Assignment 2 – CS 4536/536: Programming Language Design   
Instructor: Rose Bohrer (pronouns: she/her)  
Instructor Office: Fuller Labs 139

# Overview

The primary goal of the programming assignment is to develop a parser for a simple programming language, i.e., a program that recognizes the syntax of a simple language.

## 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.

## Grading

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 20 \* the fraction of problems for which you submit an honest attempt at a solution plus 5 \* the fraction of assigned peer reviews you complete.

## What to Read While Working on This Assignment

In addition to the lecture readings about Rust, you should refer to the official documentation for the Rust peg crate. You are welcome to refer to other resources about parsing expression grammars.

## Programming Assignment

Download and extract the programming assignment from Canvas. Your assignment is to implement all functions (and parser rules) to meet their specifications as given in the comments. Point breakdown:

* 10pts – identifiers (e.g. variable names)
* 10pts – numerals (i.e., numeric literals)
* 35pts – other expressions
* 20pts – declarations

## Written Assignment

Over the next few written assignments, you will prepare (and then perform, in-class!) an experiment (user study) to answer a question that you personally have about programming languages

20pt Submit a pitch for an experiment. This typed document should contain all of the following:

* A high-level question that you are interested in answering about programming languages (in general, or a particular language)
* Describe why you are interested in the problem
* A break-down of the high-level question using the School-of-Thought personas. List all personas that can help answer the question, and identify which parts they would help with
* You will act as a Social Scientist to answer the question in a future assignment, so make sure the Social Scientist can contribute significantly to the answer. Make sure to include at least one falsifiable, testable hypothesis that could be confirmed or denied by experimental user tests
* Describe how you would imagine testing that hypothesis, given a window of at most 10 minutes of testing per user (due to time constraints we will have for in-class testing)
* There is not a specific length requirement, but we expect most responses to be about 1 page. What is important is that you satisfactorily answer all the questions. Remember that you will actually carry out a study based on your question in class with your fellow students, and that giving serious thought now will save you effort in the long run. Your pitch will be peer-reviewed in the next assignment and you should write for your classmates as your audience.

## Peer Reviews

Complete all peer reviews that were assigned to you in Canvas, which are reviews of your classmates’ written assignments from the previous assignment. You will receive credit for all reviews you complete which meet the criteria given here. Please note that you are required to maintain a respectful and professional tone in reviews, and that your reviews are non-anonymous in order to encourage professional interactions. In cases of extreme violations of professionalism, the instructor retains the right to revoke credit. Advice on professional tone is provided in these instructions.

Each review should follow this format. First, start by assigning “badges” to the work you are reviewing. Badges are a fixed set of judgements about the work, i.e., assign a set of words/terms from the following list (you may assign badges to individual questions or the assignment as a whole. **If you would like to use a badge that is not in this list, ping the course staff on Slack**:

* “Incomplete” The work is not complete enough to provide quality feedback
* “Abstract” The work is detailed enough to provide feedback, but still relatively high level
* “Self-Directed” The work shows initiative in setting one’s own direction. An example would be setting a learning objective outside the ones proposed in the syllabus
* “Concise” The work makes efficient use of space to express itself.
* “Thorough” The work makes sure to cover every perspective and provide necessary detail
* “Experienced” Your classmate shows substantial preparation for the course
* “Enthusiastic” Your classmate shows excitement for the topic
* “Growth-Minded” Your classmate shows openness to learning topics they don’t yet know
* “Growth Mindset Encouraged” You encourage your classmate to give themselves a chance learning topics they don’t yet know

You should aim to assign at least 2 badges in each review, except when the submission is incomplete. Write at least one sentence elaborating on at least one of the badges. Your elaboration should follow these rules for professional tone:

* Criticism is welcome, but it should not be personal, it should be focused on the work (“the response to question 1 is missing” vs. “you forgot to answer question 1”)
* Use “I” language when appropriate (“I didn’t understand X” vs. “X makes no sense”)
* Avoid superlative words or exaggerating language “very confusing”
* Be specific (“a plan for interpreting user data was not provided,” “the proposal is vague”)
* Do not imply a person should give up. Reviews should be consistent with growth mindset.