2021 Computer Architecture Problem Set #2

※ If there is no specified data type, you can just assume that 1 variable or 1 array element size is 8-byte.

1. Translate the following RISC-V code to C. Assume that the variables f is assigned to registers x5. Assume that the base address of the array A is in register x10.

```
addi x30, x10, 8
addi x31, x10, 0
sd x31, 0(x30)
ld x30, 0(x30)
add x5,x30, x31
```

2. For the RISC-V assembly instructions below, what is the corresponding C statement(s)? Assume that the variables f is assigned to register x5. Assume that the base address of the arrays A and B are in registers x10 and x11, respectively.

```
slli x30, x5, 3
add x30, x10, x30
slli x31, x6, 3
add x31, x11, x31
ld x5, 0(x30)
addi x12, x30, 8
ld x30, 0(x12)
add x30, x30, x5
sd x30, 0(x31)
```

3. Consider the following RISC-V loop:

```
LOOP:   beq x6, x0, DONE
        addi x6, x6, -1
        addi x5, x5, 2
        jal x0, LOOP
DONE:
```

Assume that the register x6 is initialized to the value 10. What is the final value in register x5 assuming the x5 is initially zero?

4. Translate the following loop into C code. Assume that the C-level integer i is held in register x5, x6 holds the C-level integer called result, and x10 holds the base address of the integer MemArray.

```
        addi x6, x0, 0
        addi x29, x0, 100
LOOP:   ld x7, 0(x10)
        add x5, x5, x7
        addi x10, x10, 8
```

```
addi x6, x6, 1
blt x6, x29, LOOP
```

5. Translate the following C code to RISC-V assembly codes. Assume that the variables i and j are assigned to registers x5 and x6, respectively. Assume that the base address of the arrays A and B are in registers x10 and x11, respectively. Assume that the elements of the arrays A and B are 8-byte words:

   B[8] = A[i] + A[j];

6. For the following C statement, write the corresponding RISC-V assembly code. Assume that the variables i and j are assigned to registers x5 and x6, respectively. Assume that the base address of the arrays A and B are in registers x10 and x11, respectively.

   B[8] = A[i−j];

7. Translate the following C code to RISC-V assembly code. Assume that the values of a, b, i, and j are in registers x5, x6, x7, and x29, respectively. Also, assume that register x10 holds the base address of the array D.

```
for(i=0; i<a; i++)
    for(j=0; j<b; j++)
        D[4*j] = i + j;
```

8. Suppose the program counter (PC) is set to 0x20000000. What range of addresses can be reached using the RISC-V jump-and-link (jal) instruction? (In other words, what is the set of possible values for the PC after the jump instruction executes?) What range of addresses can be reached using the RISC-V branch if equal (beq) instruction? (In other words, what is the set of possible values for the PC after the branch instruction executes?)

9. Assume the following register contents: x5 = 0x00000000AAAAAAAA, x6 = 0x1234567812345678. For the register values shown above, what is the value of x7 for the following sequence of instructions?

```
slli x7, x5, 4
or x7, x7, x6
```

10. Translate the following C code into RISC-V assembly language. You can assume the registers that correspond to the C variables.

```
temp= v[k];
v[k]= v[k+1];
v[k+1] = temp;
```

11. Translate function f into RISC-V assembly language. Assume the function declaration for g is **long long int g(long long int a, long long int b)**. The code for function f is as follows:

```
long long int f(long long int a, long long int b, long long int c, long long int d){
    return g(g(a,b), c+d);
}
```