210: Compiler Design

Fall 2015

Assignment 0

Due: (Sept 23, 2015, 10 a.m.)

0 Points

1 Introduction

The assignments for this class will require you to construct a simple compiler for a simple (yet non-trivial) language. The task for this assignment is to explore your target machine. (Assignment 1 will ask you to produce part of a code generator for your target machine.)

The compiler's target is the IA32 architecture, which is found on most PCs and modern Macintosh machines. Because we do not cover advanced optimizations in this course, you will not really be able to make good use of recent extensions and we therefore ask you you stick with the traditional IA32 instruction set. That is, please do *not* use the 64-bit instructions. Note that the course Advanced Compiler Design, scheduled for next semester, may provide opportunities to explore such features.

In the framework used in this class we officially support Linux. A number of students have successfully used the framework with Mac OS X or Windows, (Windows typically with Cygwin installed), and you are welcome to do so as well. However, as we do not have access to Windows machines for grading, *you must test with Linux.* In general, only small changes are required to support multiple platforms: many of the required changes have already been encapsulated in the class Config included in the framework.

2 Issues

Your compiler (to be constructed in subsequent assignments) must generate code for the target machine. So you must be able to determine the instructions that are generated and you must be able to read your compiler's output. Also, to compile interesting programs, these programs must contain input (and output) statements. At the very least, you must understand how to read an integer value from standard input (or a file, depending on the operating system). To allow checking of computation, output is necessary as well. For input and output, it is perfectly acceptable to use existing libraries (e.g. `libc`) – one of our goals is to explore software reuse, and you are encouraged to build on existing OS services of your target platform.

3 Your task

You will work in a team of two students. Submit the names of the members of your microteam using the link provided on the course web page.

To demonstrate that you can use your target machine, write a simple assembly program that reads an integer from standard input, adds one to it, and prints the output to standard output.

4 Hand-in

The solution is to be submitted electronically. To submit your solution to this and later assignments, you first have to register your microteam (see class home page at `http://www.lst.inf.ethz.ch/teaching/lectures/hs15/210`). *After* the due date, we will set up SVN repositories for you.

As you can see, this assignment carries no points and we will do no grading. Still, we ask you to check in your solution into a directory `a0` of your SVN repository. The purpose being to ensure you can submit a solution for future assignments and to confirm that you know enough about your target system to get started.