

# Bard Computer Science Program

## Senior Project Guidelines

The student handbook states that the **Bard Senior Project** is “**an original, individual, focused project growing out of the students cumulative academic experiences.**” More than anything, you should own your project, with your adviser acting as a guide and resource. This document provides guidelines to ensure your computer science senior project proceeds smoothly and is successful. The following guidelines are grouped by time-scale, from project-long to daily expectations and responsibilities.

### Project Expectations and Responsibilities

- **Document:** The senior project document must be submitted on-time (by the **fourteenth Wednesday by 5:00 PM**) in the required format. Some students use Microsoft Word, but most use LaTeX for typesetting<sup>1</sup>. Include long excerpts of formatted code, scripts, data in appendices.
- **Poster:** You will present at the SM&C senior project poster session approximately two weeks after the project deadline. Before final printing, you will present a draft of your poster to the computer science program to gather feedback.
- **Topic:** Your project should be interesting and stimulating, both to you and the broader intellectual community. The topic should satisfy one or more of the following:
  - within the adviser’s area of expertise;
  - you have successfully completed coursework in area;
  - you have work experience in the area (e.g., BSRI, REU, internship).
- **Evaluation:** Computer science senior projects (and CS research in general) vary greatly, from the theoretical to the applied. Whatever the topic, the project should have clear goals, expectations, and an evaluation plan. One of the tenets of so-called *test-driven development* is to write tests first, before writing the production software. You should apply this approach to your project at the micro- and macro-scale. You should define and write the *testing* parts of your project first. This includes defining objective metrics of success and ways of measuring and visualizing those metrics.
- **Prospectus Talk:** You will give a short five minute presentation to the computer science community in the **seventh week** of the first term. You should describe the problem your senior project is addressing, explain why it is interesting, and include an evaluation plan.
- **Midway Board Meeting:** A midway board meeting will be held during completion week at the end of the first semester. Your board will consist of three faculty members, with two of those members from the CS program. As part of this meeting, you will provide the members of the board with 4–8 pages of your project so far, including a literature review.
- **Final Grade:** The quality of the work, the quality of the write-up, and your overall effort all factor into your final grade. Based on those factors, your board will recommend a grade, and the program as a whole will assign the final grade. More details can be found later in the **Patterns of Successful Projects** section.

---

<sup>1</sup> <http://math.bard.edu/bloch/tex/>

## Weekly Expectations and Responsibilities

- **Weekly Meeting:** Meet with your adviser weekly. You should come prepared with recent accomplishments and challenges, and a tentative plan for the next week.
- **Weekly Report:** Have something to show every week, even if it is just a paragraph summarizing what you did that week. This is not just for your adviser's sake, but also for you. Individually, your weekly report allows you to take stock, reflect a bit, and plan for future weeks, and collectively the reports allow you to chart your progress. For example, the report might note the latest entries in your annotated bibliography, a code diagram, or a graph of recently collected data.

## Daily Expectations and Responsibilities

- **Work Independently:** This is your project, so work as independently as possible, but listen to advice.
- **Substantial Effort:** Your project should be demanding. Although the work does not have to result in a publication, it should strive for publication-quality.
- **Calendar:** Keep a calendar for planning milestones in your project, for example, due dates, board meetings, but also for planning design and coding milestones, and data collection. Each senior project timeline will be unique, but it is important you chart out your own set of milestones with your adviser.
- **Lab Notebook:** Keep a notebook detailing challenges, questions, hunches, designs, accomplishments, and references. The non-electronic, offline notebook works just fine, but the online electronic variety is recommended (e.g. wiki, google doc, text file).
- **Annotated Bibliography:** As part of your project you will read scientific literature. From the earliest stages of your project, it is wise to create an annotated bibliography (including citation information and a small summary). Software like Mendeley, Bibtex or Endnote can be initially painful to use, but result in less pain overall. Trust us.
- **Version Control:** Use some version control system (e.g. git, mercurial) for any software you develop. Versioning your data and documentation is also recommended. Again, making your software repository available online (like your notebook, and bibliography) will improve communication with your adviser.

## Patterns of Successful Projects

The quality of the work, the quality of the write-up, and your overall effort all factor into your final grade. The following factors are used to determine if a senior project is ultimately successful. Note, these categories are not necessarily mutually exclusive and not assessed in such a binary fashion.

ASPECT	SUCCESS PATTERNS	SUCCESS ANTI-PATTERNS
<b>effort</b>	works steadily both semesters	waits until last two months for coding and writing
<b>command of topic</b>	full understanding of background literature and methods employed	substantial gaps in knowledge of content or methods
<b>creativity</b>	related research has been published in the last ten years, and this work offers novel ideas	content and method covered in standard classwork
<b>independence</b>	effectively negotiates technical and conceptual obstacles and explores interesting new directions while incorporating feedback from adviser	dependent on adviser to set goals week by week; easily derailed
<b>evaluation</b>	objective evidence of success (e.g., statistics, proofs, software requirements); compared with alternative approaches	relies on personal, subjective measure of novelty
<b>presentation</b>	effectively communicates work during boards and during poster session; effective communication includes rhetoric and use of audiovisual aid	sloppy poster; unable to answer board questions
<b>writing</b>	the paper is well-organized, provides a thorough literature review, free of typos, properly formatted. Uses lexical, sentence, paragraph, and larger structures to construct an effective argument.	paper lacks organization, figures are badly formatted and captioned, typos galore, missing important technical descriptions

## Resources

- **Writing for Computer Science** by Justin Zobel
- **Style: Lessons in Clarity and Grace** by Joseph M. Williams et al.
- [“How to Write a Great Research Paper”](#) by Simon Peyton Jones
- [“Ten Simple Rules for Mathematical Writing”](#) by Dimitri P. Bertsekas