

CS162: Programming Languages

Winter 2014

Instructor: Kyle Dewey (kyledewey@cs.ucsb.edu)

Teaching Assistants:

- Lawton Nichols (lawtonnichols@cs.ucsb.edu)
- Jared Roesch (jroesch@cs.ucsb.edu)

Website: <http://cs.ucsb.edu/~kyledewey/cs162>

Piazza Website: piazza.com/ucsb/winter2015/cs162

Course Description:

Various topics in programming languages, along with their connection to formal logic. Special emphasis is placed on the following topics:

- First-order logic
- Type systems and typechecking
- Functional programming
- Logic programming
- Constraint logic programming

Prerequisites:

- CS130A
- CS138

Grading:

Based on 7 assignments, each worth 1/7th of the course grade (approximately 14.3% apiece)

Assignments: (subject to change)

- Introduction to Scala (1.5 weeks)
- Functional Images (1.5 weeks)
- Implementing a typechecker for a simply-typed language (1.5 weeks)
- Implementing a typechecker for a polymorphically-typed language (1 week)
- Introduction to Prolog (1.5 weeks)
- Implementing a subset of Prolog (2 weeks)
- Implementing a subset of a constraint logic programming language (2 weeks)

Assignment Late Policy:

Each person has 3 "slip days" that they can use over the course of the quarter to delay turnin of an assignment. Slip days can only be used in increments of 24 hours—30 minutes late is the same as 23 hours late. Slip days will be applied automatically by the TA based on when an assignment is turned in.

Communication Policy:

For anything that is not personal (e.g., grading), please use Piazza as opposed to email. This allows the three of us to respond independently to the same question, allowing for better response times. Additionally, if questions are posted publicly, it gives other students the opportunity to respond, meaning even better average response times.

With email sent to me (Kyle), assume that I will take at least 24 hours to respond. Typically my response time is much, much faster than this, but I do occasionally take this long. Historically, this has only been an issue the last hours before a project deadline, and only for students who started far too late. The point being: start early!

Academic Honesty:

Each person is responsible for their own work, and is expected to complete the assignment without collaborating with anyone else. High-level discussion of the concepts without anything specific to an assignment is OK; talking specifically about the solution to an assignment or sharing code is not OK. Violation of this policy can lead to an F on the assignment or, in extreme cases, an F for the entire course. When in doubt, refer questions to the professor or TA.

Seriously, Do Not Take Code, and Protect your Code from Being Taken!

We have an automated mechanism which can detect similarities between code. If your code ends up being similar to someone else's code, the mechanism will inform us, at which point we will manually inspect both sources side-by-side. While it is possible for two codebases to be similar by chance, we have historically found that patterns are far from random. The same groups often come up repeatedly as being similar, even when plagiarism has not occurred. That is, we can detect even when people have discussed ideas with each other - we just do not pursue it in these cases.

On the other side of the coin, you must also make sure to **protect your code from being stolen**. On CSIL, the default permissions are such that everyone can view everyone else's code. To guard against this, do:

```
chmod 700 cs162
```

...where `cs162` contains your class code (or whatever directory you use). Additionally, if you use version control, make sure your code is not publicly accessible (as with public repositories on GitHub). On GitHub, you can get private repositories for free by filling out https://education.github.com/discount_requests/new, and Bitbucket offers unlimited private repositories. Even if you obfuscate project names or filenames, it is trivial to perform a code search for code we have supplied to find it.

Historically, code has been taken both from CSIL and from public repositories multiple times. In general, we cannot figure out who took what from whom, and both parties can end up being penalized.