Submission Deadlines (11:59pm for all dates):

- **Proposal:** Friday 11/18
- **Project Update:** Tuesday 11/29
- **Final Project Docs and Code**: Monday 12/5
- Prepare to have another student run your code from your machine using a hard-copy of your README to understand what to do and what results to expect during our final exam period.

Instructions

For your final project, think of an AI task that you can implement. Here are a few examples:

- Create a program that puts a natural language front end on a specific task, like getting movie reviews or booking airline tickets.
- Create a Bayesian network that comes up with the most probable solution to a crime given the evidence.
- Make an AI player for a game that uses some strategies beyond alpha-beta pruning.
- Explore Supervised Learning by implementing the ID3 algorithm to make an expert system.
- Make an automatic text generator. Or better - use music or art! (no open.ai)
- Solve a problem like the traveling salesman problem using genetic algorithms.
- Make a simple neural net to classify handwritten symbols.
- Create an expert system on some domain of your choice with a natural language front end.
- Use prolog to make a constraint satisfaction solver for a logic game.
- Make a joke-generator.
- Explore an online AI tool - and use this in an implementation of some kind:
  - Amazon Web Services
  - IBM Watson
  - Google's Machine Learning Tools, like TensorFlow

- Google's Natural Language API

Submissions

For your project, in addition to handing in your source code, you will submit two documents:

- README file that must include:
  - Project title
  - Concise yet thorough description of the project
  - Instructions for how to use your project
  - Overview of some sample use cases and results
  - Credits - particularly if you used resources to help you build the code
  - More detail and optional things to include here: https://www.geeksforgeeks.org/what-is-readme-md-file/
- A project report that must include:
  - An overall project statement
  - An overview of the algorithm used to solve the problem. Explain how it works, and include pseudo code.
  - An analysis of your solution - what are strengths/weaknesses, how does it relate to what we did in class
  - A summary of your results - how well does it work?

Grading

This project is worth 20% of your class grade. It will be assessed as follows:

- 20%: Quality of README document. Is this sufficient for another student to understand your project, how to run your code, and what results they should expect?
- 45%: Sophistication, difficulty, and functionality of the code.
- 20%: Written document, results, and thoughtful analysis.
- 5%: Effectiveness of README for peer running of code, and quality of short demo of project (is it clear you prepared to demo your work?)
- 5%: Good coding practice, particular focus on variable names and code comments.

    - 5%: Meeting all due dates on the checklist - proposal and update

During the final exam period (after 1 hour paper exam) we will have 2 hours. You will have someone else run your code and evaluate using the README (15 minutes), then each student will have 3 minutes to show their project to the class.