

# CMPT350: Lab Demo 2

---

*Writing a MIPS Program with string input/output*

1%

Due: September 21, 16:50, End of Lab period.

This lab demo will ease you into writing your own MIPS programs with some simple string input, output, and manipulation. Your program will prompt the user to enter a string (such as their name), and modifies it a few different ways, including capitalizing and lower-casing the word

## Laboratory Procedure:

On the course Moodle, look for the Lab Demo 2 assignment in the Lab Demonstration section. Inside is a premade skeleton MIPS files, `demo2.s`. Save this somewhere handy on your machine. You may use both the command line version, `spim`, and the GUI version, `QtSpim` to run your program. As well, you will need a text editor to edit your `demo2.s`. Gedit comes preinstalled on the lab machines and is a suitable editor. You can start `QtSpim` by pressing Super (Windows key) and searching for it, or by typing `qtspim` in the command line.

## Assignment Requirements:

- **Basics**
  - Run your program successfully using either `QtSpim` or the `spim` command line tool.
  - Sufficient commenting is present in the program, including the preamble comment block at the beginning of the file. **This preamble block must contain a line which contains the contents `"# Author: <YourName>"`**
  - 
  - The program terminates properly.
  - Output looks "proper" (e.g. appropriate newlines, no newlines after the name)
- **String I/O**
  - When run, the program prompts the user for their name.
  - The program greets them, and describes what the program does (e.g., Hello Quinn, I will modify your name).
  - The program modifies the given input to have all capital letters.
  - The program modifies the input to have all lower case letters.

**\*\*\* NOTE:** Your code **must be interpreted successfully and execute** within the **laboratory environment, using the command line version of SPIM**. Failure to do so will result in a **mark of ZERO** for the program. **\*\*\***

### Creating a trace file

As part of all demo and laboratory submissions, you must submit what's called a trace file. This is a text file you'll create that in essence replicates what your terminal looks like when you demonstrate your programs. To assist with this, a tool called `script2` has been provided for you. To create a trace file, simply run `script2` in your terminal, and then proceed to use the terminal as normal to navigate to and run your programs. When you have demonstrated everything, press CTRL+D to signal that you are done, and `script2` will exit, creating a file called `typescript` in the directory you ran `script2` from. This is your trace file. **Please rename your tracefile to include your name, e.g. `baker_typescript`**

### Submission

When you feel your program meets all of the above requirements, call the lab instructor over and they will review your program with you. Please wait until you are confident in your program's results. (Feel free to ask for clarification on anything at any time though!)

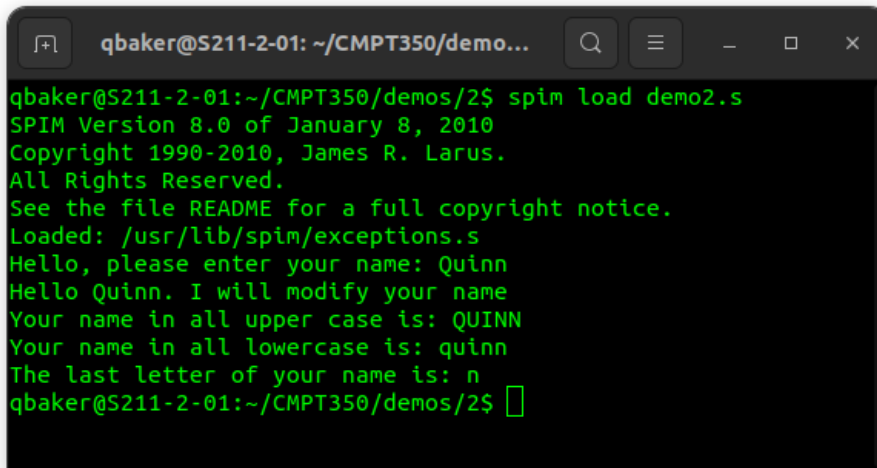
Your program should be named using your name and end in a `.s` extension. For example, `BakerDemo2.s`

When you feel your program meets all of the above requirements,

After demonstrating your program to the lab coordinator, create a zip archive with your code submission and trace file included. An easy way to do this is with the `zip` command. For example, if your solution file and trace file are in a folder called `baker`, then the command `zip -r baker.zip baker/` will create a zip file containing your submission files called `baker.zip` (Replace this with your name when creating your zip file). Upload this zip file to Moodle.

## Example Output:

This is one example of what your program should look similar to.



```
qbaker@S211-2-01: ~/CMPT350/demo...
qbaker@S211-2-01:~/CMPT350/demos/2$ spim load demo2.s
SPIM Version 8.0 of January 8, 2010
Copyright 1990-2010, James R. Larus.
All Rights Reserved.
See the file README for a full copyright notice.
Loaded: /usr/lib/spim/exceptions.s
Hello, please enter your name: Quinn
Hello Quinn. I will modify your name
Your name in all upper case is: QUINN
Your name in all lowercase is: quinn
The last letter of your name is: n
qbaker@S211-2-01:~/CMPT350/demos/2$
```

h

## Hints:

- Remember your ASCII table for converting numbers to characters
- Likewise, remember the gap between lowercase and uppercase characters (A=65, a=97, B=66, b=98, ...)
- Variable offset memory access is not valid syntax in MIPS, e.g.

```
la $s0, message
li $t0, 1
lbu $t1, $t0($s0)
```

will not work. However, modifying the register holding the memory address will, e.g.

```
la $s0, message
addi $s0, $s0, 1
lbu $t1, 0($s0)
```