Assignment 1 – CS 4536/536: Programming Language Design

Instructor: Rose Bohrer (pronouns: she/her)

Instructor Office: Fuller Labs 139

Due Date: See Canvas (I use the same handout in both courses, with different dates)

Overview

The primary goal of this assignment is to install and gain proficiency with Rust, the programming language that will be used in this course.

How Long Will This Take?

My goal is that most students will spend at most 13 hours/week on this course and that few will spend more than 15 hours/week, in total.

If you have spent more than eight hours on this assignment and are not done, you are **expected** to reach out to the course staff for help (office hours, chat, or email). It is our job to help you and your job to seek help when appropriate. The same goes if you have been stuck for more than half an hour on the same problem without feeling like you've made new progress.

Grading

This assignment will also get you used to the basic structure of assignments in this course. Most of the assignments are divided into a programming part, which is autograded, and a written part, which is evaluated by your peers. The written problems are graded by effort: full credit is given for each problem where an honest effort is made. However, students are expected to take all problems seriously and to keep in mind that your fellow students will read your work: write the kind of answers you would want to read.

This assignment is worth 100 points. 75 points are for the programming part and 25 points for the written part. Your grade for the programming part is the sum of scores for each passing test case. Your grade for the written part is 25 * the fraction of problems for which you submit an honest attempt at a solution.

What to Read While Working on This Assignment

In addition to the textbook readings about Rust, you should refer to the official Rust learning resources as-needed: https://www.rust-lang.org/learn

Setup

If you do not have VSCode installed, install it from https://code.visualstudio.com/
Follow these instructions to install **both** Rust and the rust-visualizer extension for VSCode, which adds support for Rust.

https://code.visualstudio.com/docs/languages/rust

Programming Assignment

The code you download will contain several files:

- asgn1.rs is the only file you edit and submit
- main.rs is the file you run if you want to debug locally. It is what the autograder runs. You can look at this to understand the test cases, to better understand the assignment and your solution
- gradelib.rs is just helper functions used in grading

Download and extract the programming assignment from Canvas. Your assignment is to implement all the unimplemented functions to meet their specifications as given in the comments.

The point breakdown is as follows:

- 5pts – Hello World
- 15pts – Array Basics
- 30pts – Loops
- 25pts – Recursion

Written Assignment

Your grade for this assignment will be assigned based on completion. In the next assignment, you will peer-review your classmates' answers to these questions and they will review yours. You are expected to write answers of the same quality you would want to receive.

10pts In 2-4 sentences, summarize your personal learning objectives for this course. Of the objectives listed in the syllabus, which are most important to you? Are there any topics you'd like to learn about which are not on the syllabus? If so, which? (If so, you are encouraged to reach out to the instructor for supplemental material)

10pts In 2-4 sentences, summarize your prior experiences of programming languages by comparing yourself to the different personas discussed in the first lecture.

5pts What are your favorite and least favorite programming languages? Share your favorite and least favorite experiences using those languages.

(Bonus) 5pts Students are encouraged to keep a personal glossary of key vocabulary terms that are introduced during the lectures. This should be a simple, typed document with a bullet list of each key term and its definition. For extra credit, you can submit this glossary with each week's assignment.

Preview of Peer Assessment

In future assignments, you will assess each other's written work based on a loose rubric. The feedback is meant to be open-ended; the rubric tells you topics that should be addressed,e.g:

- Completeness and thoroughness
- Self-direction: For open-ended problems, did they bring in their own ideas and follow through?
- Expression: Do they express themselves clearly, concisely in writing?
- Creativity: Did they explore new ideas?

Instructions will be provided for writing feedback in a constructive fashion.

Submission

You make two submissions. You submit a single Rust file asgn1.rs under the programming assignment and a single PDF file under the written assignment.