CS 182: Artificial Intelligence
Final Projects

1 Important Deadlines

The final project will consist of the following four aspects:

1. Project Proposal
2. Status Update
3. Final Paper
4. Presentation, either:
   - Oral presentation in class
   - Poster presentation at computer science fair

No late days for any course project deadline.

These deliverables will be due at the following deadlines:

Aspect	Deadline
Project Proposal	10/30, 5pm
Status Update	11/24, 5pm
Oral Presentations	12/3, in class
Posters to Printer	TBD
Poster Presentations	12/7, 12-2pm
Final Project	12/10, 5pm

2 Goals and Scope

The CS 182 final project provides an opportunity for you to apply and extend the foundational concepts you have learned in the course. It is intended to encourage you to integrate ideas from different course components and allow you to delve deeper into areas of interest. Projects may be implementation, testing, and analysis of algorithms mentioned in lecture or described in the text but not covered in assignments (e.g., alternative search or planning algorithms) or design and implementation of an intelligent system that combines algorithms and representations from several course components (e.g., a more complex game-playing program or an intelligent advising system). Pick something that interests you. Students are expected to design and carry out final projects working in teams of two or three. You can use the Piazza forum to find partners. **Special Note:** If you want to undertake a theoretical project or work alone, please talk to Sasha or one of the teaching fellows before beginning your proposal.

A list of possible course projects is given on the projects wiki. Their content and scope are meant to be suggestive, not definitive. The teaching staff would be delighted to talk with you about possibilities. Each suggestion has an associated TF; he would be a good person with whom to speak first, but feel free to contact any of us. After the proposal, each final group will be assigned a course mentor who will be the best point of contact throughout the project.

Final projects that fall outside the specifics but within the spirit of these suggestions are encouraged, as are projects that draw on other computer science courses you have taken. It is, however, essential that the project draw on, and make connections with, major issues and ideas in search, representations, or uncertainty covered in the course. If the idea you have for a project varies significantly from the suggestions, or you are unsure of its suitability, please contact the teaching staff enough in advance of the project proposal due date that our discussions with you can help shape your proposal. Some places beyond the text and course readings to look for inspiration: Proceedings of the major AI conferences (AAAI, IJCAI, ECAI, ICAPS, AAMAS, HCOMP, UAI) and journals (Artificial Intelligence, JAIR: J. of AI Research, JAAMAS: J. of Autonomous Agents and Multi-Agent Systems). Most CS papers are available free online either through the ArXiV or through search on resources like Google Scholar.

If your project involves building a system that includes designing a GUI, make sure that you focus most of your effort on the AI aspects of the project and not the GUI per se. We will evaluate your project on the concepts it investigates and the results and on how well it demonstrates your comprehension of the concepts, techniques and issues we have covered in the class. The project grade will incorporate evaluation of the proposal, presentation (oral or poster) and final paper quality. As there is no final exam for CS 182, your final project is the major integrative element of coursework. It will account for 20% of your overall course grade.

3 Deliverables

3.1 Proposal

To ensure that you choose an appropriate project, you are required to turn in a 1–2 page project proposal by 5pm Friday, October 30. The proposal should begin with a clear, unambiguous statement of your topic, and include all of the following:

1. brief discussion of the problem you intend to investigate, a description of the system you intend to build and the algorithms you intend to implement; this description should mention the specific course topics (e.g. heuristic search, logical representation, sequential decision making) to which the project relates;

2. examples of expected behavior of the system or the types of problems the algorithms you investigate are intended to handle;

3. the issues you expect to focus on; and

4. a list of papers you intend to read to inform your project effort; this list will form the core of your project report reference list. If your project includes anything unusual (such as having significant systems demands), please state this as well.

This document will be worth 5% of your grade for the final project. Use it to demonstrate to us that you have completed some background work on your chosen topic. The proposal should resemble a CS 182 problem set in the sense that if one were to give your proposal to another student, s/he should be able to complete the project as well. Proposals should identify all members of the team and indicate how you intend to divide work on the project. We will review your proposal and return it to you with comments and suggested modifications.

3.2 Update

To ensure that you are on track with the project and to identify any issues on time, you are required to submit a short (1-page) report with a status update. The report should describe the problem you are working on, the progress you've made so far, and any problems you came across that you would like to get help with. With the status update each group should also identify whether they would prefer to give an oral presentation or present at the poster session.

3.3 Presentation

The oral presentations will be on Thursday, December 3 and the poster session will be on Monday, December 9. These presentations are a chance to explain your problem and approach, showcase what you've accomplished, and get advice on surmounting any hurdles you've encountered. Students are expected to attend both presentation sessions, as they provide an opportunity for you to learn from each other.

- Oral presentations will be allocated 10 minutes, and should be focused on key issues. You are encouraged to bring less than 4 slides, because often just one diagram or chart can explain the essence of your idea and save lots of presentation time. You need not prepare fancy graphics; just come prepared to explain your topic and share what you have discovered.
- Posters will be printed through a SEAS allocated fund for student poster printing. A good rule of thumb is to design a poster to act similarly to a presentation as a tool for walking readers through your work. Along with the poster each member of your group should be able to explain the project to a general audience at the fair as well as the TFs and other students in the class.

3.4 Report

You must submit a written report on your project and the complete, well-documented source code for it. The report should be 5-10 pages in length. It should describe the algorithms you implemented and the data on which they were tested and include an analysis of results or system performance (depending on the scope of the project). We recommend code.harvard or GitHub as a method of collaborating and submitting source code. We also recommend recommend using LaTeX for writing your report.

The report must contain all of the following content:

- A description of the purpose, goals, and scope of your system or empirical investigation. You should include references to papers you read on which your project and any algorithms you used are based. Include a discussion of whether you adapted a published algorithm or devised a new one, the range of problems and issues you addressed, and the relation of these problems and issues to the techniques and ideas covered in the course.
- A clear specification of the algorithm(s) you used and a description of the main data structures in the implementation. Include a discussion of any details of the algorithm that were not in the published paper(s) that formed the basis of your implementation. A reader should be able to reconstruct and verify your work from reading your paper.
- Analysis, evaluation, and critique of the algorithm and your implementation. Include a description of the testing data you used and a discussion of examples that illustrate major features of your system. Testing is a critical part of system construction, and the scope of your testing will be an important component in our evaluation. Discuss what you learned from the implementation.
- For algorithm-comparison projects: a section reporting empirical comparison results preferably presented graphically.

In addition, the report should include three appendices:

- Appendix 1 – A trace of the program showing how it handles key examples or some other demonstration of the program in action.
- Appendix 2 – A clear description of how to use your system and how to generate the output you discussed in the write-up and the example transcript in Appendix 1. N.B.: The teaching staff must be able to run your system.
- Appendix 3 – A list of each project participant and that participant's contributions to the project. If the division of work varies significantly from the project proposal, provide a brief explanation.

Your code should be clearly documented. Submit your code to the course dropbox. If that is not possible, you must provide a link to the code in Appendix 3. In either case, you should clearly indicate any code base that you used but did not develop. If you have any questions about these specifications, please ask the teaching staff.