CS 314: Principles of Programming Languages

Fall 2019 - Prof. Francisco

Objectives

- Primary objective: Learn new ways of thinking about problems and programs.
- Secondary objective: Make it easier to learn new languages by learning principles that apply to many languages.
- Tertiary objective: Learn some interesting languages.

Prerequisites

- CS 205, CS 211
- Java; the memory model of C, including pointers; predicate calculus

Books

- Recommended: Michael L. Scott, *Programming Language Pragmatics*, 3rd edition

The textbook does not cover all of the material for this course, and is not a substitute for attending class.

Work

- You are expected to attend all lectures and recitations, in both body and mind. That is, please do not access Facebook, games, email, or the like during class.
- There will be three exams, one of which will be a final, as well as graded programing projects and homework. The homework and the projects only count for a small part of your grade directly, because there is no way to be sure that they are really your own work. They will however count for a large part of your grade indirectly. If you do not do the homework and projects you will not learn the material as well. Furthermore, there will be exam questions directly based on the projects and homework.

Tentative Grading

Your course grade is based on homework, projects and exams, with the following tentative weighting:

Homework (all together): 10%
Projects (all together): 15%
Midterm 1: 20%
Midterm 2: 20%
Final exam: 35%

Late work

Homework and projects must be submitted via Sakai. Work will not be accepted late. There will be no “grace periods” or degrees of lateness. Work submitted one second late is late. You should know by now that under load computers are more likely to malfunction, like Sakai when 100 student press 'submit' at the same time. Plan ahead, and do not try to submit at the last moment.

Course grades

Your course grade will be computed as follows: Each score (for each exam and each project) will be converted to a percentage score. These scores will be averaged (separately) for homeworks and for projects. The percentages will be added, with weights as above. I do not publish or discuss letter grade cutoffs. Two rules of thumb to gain plesant outcomes:

0. Relative: Be above the mean

1. Absolute: Have >75% of the available points

Collaboration

All students in this class are expected to be familiar with the The DCS academic integrity policy, which can be found at http://www.cs.rutgers.edu/policies/academicintegrity/ , and to abide by it.

We do encourage you to talk to your classmates about projects, provided you follow the Gilligan's Island Rule; after a discussion each student should discard all written material and then go do something mind-numbing for half an hour, like watching “Gilligan's Island”. The point of this rule is to make sure you understand the ideas behind your solution and are not just copying in a rote fashion.

Exams

Exam schedules and rules are in Resources > Exam Info > exam-rules-schedule.html . You are expected to know and to abide by these rules.

Poor grades

If you get less than 70% on an exam please contact the 314 TAs or instructor *immediately*. Often students wait until after the final, and then ask how they can raise their grade. At that point there is no way to raise their grade. The earlier you contact us the more chance we can be of help.

Topics

The following list is organized by topic, not necessarily by chronological order of coverage in the course.

1. Formal Languages
    - Context free grammars
    - Regular grammars
    - Finite state automata
    - Regular expressions
    - Using all of the above to define a language
2. Functional Programming
    - Scheme
    - Repetition through recursion, not iteration
    - Stateless programming
    - Closures
3. Logic Programming
    - Prolog
4. Scripting
    - Python
5. Principles
    - Types and type inference
    - Scope & binding,lexical and dynamic
    - Parameter passing modes