There will be a single programming project for CS 4701. The goal is to exercise knowledge you acquire from your AI courses (CS 4700 or others in our curriculum) through the development of a large-scale system. The major effort of the project should involve AI, as opposed to user-interface design, client-server software, etc. These can certainly be part of your system, but it should not be where most of the work takes place – beware, it is very easy to get sucked into programming effort that is not central to the goal of 4701.

Students should work in teams of 3-5. Expectations of projects will be proportional to the number of students on a team.

Your project will have the following milestones/assignments. The only thing that will be graded are final submissions; however, points will be deducted if you do not follow the instructions concerning submissions for the milestones – many of them require you to do more than one thing as part of your submission. PLEASE READ ALL SUBMISSION INSTRUCTIONS CAREFULLY.

1. **Friday, March 9, 1:15pm: Teammate Selection**: Submit your team information as follows:
   a. ONE MEMBER of your team should go to Gradescope and do the following:
      - Upload a file with this same teammate information. This can either be a screen capture of the Google form or as a separately prepared file.
      - Link in each of the other team members via Gradescope.
   b. ONE MEMBER of your team should go to [https://goo.gl/forms/6zxAB30pruT3EbvK2](https://goo.gl/forms/6zxAB30pruT3EbvK2) and provide the requested information.

2. **Friday, March 16, 1:15pm: Project proposals**: Submit your proposed project.
   a. ONE MEMBER of your team should upload a copy of the proposal to Gradescope and link in your teammates.
   b. Submit a HARDCOPY of your proposal to my office (Gates 352) by 1:15pm on March 16. I will also accept submission of the hardcopy at the start of the 4700 lecture that day at 1:25pm.
   c. ONE MEMBER of your team should go to [https://goo.gl/forms/CGK6vu00mOqcYaK52](https://goo.gl/forms/CGK6vu00mOqcYaK52) and provide the requested information.

Your proposal is intended to be short and to the point. It should have the following components:
- Title of your project
- Team name
- Team member names with Cornell NetIDs
- Keywords describing your project. This should include the area of AI that your project involves. It can also include other helpful information, such as, if you’re building a game playing program, the name of the game.
- A clear and concise description of what you plan to do (3-4 paragraphs)
  - Make sure to clearly identify what aspect of AI you are exploring in your project.
  - Discuss how you are thinking about evaluating your system. Saying, for example, “I’m going to build a system to play <game>” isn’t enough. How will you know how good
your system is? For example, if you’re building a game playing program is there an existing program you will go head-to-head against? (And if so, be realistic about it – for example, it’s easier to go head-to-head with an open-source game playing program that you can download and run locally than to do so with a game that you can only play via, say, an iPhone.) Or will you have it play against humans? (And if so, how will you know how good the human players are, to calibrate your results?) If you plan to explore different solution approaches, how will you determine their relative merits? Your final report will need to have a section on evaluation, and this documents that you’ve selected and scoped out your project with this in mind.

- A timeline for your implementation and evaluation. Don’t just say “We’ll do X first, then Y second, then Z third.” Think about how you will break up your project into pieces, how long each piece should take, which things can be done in parallel versus must be done sequentially. This timeline is not a contract, it’s to help you think about how much you’re proposing to do.
- Who will do what. Again, this is not a contract, it’s to make sure you don’t just say, “We all thought this will be cool” and have thought about how your team is well-suited to tackling the project.
- Any existing resources (software, data, etc.) you are thinking of relying on. You do not need to reinvent the wheel. If there is software or data resources that you want to use in your project that’s ok. It is the work you do beyond these that is graded. Thus, for example, if you propose a project that involves machine learning this is where you would demonstrate that you’ve thought about where you will find data. Finding existing data is fine, just as is creating your own data resource if need be, depending on the nature of your project.
- (If relevant) How this project relates to other work team members may be involved in. Your project can be connected to something else you are currently doing. The only requirements are that (1) the work for the project must be disjoint from what you are doing elsewhere, (2) you disclose that this is the case, and (3) all parties (both this project’s teammates as well as those involved in whatever else it is connecting to) are aware of its use in this fashion. It is extremely important that you are clear what is distinct in this project compared to the work you would be doing otherwise.
- (If relevant) Citations. List any technical papers, websites, etc., you used in formulating your project

This is not intended to be long. While there is no explicit page limit for this, as a rough guideline 1 page will probably be on the short side, 4 pages on the long side. However, the most important thing is to take the space that you need to convey all the necessary information. If you are planning to do a game playing program, feel free to ask me if it’s a game I already know about (such as backgammon, checkers, chess, etc.). You would not need to describe the game in any detail if I already know it. If it’s a game I don’t know you would have to describe it, which means your proposal might very well be longer than one that’s about a game that I know. That’s fine. Similarly, depending on your project you might find that you want to include various figures to explain things, and that, too, might make your proposal longer. That is fine too. If your project is
related to something else you’re doing, or if you need to take time describing existing software or data resources, you’ll also likely have a longer proposal. All this is fine.

On (extremely) rare occasions circumstances have arisen that made a team want to change projects midstream. You cannot do so without permission. Should that wind up the case you would need to prepare a replacement proposal following the guidelines above.

3. **Wednesday, April 11, 1:15pm: Status Reports:** There will be one status report due about half-way to the due date of the final report.

   a. ONE MEMBER of your team should upload a copy of your status report to Gradescope and link in your teammates.
   b. Submit a HARDCOPY of your proposal to my office (Gates 352) by 1:15pm on April 11. I will also accept submission of the hardcopy at the start of the 4700 lecture that day at 1:25pm.

Now that you’ve actually been working on the project there are undoubtedly areas of the proposal that you can better flesh out now, or you may have found that your expectations shifted. The goal of the status report is to pin down more about the project details, and more generally to make sure work is progressing. Your status report document should take your original project proposal document and directly add to it. One good way to show what is new versus what was already there is to use a different colored font, like blue, for your new text to distinguish it from the earlier proposal text.

In an ideal world your status report would simply involve a new sentence saying saying that things are going exactly as you planned. Of course that never happens. The updates that you provide will typically include one or more of the following:
- We didn’t realize that X was hard, so it’s taking us longer to do it and we had to scale back Y.
- We said we would do X but it doesn’t seem to be working so we decided to instead do Y.
- We discovered this great software/data resource which simplifies X, which means we’re now going to try to do Y.
- We learned that the great software/data resource X that we expected to use wasn’t available, so we’re instead using Y.
- We thought more about how to evaluate the project, and instead of doing X we now realize that it will be better to do Y.

If things that were in the proposal have changed, say so. If there was something you didn’t realize you would need to do and now are need to do it, say so. And so on.

Please also provide an updated timeline. It should say which items in the original timeline have been done, and what (if anything) you are changing in the timeline given what you’ve learned about the project thus far. Similarly, if the planned division of labor has changed, say so. Please also add any additional references to technical papers or websites if relevant.

4. **Monday, May 14, 4:30pm: Final Project Report: GROUP:** BECAUSE OF THE TIMING OF THE 4700 FINAL EXAM THERE WILL BE NO EXTENSIONS TO THIS DEADLINE. Please plan accordingly!
a. ONE MEMBER of your team should upload a copy of the proposal to Gradescope and link in your teammates.

b. Submit a HARDCOPY of your proposal to my office (Gates 352) by 4:30pm on March 14.

Your final submission should be a roughly 10-page paper. As with the proposal, this is a guideline, but to help you judge this 5 pages is probably too short, 15 pages too long. Your report might be longer if you require many figures, for example. Your report should include the following:

- Title of your project
- Team name
- Team member names with Cornell NetIDs
- Keywords describing your project
- Say what you had planned to do, then explain what your project wound up being. Make sure to clearly identify what aspects of AI were involved.
- Explain what assessment you conducted concerning your project. How well did you do? Be clear about what questions you asked and what answers you got. They will likely be quantitative. They might be about how well your system worked (perhaps along multiple dimensions), or it might be about how different elements of the system design contributed to overall success. Your project is not complete if you have not thought about how you can evaluate your accomplishments; the nature of this evaluation is part of the design of your project.
- Make sure to include citations to any technical papers, books, websites, etc., that played a role in the formulation and execution of your project.

Note what this DOESN'T include: code listings, software architectures, etc. If you’re discussing the different modules of your system your report is at the wrong level. Your report should be about the ideas. At this point in your studies I’m presuming that you can architect and code software, and what heft there is in your hacking should be apparent through the ideas and evaluation you conducted.

I encourage you to get feedback on your report from others before you submit it, so that my eyes are not the first to read it. There is no excuse for a report that has many needless typos or poor writing, and it will affect your grade.

4. Monday, May 14, 4:30pm: Final Project Report: INDIVIDUAL

Each team member should also use Gradescope to submit a short (1-2 page) personal report on the project. It should discuss your individual contributions to the project, and what personal lessons were learned from doing it.

BE AMBITIOUS! It is far better to be mediocre on something interesting than perform perfectly on something easy. The amount of effort that your project shows will be an important factor in your grade. Note that this refers to getting disappointing results, not that the evaluation execution can be disappointing. Be clear why you think you were successful, or not. Sloppy or hand-wavy discussions of performance will significantly negatively impact your grade.
Sample project ideas:

You are given free reign to pick your project, but the following are examples of possible project topics:

- **Game playing programs:**
  - Backgammon
  - Blackjack
  - Checkers
  - Othello
  - Hearts

- **Machine Learning:**
  - Make Predictions by Mining Social Media
  - Image Recognition
  - Forecast Financial Markets
  - Playing a Game
  - Learn Endgame Rules from Examples

- **Genetic Algorithms:**
  - Genetic Programming
  - Learning Natural Language Grammars
  - Using genetic programming for the automatic generation of computer programs

- **A theorem-proving system for some (small) subset of mathematics**
- **A program that generates automatic crossword puzzles, starting from a dictionary and an empty board**
- **A system that plays a video game (such as Tetris)**
- **A system that plays the repeated prisoner’s dilemma (see for example [http://serendip.brynmawr.edu/playground/pd.html](http://serendip.brynmawr.edu/playground/pd.html))**
- **The textbook includes project ideas among the exercises at the end of various chapters that might also provide inspiration for possible project ideas.**