MC28 Lab 1: SLIM assembly language programming

Max Hailperin Karl Knight

Due: Wednesday, September 20, 1995, in class

Introduction

For this lab, you will do some SLIM assembly language programming exercises from chapter 11 of our book. For certain of these exercises, you are asked to gather statistics from the SLIM simulator in order to determine the relative efficiency of various versions of the same problem. You will be working in pairs on this assignment, so please decide who you will be working with before coming to lab the first day.

Prelab

You may well find the first lab session on Thursday frustrating, since you will be doing a type of programming that is wildly different from what you did last semester. Also, Max will probably not get to the section on assembly language until Friday. Nevertheless, please persevere, and use the lab as an opportunity to begin to understand assembly language and the SLIM simulator.

Before coming to lab on Thursday, you should read sections 11.1 and 11.2, at least through exercise 11.2 on page 355, and hopefully to the end. These sections will be discussed in class on Friday.

In Lab

To load the simulator, simply evaluate the expression

```
(load "/mcs-server/Public/ConcAbs/slim-simulator")
```

Having done this, you can run it as described in section 11.2. Try out the simulator by running the programs starting on page 354 of the text. (Recall that you can access the code from this chapter by opening the file `/mcs-server/Public/ConcAbs/chapt11.scm`.) Remember that in order to run a program, you must first `assemble` it, then `load-and-run` it.

Once you feel comfortable with the workings of the simulator, you should go on to the following problems, which comprise your assignment:

1. [§11.2] Exercise 11.2 on page 355.
2. [§11.3] Exercise 11.4 on page 362.
   Note: the page cited in this problem should be 72, not 78, in your blue textbook.
3. [§11.3] Exercise 11.5 on page 362.
4. [§11.3] Exercise 11.9 on page 368.
5. [§11.4] Exercise 11.10 on page 373.
   Note: exercise 2.1 cited in this problem is actually on pages 37 and 40 of your blue textbook.

Postlab

Each group should write a single lab report, authored by both members of the group. As usual, write up a lab report that explains all of what you did to an audience generally knowledgeable about Scheme (and also SLIM for this lab), but ignorant of specifically what you did.

You are asked to compare the efficiency of the three different assembly language versions of `power` in exercises 11.4, 11.5, and 11.10. You should do this by comparing how many instructions the three versions execute for various exponents. For example, you might test each version with exponents 2, 4, 8, 16, and 32, and report your results in a table. Or you could graph the results.

Note that the Scheme code from which the three versions are written are linearly iterative (11.4), logarithmically iterative (11.5), and linearly recursive (11.10). Do your results match these expectations? In the case of the linear versions, you might expect the number of instructions to be a linear function of the exponent n (that is, expressible as $An + B$ for some values of A and B). Can you find such A and B for both of the iterative versions? How do the values of A and B compare for the two versions? How could you check that exercise 11.5 has logarithmic growth?