CIS 1921: Solving Hard Problems in Practice

Instructor: Ishaan Lal
TA: Cindy Su
Location: Towne 327
Time: Thursday, 7:00 - 8:30 PM
Office Hours: Wednesday, 6-7PM (Cindy), Thursday, 6-7 PM (Ishaan)

Course Overview

*What does **Sudoku** have in common with **debugging**, **scheduling exams**, and **routing shipments**? All of these problems are **provably hard** — no one has a fast algorithm to solve them. But in reality, people are quickly solving these problems on a **huge scale** with clever systems and heuristics!*

*In this minicourse, we'll explore how **researchers** and organizations like **Microsoft**, **Google**, and **NASA** are solving these hard problems, and we'll get to use some of the tools they've built!*

Topics covered include: routing, scheduling, and packing problems; constraint programming in Python; SAT solving; optimization; operations research.

Prerequisites

CIS1210 is a prerequisite for this course. You are expected to be fairly comfortable with programming and familiar with graphs and graph theory. Experience with Python is helpful but not necessary. Knowledge of NP-completeness is not necessary but useful to motivate the course.

Grading

Homework: 60%, **Final Project:** 30%, **Attendance:** 10%
Homework 0 will be worth 8%, and the remaining 4 homeworks will be worth 13%

Homework will consist of 4.5 medium-sized programming projects, assigned roughly biweekly.

You will work on a final project of your choosing – you might choose to solve a practical problem using the tools we'll learn, implement a solver with modern techniques, or even explore new methods of your own!

Late Policy

You will have 48 free late hours to use throughout the semester. You do not need to inform the instructor before using your late hours, as Gradescope will take care of this. *Homeworks cannot be submitted more than 48 hours after the due date.* For each minute exceeded of the 48 late hours, there will be a 1/60% deduction on the overall grade (this is equivalent to 1% for every hour, but broken down to the minute).

Anticipated Schedule

Week 1	Introduction to Hard Problems Review of NP-Completeness Introduction to SAT	HW0: Finger Exercises Assigned (1wk)
Week 2	SAT solvers (Intro, and utilization in Python) Graph Coloring Stable Matchings	HW1: Sudoku Assigned (2wks)
Week 3	Algorithms for SAT Backtracking, 1 & 2 Watched Literal, DPLL Algorithm	
Week 4	Implication Graphs Backjumping Unique Implication Points Debugging SAT Solvers	HW2: PennSAT Assigned (2wks)
Week 5	Linear Programming Integer Linear Programming Mixed Integer Programming	
Week 6	Mixed Integer Programming Continued Knapsack Problem	HW3: Kidney Exchange Assigned (2wks)
Week 7	Potential Guest Lecture	
Week 8	Constraint Programming (CP-SAT) Cryptarithms, Interval Variables, etc.	HW4: Grace Hopper (2 wks)
Week 9	More Constraint Programming (more constraints, optimizations, etc)	Project proposal due

Week 10	From CP to SAT Finite Domain Propagation Lazy Clause Generation	
Week 11	Traveling Salesman Problem Vehicle Routing Problem	
Week 12	Local Search in SAT, TSP Greedy SAT WalkSAT	
Week 13	Genetic Algorithms Stochastic Universal Sampling Knapsack Revisited	Project check-in due
Week 14	Project Presentations	