Project CP – CS 536: Programming Language Design
Instructor: Rose Bohrer (pronouns: she/her)
Time: T,Th | 4:00 PM – 5:20 PM
Classroom: Higgins Labs 114
Instructor Office: Fuller Labs 139

Overview

This course is an ungraded, project-based, team-based course. This means that you identify a team, design your own project together, and assign yourself a grade based on how well your project meets the course goals. In understanding the role of assignments, we should distinguish "structure" from "requirements." If you already know how you want to run your project or what you want to do, you should do so – be as radical as you like. You choose how to engage with this structure, but the checkpoint submissions provide you a structure so that you have lots of guidance if you like.

When I review your submissions, I **only** provide formative feedback, never a grade. You are the person who assigns your final grade, you are in control.

Goal

This week's goals are team formation, choice of language you will program in (if applicable), and installation of that programming language (if needed). You will practice reading the language I use in lecture (Scala) but it is not your primary goal yet.

Logistics

This week, you will tell us who your team is. If you already know who your team will be, you can go ahead and use that team. We spent time in class to help pick teams, but I realize plans change during the first week, so please confirm your team choice on Canvas and reach out to me if you'd like help picking a team.

You pick the team. Each team must be at least 3 people. There is no upper limit on size. It is recommended to keep the same team all semester. The checkpoints are structured based on the idea that teams meet at least once per checkpoint in some form. You decide what to do, I tell you a good default.

Checkpoints are due at the times listed in the Schedule section of the syllabus.
As a group, fill out the form on the following pages and submit it via Canvas by the deadline.

Task Instructions

- Identify a team based on people you met in class, people you find on the course Slack, or the recommendations I sent by email.
- Schedule a meeting time
- Before the meeting:
  - Decide your own preferences: do you prefer to do a theory project, a programming-heavy project, a humanities project?
  - If you are doing a programming project, what programming language do you prefer to program in?
- Meet before the deadline (can be short, can be online)
- Fill out the form at the meeting
- Read the handout for next checkpoint and make sure everyone in the group knows what work is planned.
- Create a new group on Canvas
- One person submits on Canvas, under that group

Background Survey Form

Part 1: Self-assessment

Describe your current knowledge level for each course objective. Use words, numbers, or letter grades.

Objectives:

1. Identify a clear problem where programming language design can be used

   _____

2. Communicate with clarity and technical depth about language design

   _____

3. Develop a mathematically-precise definition of your language's syntax

   _____

4. Work effectively with your team to plan your project

   _____

5. Implement a well-scoped prototype in the language of your choice

   _____

6. Situate your project among the schools of thought discussed in class

   _____

Describe your goals for this course + semester. This could be anything from "focus on #6" to "practice time management or interpersonal skills". Describe what NR, C, B, and A work would look like to you. Spend time on this – this answer is important because you will use it to decide your semester grade.

_____

_____

_____

_____

_____

The next page gives an example of the form could be filled out

EXAMPLE

Describe your current knowledge level for each course objective. Use words, numbers, or letters.

Objectives:

1. Identify a clear problem where programming language design can be used

   ___"C" I'm not really sure what this means. I know what kinds of problems programming can solve, but I think you're asking something different. But I'm a biologist so I might try and find a biology problem to solve.
2. Communicate with clarity and technical depth about language design
   ___"B"___I'm good at creative writing but don't know the specific vocabulary for this course yet
3. Develop a mathematically-precise definition of your language's syntax

   __"A"___I have a math minor and do well with definitions. I'm sure I'll pick it up fast.
4. Work effectively with your team to plan your project

   __"A"___I have lots of successful experience working on teams
5. Implement a well-scoped prototype in the language of your choice

   __"NR"__I have no programming experience.
6. Situate your project among the schools of thought discussed in class

   _"B"__I've done some reading about the social implications of CS and feel I could see why a social scientist would care, but I don't have nearly the amount of breadth I want.

Describe your goals for this course + semester. This could be anything from "focus on #6" to "practice time management or interpersonal skills". Describe what NR, C, B, and A work would look like to you.

____I want to use my strengths in team management to make up for my lack of prior programming experience. I want to team of with experienced programmers and be their test subject to help figure out how to make their language design helpful for newcomers. For #1,2,6: I want to be able to talk to other biologists about whether and how the course material could help with our work.

___"A" means 1: I can point to our project report and see how my experiences as a newbie were helpful, 2: I could talk to a biologist about how to apply the course material to our work, and 3: My teammates don't hate me

___"B" means I meet 1 or 2 of the 3 objectives, maybe I helped my teammates but didn't strengthen my communication skills_____

___"C" means I don't meet any of the objectives, but I consistently tried by showing up to group meetings and attending office hours.

___ "NR" means I did not meet any of the objectives and did not seek much support.

Part 2: Team Selection

If you have already picked a team, you can skip this part. Make sure to fill in your team information during the "in-meeting form".

1. **Some students prefer high-risk projects which could succeed impressively or fail spectacularly. Others prefer lower-risk projects where the fundamentals are well-established, enabling the team to go into greater depth. What are your preferences?**

   _____

2. **What topics do you prefer doing the project on? I give some suggestions, but your suggestions are usually more fun. The sky's the limit. You can circle suggestions and fill in your own.**
   - i. **Critical studies (part of Humanities)**
     1. **How do languages exclude disabled programmers?**
     2. **How do cultural norms of the PL research community reflect white-supremacist capitalist cisheteropatriarchy?**
     3. **What would it mean for a programming language to be trans?**
   - ii. **Social sciences**
     1. **When designing a list of key concepts + features for my language, how do I design them in a way that is easy for a new user to learn?**
     2. **If I use surveys developed for clinical psychology to assess the emotional experience of a user, how can the results inform language design?**
     3. **What role does language play in the formation of professional social hierarchies?**
   - iii. **CS, implementation-heavy**
     1. **Find a language that has only been implemented with an interpreter, then implement it as a compiler.**
     2. **Design and evaluate some compiler optimizations for a language that doesn't have them.**
     3. **Design and implement a relatively simple language, but write lots of example code with it**
   - iv. **CS, theory-heavy**
     1. **"The Linear Logic of Pronouns" (ask instructor)**
     2. **Add a type system to a language that doesn't have one**
     3. **Give a mathematical definition of what programs mean (semantics) in some language that doesn't have that**
     4. **Implement an advanced type system design from the research literature**
   - v. **CS, normal**
     1. **Language for configuration files**
     2. **Language for music**
     3. **Language for specifying video game levels**
     4. **Language for specifying security or privacy policies**

- **And so much more! Write your own ideas:**
  - i.
    _____
  - ii.
    _____
  - iii.
    _____

Only fill out the next two questions if you want course staff to find team members for you:

Is there anyone specific you would enjoy having on your team? You can list their name here. If you belong to any under-represented groups and if you're specifically interested in being matched with other students from those same groups, you can also list those groups here**:**

_____

Is there anyone that should not be on your team? Do not give me any reason why. Just list the name and I will make sure you are not matched with them.

_____

Only fill out this form if you **did** pick teammates and do not need help from course staff to pick them.

Student Structured Meeting Form

Meet with all your teammates, follow the meeting instructions, and fill out the form.

1. **Introduce yourselves to each other and to us. Include your name, WPI username (i.e., gompei if you're gompei@wpi.edu), and pronouns, if you wish.**

    We are: _____

    _____

    _____

2. **Exchange contact info and decide a meeting schedule for the term.**
3. **Discuss (and list) topics that you are interested in using for your project.**

    _____

    _____

4. **Pick one of those topics. You should plan to do your project on this topic but can change your mind during the next checkpoint.**

    _____

5. **Have everyone in your group download and install and run any necessary software.**
6. **Read these code samples, taken and revised from** Tour of Scala
    a. val result: Int = 2 + a

```
b.  def main(args: Array[String]): Unit = {
       println("Congratulations! You have printed a string") }
c.  case class Person(firstName: String, lastName: String)
    val result: Person = Person("Leo", "Lion")
d.  val n : Int = someFunction()
    n match {
      case 0 => 0
      case n if n < 0 => -n
      case n if n > 0 => n
    }
```

Write down what you think each code snippet means. Discuss that with each other and write notes of the discussion here. You can skim the tutorial to help answer some of your questions, but you will likely still have some questions:

_____

_____

7. **I want to support you. A great way to do this is if you proactively ask questions and share comments/concerns with us on each checkin, at least once per checkin, so I can give you answers in our feedback. Put questions/comments here.**

_____

_____

8. **What do you (each) plan to do on your project in the next week?**

_____

_____

END OF FORM

Rubric for Feedback from Instructor

I will use this rubric to write feedback. I will not write grades. It is important to read the feedback and discuss it with your teammates.

1. What questions and concerns can I answer for the team?
2. How did the teammate search go? If you're still looking, just let me know in your submission and I'll send some email to help out.
3. Did the students show creativity in brainstorming project topics? Are you having fun?
4. How can I help you identify a realistic scope for your project?
5. How can I help you finalize your choice of topic and programming language?
6. Any challenges installing picking or installing software?
7. Are all project groups entered correctly on Canvas?
8. How challenging are students finding Scala syntax? It is expected to struggle with it in the beginning, because I am only starting to teach syntax. But it is important to keep learning the syntax as we go. How can I help with that?
9. Did the students say which each student specifically will do during the next week (not just the group as a whole?) Is their plan is concrete enough that they can work together effectively? Anything I can do to help the team dynamic?