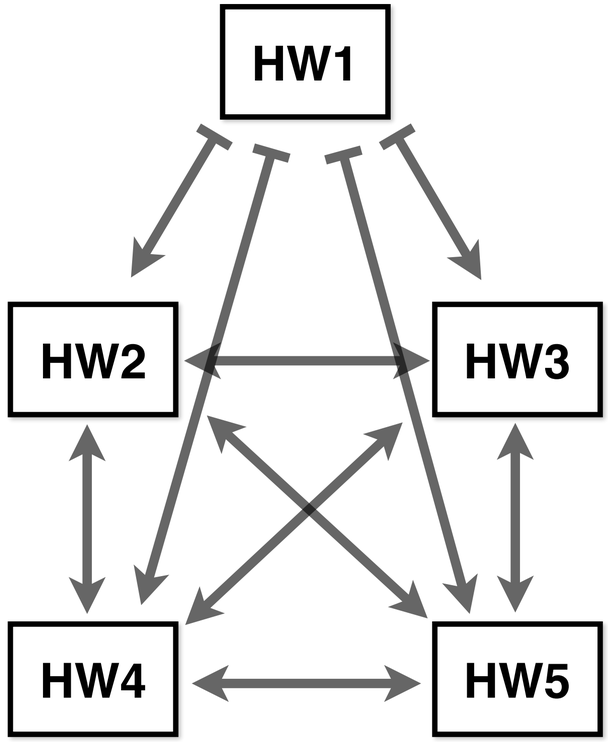
Time Commitment

- 15 hours per week
  - 1 to 3 hours of lectures
  - 10 to 14 hours of coding

- The assignments remain open
  - due date (recommended completion)
  - hard deadline (last chance)
  - return to assignments as you learn more

Open Course Design





How to Succeed in this Course?

Solving Tricky Puzzles

8	3	6	1		2	9		4
2	4		6	9		3	8	1
	9		3	4	8	2		6
	8			3			6	
	6	2					1	
	7	9		1	6		4	
9	2	8	5	6	1	4	3	7
6	5	4	7	2	3	1	9	8
7	1	3	9	8	4		2	5

Collaboration

- Do
  - discuss your solution ideas in the forums!
  - **share** your best objective values
  - refer to the syllabus for a detailed collaboration policy

- **Don't**
  - implement cutting edge research
  - copy solutions or code from anyone, your algorithm and code should be your own work

Enjoy!

- Optimization is fun!