CS-UY 2214 — Recitation 5

Introduction

Complete the following exercises. Unless otherwise specified, put your answers in a plain text file named `recitation5.txt`. Number your solution to each question. When you finish, submit your file on Gradescope. Then, in order to receive credit, you must ask your TA to check your work. Your work should be completed and checked during the recitation session.

You may consult the E20 manual, which is available on Brightspace.

Problems

1. Briefly describe two ways in which E20 differs from E15, and two ways that they are similar.

2. For each of the following E20 assembly instructions, represent the instruction as a 16-bit binary number. Write all 16 bits. State in prose what the instruction does, and specifically state what register or registers it modifies (including the general-purpose registers `$1` through `$7` and the program counter), and what memory cells it modifies.

   (a) `slt $4, $1, $2`

   (b) `j foobar`
   For the purpose of this question, assume that a label `foobar` has been declared on the instruction at address 15_{10}.

   (c) `movi $1, 1`

   (d) `sw $0, 0($1)`

3. For each of the following 16-bit numbers, interpret the number as an E20 instruction. Write the corresponding assembly language instruction, including its opcode and arguments, and state in prose what it does, and specifically state what register or registers it modifies (including the general-purpose registers `$1` through `$7` and the program counter), and what memory cells it modifies.

   (a) `000 000 000 000 0000`

   (b) `110 001 000 1111110`

   (c) `100 010 101 0011111`

4. Write E20 assembly instructions as described. When directed to write "a pair" of instructions, you should write two instructions that, when executed in sequence, will have the described effect.

   (a) Write an E20 assembly instruction that will store the value 34 into register `$5`.

   (b) Write an E20 assembly instruction that will add the values of register `$1` and register `$2`, and store the sum into register `$3`.

   (c) Write an E20 assembly instruction that will calculate twice the value of register `$5`, and store the value into register `$6`.

(d) Write an E20 assembly instruction that will set the value of register `$5` to 1 when the value of `$4` is less than 10, and to 0 otherwise.

(e) Write a pair of E20 assembly instructions that will jump to the address identified by the sum of 50 and the value of register `$1`.

(f) Write a pair of E20 assembly instructions that will jump to the address identified by label `destination` when the value of register `$1` is less than or equal to the value of register `$2`; when that condition is not met, execution will continue to the subsequent instruction.

(g) Write an E20 assembly instruction that will read the value from memory cell 9, and store it into register `$1`.

(h) Write an E20 assembly instruction that will read the value from the memory cell identified by label `hyoid`, and store it into register `$2`.

(i) Write an E20 assembly instruction that will read the value from the memory cell identified by the value of register `$4`, and store it into register `$3`.

5. Consider the following E20 assembly language program.

```
        add $1, $0, $0
        add $4, $0, $0
        lw $3, data($0)
    Loop:
        slti $1, $3, 20
        jeq $1, $0, skip
        add $4, $4, $3
        addi $3, $3, 1
        jeq $0, $0, Loop
    skip:
        halt
    data:
        .fill 16
```

What will be the final value of the registers `$3` and `$4`?