CSC 4101: Programming Languages

Syllabus

Fall 2020

Instructor

Gerald Baumgartner
Email: gb at lsu.edu
URL: https://www.lsu.edu/eng/cse/people/faculty/baumgartner.php
Pronouns: he/his/him

Course Summary

Principles of programming language design; specification of syntax and semantics; underlying implementation of block structured languages; dynamic memory allocation for strings, lists, and arrays; imperative versus applicative programming; logic programming; modern programming languages.

Prerequisite

According to the course listing, the prerequisites are:

> CSC 3102: Advanced Data Structures and Algorithm Analysis.

Course Format

We will use the flipped classroom teaching model. I will make lectures available as videos for you to watch at your own time. During regular class times, I will answer questions about the lecture content, homeworks, and programming assignments and, if needed, go over additional examples. I will also record these classes and make the videos available afterwards. The amount of class time we will need for that will depend on what is happening in the lectures and with the programming assignments. Occasionally (rarely) we might skip class, but it will also be rare that we will need the full class period.

Office Hours

> Gerald Baumgartner: MW 9:30-11:00am, https://lsu.zoom.us/j/2180487982, gb@lsu.edu
> TAs: Monazil Chowdhury, mchow15@lsu.edu; Qifan Zhang, qzhan25@lsu.edu

Other office hours by appointment.

Please, let me know by email when you are planning to connect during office hours, so I can space out students if needed and to make sure that I'm in front of the computer at that time. I will maintain a Zoom waiting room.

Important Dates

- Hurricane Marco: Aug 24
- Hurricane Laura: Aug 26
- Labor Day: Sep 7
- Midterm Exam: Mon, Oct 12, 12:00–1:20pm
- Turkey Day Break: Wed, Nov 25
- Final Exam: Thu, Dec 10, 7:30–9:30am

Both exams are comprehensive.

Reading

- Michael Scott, *Programming Language Pragmatics*, 3rd or 4th ed., Morgan Kaufmann Publishers, 2009/2015. An e-book version is available through the LSU Libaries' webpage: https://www-sciencedirect-com.libezp.lib.lsu.edu/book/9780123745149/programming-language-pragmatics
- Robert W. Sebesta, *Concepts of Programming Languages*, 11th ed., Addison-Wesley, 2016 (optional).
- Peter Norvig, *Teach Yourself Programming in Ten Years*, http://norvig.com/21-days.html, 2001.
- Guy L. Steele Jr., *Growing a Language*, OOPSLA '98, ACM, 1998.
- Henry G. Baker, *I Have a Feeling We're Not in Emerald City Anymore*, ACM SIGPLAN Notices, Vol.~32, No.~4, April 1997.
- Richard P. Gabriel and Ron Goldman, *Mob Software: The Erotic Life of Code*, OOPSLA '00, ACM, 2000.
- CSC 4101 Mailing List csc_4101@eng.lsu.edu.

Programming Assignments

There will be three programming assignments, which will be due at midnight (11:59pm) of the due date. The due dates will normally be at the end of a week. The first two programming assignments will be in Python. The third programming assignment will be in Scheme. The first two programming assignments will be in teams of two, the third will be done individually. Since all program assignments are submitted and graded on a Linux server, submissions must be compatible with the language versions installed there. There will also be small programming problems, such as 1-10 liners in Scheme, ML or F#, and Prolog as parts of homeworks.

A penalty of 10% will be assessed for each day a project is late up to a maximum of 30% after which it will not be accepted. The final project may not be turned in later than Saturday midnight after the last day of classes.

Homeworks

There will be five homework assignments, which will be due at midnight (11:59pm) on the due date. The due dates will normally be on a day on which we have class. Homeworks will be submitted as PDF files (scanned or converted from images) on gradescope.com. A penalty of 20% will be assessed if a homework is submitted two days late. Homeworks later than that will not be accepted.

Quizzes

There will be several short (about 5-10 minute) quizzes on Moodle, which are normally due at the end of the week. The purpose of the quizzes is to ensure you stay on track with watching the lecture material. I expect that we will have approximately 10-12 quizzes, i.e., nearly one per week.

Annotated Bibliography

Honors students who want to honorize this course as well as graduate students also must submit an annotated bibliography on a programming language research topic as an additional assignment.

Grading

The following weights will be used for grading. The quizzes all weigh equally. The homeworks all weigh equally. The first two projects are worth 13% each, while the third project is worth 10%.

Projects:	36%
Homeworks:	15%
Bibliography:	3%
Quizzes:	5%
Midterm:	20%
Final:	24%

For honors and graduate students the maximum score with the bibliography assignment is 103 points. Before computing the letter grades, that total score is then divided by 1.03.

The course will be graded partly on a curve. For this reason, I will deduct points rather liberally and I will encourage the grader to do the same. Don't be too upset if you don't get what you consider to be a high score. When grading on a curve the absolute score is not that important. To give you an idea of where you are standing in class, statistics about the scores will be provided periodically.

In the past, the letter grade cutoffs have been usually in 15 point increments. Depending on the difficulty of the exams and the strictness of grading, the cutoffs might be moved down a little. After that, each letter grade range will be divided into three parts for +/- grades. E.g., the cutoffs for the major letter grades might be 85-70-55-40.

Topics

This list of topics is an ordering of subjects that will be covered with the corresponding assignments. Each topic will roughly take a week and a half, with a bit less time for topics 1 and 10.

1. Introduction

2. Functional Programming, Scheme, HW 1
3. Syntax and Parsing, HW 2
4. OO Programming, Project 1
5. OO Language Design, Midterm
6. Types and Scopes, HW 3
7. Functions and Parameter Passing, HW 4
8. Function Implementation, Project 2
9. Functional Programming, ML or F#, HW 5
10. Logic Programming, Prolog, Project 3

Due Dates and Grading

The time allotted for each homework assignment will be made quite generous, as such, the penalty for turning in late is high: 20% for turning it in in the next class period. Homeworks need to be submitted as paper copies, not electronically.

Since the time needed for finishing a programming assignment is harder to estimate and to allow fixing severe bugs that show up close to the deadline, programming assignments can be submitted up to three days after the official deadline. For each day past the deadline, a penalty of 10 percent will be incurred. Programming assignments will be submitted electronically. Projects will be due at midnight.

Without prior arrangements in case of extenuating circumstances, submission of homeworks and projects past the late deadline is not allowed, such work will not be graded, and you will receive no credit. It is your responsibility to make sure that you have completed your work with enough time to submit your materials.

Grading disputes can be submitted in writing with accompanying documentation, or in person during regular office hours. It is course policy that whoever graded the work will be responsible for handling disputes. In general, I will grade the midterm exam and the final exam. The graders will grade the homeworks and the programming assignments. Grades become final one week after a homework, project, or exam is handed back. This should leave ample time for resolving grading disputes.

Homework Standards

All written work submitted must carry the student's name and must be reasonably neat and well organized. Any work that cannot be read easily may score zero points. A reasonable standard of English expression and grammar is also required. The same requirements apply to exams.

Programming Standards

The algorithm used must be essentially correct. Obviously, the program should (compile and) run. Because of the complexity of some of the programs, very little or no credit can be given for a program that doesn't run. If a program dumps core or throws a runtime exception, only partial credit will be given.

Since programming assignments might build on top of a previous project, it is very important to get each submission to run without runtime exceptions or core dumps and to structure the program so it can be easily extended.

I expect your work to exhibit high standards of programming style and layout, reflecting your expertise as a computer professional. Poor style and documentation may result in points being deducted.

Honesty

I will treat you as professionals, and you should plan on conducting yourself as such. This course presents many important concepts you will need throughout your career as a computing professional, so it is important that *each student* do *all* the assignments and projects and learn the material.

There will be several homework assignments and programming projects. You are free to discuss these assignments with others. However, the programs and homework solutions you submit are to be developed by yourself (or by your team). *Cheating is a very serious offense and will not be tolerated.* The grader or I may use tools for detecting cheating on programming assignments. Reverse-engineering the reference implementation or supplying others with material is also against this rule. Similarly, making homework solutions or code available publicly on a web site, such as Github, is also considered a cheating offense. I have archived submissions of programming assignments from previous years. Copying from a student from a previous year or making your code available to a student in the future is similarly a cheating offense. The policy is that the supplier and receiver of information will both be reported to the Dean of Students.

Save all handwritten notes and printouts you generate as you work on a project and keep them until the end of the quarter so as to protect yourself in the event that someone "borrows" your program, or the version you submit is lost.

Computer Account Security and Use

To help others resist the temptation of using your work, you should maintain proper security on your computer account. In particular, keep your password from others and do not alter the protection on any of your files. To give others access to your account or files or printouts of your programs is the same as giving them the information directly and will be dealt with accordingly. Any trouble with computer accounts should be referred to an instructor as soon as possible.

When a program has been submitted electronically, you should maintain an *unedited* version of what you submitted (with the correct time stamp) until after that program has been graded. It is also beneficial to use version control software such as git or CVS to keep track of all versions of files so you can revert back to an old version if necessary.

Intellectual Property

The videos of lecture material and classes as well as any handouts and the skeleton code for programming assignments are the intellectural property of myself and LSU. It is a copyright violation to share them with anybody outside the class. I will make videos available on YouTube, but these videos will not be public. They will be unlisted vidoes on my YouTube channel. You can access them through the links I provide on Moodle, but do not share them with anybody outside the class without my permission and do not upload them to other sites.

Behavioral Misconduct

Per section 5.1 of the Code of Student Conduct, the Code applies to conduct that occurs on the Campus, at LSU-sponsored activities, and/or when the Student or RSO is representing LSU. The University shall have discretion to extend jurisdiction over conduct that occurs off campus when the conduct adversely and significantly affects the learning environment or University community and would be in violation of the Code if the conduct had occurred on campus. This includes behavior that may occur in a remote learning environment, such as email, discussion forums, zoom webinars, or any other platform or solution used for a course. In determining whether or not to extend jurisdiction, the University may consider its ability to gather information. Potential violations of the Code can be reported through LSU Cares.