CSSE 132 – Introduction to Systems Programming
Rose-Hulman Institute of Technology

Quiz 5 Assembly II

Name (Print):_____ RHIT Username:_____

1. Without any branch (jump) instruction, how will instructions/code be executed?

   The special register [ ] *will keep fetching* [ ] *to run. Namely, without branching, the code will* [ ].

2. What do the following ARM instructions do? (Describe each)

```
  b .LABEL
```

   [ ] *to the line/instruction tagged/labeled as* [ ]

```
cmp w0, w1
ble .LABEL
```

   Compare `w0` *and* `w1`*, branch to the line/instruction labeled as* `.LABEL` **IF** [ ]

3. Translate the following code to corresponding pseudo code.

```
  mov w3, 10
.LOOP:
  cmp w3, 0
  ble .HERE
  sub w3, w3, 1
  b .LOOP
.HERE:
```

4. Translate the following code to corresponding pseudo code.

```
  mov w2, 0
  mov w3, 10
.LOOP:
  cmp w3, 0
  ble .ENDLOOP
  add w2, w2, w3
  sub w3, w3, 1
  b .LOOP
.ENDLOOP:
```

5. Walk through the code, and identify the final values of `w1` and `w0`.

```
  mov w1, 0
  mov w3, 0
  b .L2
.L3:
  add w1, w1, w3
  add w3, w3, 1
.L2:
  cmp w1, 9
  ble .L3
  mov w0, w3
```

6. Assume `x0` is the address to an `int` array with 2 entries in total. Each `int` is 4 bytes. Complete the Assembly code below to swap the values of the two entries. If the array was `[a, b]` before, after running the code the array will be `[b, a]`.

```
  ____ w1, ______ // read 1st entry from the memory to w1

  ____ w2, ______ // read 2nd entry from the memory to w2

  ____ w1, ______ // write w1 to the 2nd entry

  ____ w2, ______ // write w2 to the 1st entry
```

7. Assume `x0` is the address to an `int` array with 10 items in total. Each `int` is 4 bytes. Complete the Assembly code below to count the number of positive (i.e., greater than 0) `int`s in this array. After finishing counting, save the final count to `w0`.

```
  mov w1, 0 // initialize the entry counter
  mov w2, 0 // initialize the positive number counter
.LOOP:
  cmp w1, 10
  ___ .ENDLOOP
  ldr ___, ____ // read an int in the array
  cmp w3, 0
  ___ .NOT_POSITIVE // branch if not positive
  add ____, ____, 1 // increment the positive counter
.NOT_POSITIVE:
  add w1, ____, ____ // update the entry counter
  add ____, ____, ____ // update the address to read data
  b .LOOP
.ENDLOOP:
  mov w0, ____ // save the positive counter to w0
```

8. **Reflection:** What is one thing from this topic that you found interesting, OR what is one question you are now curious to learn more about?