Systems I
Lab 7
Procedures in assembly

1 The assembler/simulator

Once again, we will use the *Venus Simulator*, which you can access in your web browser at:

`https://venus.cs61c.org`

Also again, this reference of RISC-V instructions may be useful:

`https://marks.page/riscv/`

2 Counting instances of a value in an array

We will once again work with arrays. Specifically, you will **write this procedure**:

```
int count (array_base, length, value)
```

The job of the `count()` procedure is to search the array that begins at the address `array_base` and contains `length` elements, counting the number of instances of the given `value` that appear in the array. The return value is that count.

Open the *Venus Simulator*, and then start a code file:

```
[user@venus] /# edit count.s
```

Open the following starter code, and then copy-and-paste what is there into the Venus Simulator's editor:

`https://bit.ly/COSC-175-2526f-count`

This code contains a call to the `count()` procedure, passing it the array defined in the `.data` segment, and searching for the value `102`. (You could change the value to seek to another value if you like.) The result of the call is then printed to the console before the program then exits.

Remember to download your code: When you have written code within the `count.s` file that is worth keeping, go back to the `Venus` tab terminal and download it:

```
[user@venus] /# download count.s
```

3 Recursion!

Go back the *Venus Simulator*, and return to the `Venus` tab and its terminal. Open a new file in the editor:

```
[user@venus] /# edit fib.s
```

`https://bit.ly/COSC-175-2526f-fib`

Copy and paste this source code into this Venus Simulator's editor, then click `Save`.

This starter code contains a call to `fib()`, passing it whatever value is given at the label `n:` in the `.data` segment. You must **write this procedure**…

```
int fib (n)
```

…which must calculate and return the n^{th} Fiboanacci number, defined as:

$$F_n = \begin{cases} n & \text{if } 0 \leq n \leq 1 \\ F_{n-1} + F_{n-2} & \text{if } n \geq 2 \end{cases}$$

Write the `fib()` procedure. You **must not** write it *iteratively*—that is, you cannot compute the answer by using a loop. Instead, you **must** write it *recursively*—that is, to obtain F_{n-1} and F_{n-2}, you must perform procedure calls to `fib()`.

Download your code for `fib.s`.

4 How to submit your work

Copy your completed `count.s` and `fib.s` to the `lab-7` folder within your Google Drive folder.

This assignment is due on Nov-07, 11:59 pm.