

Testing CSc 252 Programs

This document describes how to use the testing script along with the provided testcases with your programs for CSc 252. The technique will be the same for each programming assignment this semester. This document refers to **prog1**. To use this procedure with other programs substitute the program name for **prog1**; i.e., use **prog2** in place of **prog1** for the second programming assignment.

This technique requires that you be able to login to [lectura](#) using PuTTY (from Windows) or the command-line ssh (from Linux or Mac OS X). This technique is described in the “[Getting your MIPS program to lectura](#)” handout.

Step 1:

Copy your **prog1.s** to lectura using one of the techniques describe in “Getting your MIPS program to lectura.”

In what follows, I will assume you have **prog1.s** in a directory named **prog1** that is inside a directory named **252** on lectura. If you used a different directory name instead of **252** or instead of **prog1** then make the appropriate substitutions below.

Step 2:

Login to lectura using the PuTTY (from Windows) or the command-line ssh (from Linux or Mac OS X).

Change directory to your 252 directory:

```
cd 252/prog1
```

Step 3:

Copy the provided test cases, test case solutions, and the test script to your directory:

```
cp ~cs252/spring08/prog1/* .
```

That is tilde character in front of cs252, and a dot, or period, at the end of the command. There is a blank space in front of the dot.

Step 4:

Execute the test script:

```
perl 252test.pl
```

If your program passes all of the test cases, you will see the following result:

```
Test 1 passed!  
Test 2 passed!  
Test 3 passed!  
Test 4 passed!  
Test 5 passed!  
Test 6 passed!  
Test 7 passed!  
Test 8 passed!
```

Test 9 passed!

Test 10 passed!

If you fail one (or more) test cases, the failed test case will be reported. You will also see a printout of the output from your program, followed by the correct output. Here is an example from program #1:

Test 1 passed!

Test 2 passed!

test 3 failed!

-----Your output-----

The integers are

17 -50 3 -23 -60

The integers without any 0's are

17 -50 3 -23 -60

-----Correct output-----

The integers are

17 -50 3 -23 -60

The integers without any 0's are

17 -50 3 -23 -60

etc.

From looking at the two outputs, we can see that there is a blank line present in the output that should not be there.

Step 5:

This step is optional. The testing process will create a number of additional files in your directory. You can just leave the files there. The filenames have the following form:

test1.s-user Your program combined with the test data for test case #1

test1.s-user.out The output of your program for test case #1

There will be a pair of these files for each test case. If you wish to remove these files, use the command:

rm *user*

There is a blank space after **rm**.