195R Missive

Spring 2014

Contents

1 Description of the course

The purpose of 195R is to explore intermediate representations and the optimizations that these representations enable. The course will also examine language runtimes. The class will take a look at these topics through weekly discussions and readings, and by implementing a compiler for a full-blown language over the course of the semester.

2 Class Format

195R will meet every Monday, Tuesday and Wednesday (room TBD). Each meeting will be of the following format:

- Monday (8:30-9:30PM): Lesson and discussion lead by that week's leader
- Wednesday (5:30-6:30PM): Lab to work on the week's current work. Open discussion is encouraged.
- Friday (4-5PM): Lab to work on the week's current work. Open discussion is encouraged.

3 Collaboration

In 195R, you may talk freely about ideas with others when working on solo projects. There is a lot of material and some of it can be potentially confusing or hard to grasp at first, so it can help

to have it explained by another person. You cannot share code or algorithmic details with one another, however. During partner projects, you may only communicate with your partner.

4 Grading

There will be four main projects that all build up to the finished Pyret compiler. These projects (and the grade distribution) are:

- ANF (20%)
- Basic Compiler (Part 1) (30%)
- Basic Compiler (Part 2) (30%)
- Garbage Collector and Polish (20%)

The Basic Compiler projects will be done with the same partner, while ANF and the Garbage collector projects will be done alone.

5 Schedule

5.1 Calendar

Week	Lesson	In	Out	Current Work
1	Review Pyret's language features and behaviour (i.e., numbers)		ANF	Start ANF
2	CPS and ANF as an intermediate representation			Produce ANF
3	Introduction to SSA	ANF	Part 1	Start basic compiler, pt. 1 (stdout and #'s)
4	Introduction to LLVM			Continue basic compiler, pt. 1 (arithmetic)
5	Continue discussing LLVM	Part 1	Part 2	Continue basic compiler, pt. 2 (functions, objects)
6	Runtime discussion			Continue basic compiler, pt.2
7	Further runtime discussion			Continue basic compiler, pt.2
8	Further runtime discussion			Finish up basic compiler, pt.2
9	Garbage Collection	Part 2	GC	Start working on Garbage Collector and optimizations.
10	CPS/ANF Optimizations			Continue GC and optimizations.
11	SSA Optimizations			Continue GC and optimizations.
12	Discuss other methodologies and decisions (i.e., JIT)			Continue GC and optimizations.
13	Discuss compiling concurrent code.			Polish compiler.
14	Hot wash-up and review of the semester.			Finish up compiler.
15	Success! Reading period starts.	GC		Done!

5.2 Weekly Reading

These are the readings from the course's textbook "Compiling with Continuations" and papers that are relevant to the corresponding week's discussions.

1. None
2. Chapters 2-5 and "The essence of compiling with continuations", by Flanagan, Sabry, Duba and Felleisen, and the 1-page followup
3. "A correspondence between continuation passing style and static single assignment form", by Kelsey

4. Chapters 7-10, 14 and "Lambda, the ultimate label or a simple optimizing compiler for Scheme", by Clinger and Hansen
5. TBD
6. Chapters 12 and 16, parts of 17-18
7. Chapter 16
8. TBD
9. TBD
10. Chapter 6-11
11. Selections from "Efficiently computing static single assignment form and the control dependence graph", by Cytron, Ferrante, Rosen, Wegman and Zadeck
12. TBD
13. Chapter 17

Please let us know if you find any mistakes, inconsistencies, or confusing language in this or any other CS195R document by filling out the anonymous feedback form:

`http://cs.brown.edu/courses/cs195r/feedback`.