



Computer Science 340
Programming Languages
Siena College
Fall 2019

Language Project

Matching groups to languages through 11:59 PM, Thursday, October 24, 2019
Groups must be formed and repositories created by: Monday, October 28, 2019
Proposals due: 11:59 PM, Monday, November 4, 2019
Progress reports due: 11:59 PM, Monday, November 18, 2019
Complete paper drafts due: 11:59 PM, Monday, December 2, 2019
Presentations: during our last two scheduled class meetings on Friday, December 6, and Monday, December 9, and our exam slot (9:00-10:30) on Wednesday, December 11, 2019
Final submission: 4:00 PM, Monday, December 9, 2019

As you know, this course requires you to complete a final “language project” that will determine 15% of your grade. You will choose a programming language to study in more detail than we have as a class. Most projects will involve significant programming in your chosen language, but that is not the only focus. You will also need to investigate the interesting design features of the language, the kinds of problems the language is well-suited to solve, how it compares with the languages we have studied, and the history of the language. In most cases, you will need to find a compiler or interpreter for your language that we can run under FreeBSD, Linux, Windows, or on a Mac. A successful project could be done by choosing a language of more historical interest for which no current implementation can be located, but non-programming aspects of the project will need to be emphasized. You may also choose to focus more on the compilation and/or interpretation mechanisms of a language you already know. You will report on your project through a formal research paper and brief (approximately 20 minute) presentation at a minisymposium to be held at the end of the semester.

You are expected to work in groups of size two or three. Individual projects may be approved on a case-by-case basis, as would groups of size four. Groups must be formed and all repositories created (regardless of group size) by Monday, October 28, 2019.

Matching Groups to Languages

Through 11:59 PM, Thursday, October 24, 2019, we will work toward matching group interests to make sure we end up with a nice variety of projects by contributing to a shared document. Please start thinking about a language and what you would like to implement in that language right away and put those ideas in the document. This document can also help those looking to form groups to find others with similar interests.

The Proposal

By 11:59 PM, Monday, November 4, 2019, submit a proposal (by committing and pushing a file `proposal.pdf` in your repository), at most one page in length, that briefly describes the

language you wish to study and why, and what you plan to implement in your language, and how you plan to go about it. Describe the major milestones for your project, a rough schedule for achieving these milestones, and which milestones you believe are most important for your project to be considered a success. If you will need access to any special hardware or software, include that in your proposal. Your proposal should make it clear that you have an interesting and worthwhile language and problem to solve using that language, and that it is feasible to study and use it in the time available.

The Progress Report

By 11:59 PM, Monday, November 18, 2019, submit a progress report (committed and pushed as `progress.pdf` in your repository). This should consist of a detailed outline (or better yet, an early draft) of your paper including sources, and a description of the design and current implementation status of your software, as well as a more specific timetable for completion of the project. You must also submit an annotated bibliography with your progress report, detailing the sources you have considered (likely including some you intend to use and some you do not intend to use).

The Paper

This is to be a formal research paper, and should be organized as such. You should begin with a title, author list, and abstract. The main body of the paper should be organized into sections including *(i)* an introduction in which you describe the general topic and the particular aspects you will be examining, *(ii)* one or more sections comprising your main text, where you describe what you have done, how you have done it, and what you have learned, *(iii)* a conclusion, which should include ideas for future investigation into your topic which were beyond the scope of your paper, and *(iv)* a complete list of citations. Citations of web pages are acceptable in some circumstances, but books, articles in conference proceedings or journals, or technical reports are preferred.

Proper English and a good technical writing style are important. Writing well is very difficult – it is an iterative process and cannot be done all at once. Be precise and be concise. Check your spelling and grammar carefully. Most papers will be around 15 single-sided pages, using 1.5 spacing, one inch margins, and a 12-point Times Roman font (or as close as you can come). Please do not adjust margins and font sizes to force a certain length. You are encouraged but not required to use \LaTeX to typeset your paper. My \LaTeX example in a GitHub repository is already set up with an appropriate format. Length is not important – content and quality are. Papers shorter than 10 pages or longer than 20 pages are acceptable, if the length is appropriate for the content.

Submit a draft, as complete as possible, by committing and pushing a PDF file `draft.pdf` by 11:59 PM, Monday, December 2, 2019. The more complete this draft, the better feedback you will receive to be able to improve it and the more you can focus on programming over the last week or two. You may also submit additional drafts for feedback, but keep in mind that it may take a day or two to get to them. Submit your final version by pushing and committing a PDF file `paper.pdf` by 4:00 PM, Monday, December 9, 2019.

The Project

You should submit your source code and instructions on how to build and run it. The software should be described in the paper. Please make arrangements to demonstrate your program(s). Source code should be committed and pushed to your repository and demonstrations completed by 4:00 PM, Monday, December 9, 2019.

The Presentation

Each group will present a summary of their work to the class. Include background information on your language, the motivation for your project, a description of what you did, and a summary of what you learned. Software demonstrations may also be appropriate. Prepare slides or web pages. Rehearse your presentation, paying special attention to timing. Presentations of 15-20 minutes each will take place during a class minisymposium to be held during our last two scheduled class meetings on Friday, December 6, and Monday, December 9, and our exam slot (9:00-10:30) on Wednesday, December 11, 2019. All group members must participate in the presentation. Attendance is required for the entirety of all presentation sessions, not just when your group is speaking. Please also submit your presentation materials through a directory in your repository.

Grading

This final project accounts for 15% of the course grade. The grade will be based on all aspects of the project, including the proposal (5%), the progress report (5%) and annotated bibliography (5%), the design, documentation, style, and correctness of the software developed (0–30%), the content and writing style of the complete draft (10%) and final version (30–60%) of the paper, and quality of the presentation (15%). The breakdown between software development and the final paper will depend on the amount of programming required for each project, and should be negotiated between the submission of a proposal and the progress report. No credit is given for attendance at the minisymposium, but penalties will be applied for missing all or part of it.

Academic Honesty Guidelines

Collaboration within a group is unrestricted. Since each group is working on a different project, you are free to discuss your projects with each other. If you wish to use or refer to any software libraries or outside source code beyond those that come standard with your language, check first, and cite their usage appropriately. All sources must be cited properly. If in doubt about anything related to Academic Honesty, ask now and avoid problems later!

Final Thoughts

You have several weeks, so the expectation is for several weeks of work. You will not be able to do a good job if you put it off. The expectation is not for you to produce a Ph.D. thesis, but this project should be much more than your average problem set assignment.