**CS105 – Computer Systems** Spring 2023

# Week 3: Arithmetic and Conditionals in Assembly

SOLUTION

February 6-8, 2023

1. Match each snippet of assembly code on the left with the equivalent C function on the right.

```
foo1:
    movl %edi,%eax
    sall $4,%eax
    subl %edi,%eax
    ret


foo2:
    movl %edi,%eax
    testl %eax,%eax
    jge .L4
    addl $15,%eax
.L4:
    sarl $4,%eax
    ret


foo3:
    movl %edi,%eax
    shrl $31,%eax
    ret


foo4:
    movl %edi,%eax
    sall $4,%eax
    addl %edi,%eax
    addl %eax,%eax
    ret
```

**Solution:**
foo1 corresponds to choice3
foo2 corresponds to choice5.
foo3 corresponds to choice1
foo4 corresponds to choice8

```
int choice1(int x){
    return (x < 0);
}

int choice2(int x){
    return (x << 31) & 1;
}

int choice3(int x){
    return 15 * x;
}

int choice4(int x){
    return (x + 15) /4
}

int choice5(int x){
    return x / 16;
}

int choice6(int x){
    return (x >> 31);
}

int choice7(int x){
    return x * 30;
}

int choice8(int x){
    return x * 34;
}

int choice9(int x){
    return a * 18;
}
```

2. Consider the following assembly code for a C function `mystery` and compiled on an x86-64 machine:

```
mystery:
    movq    $47, %rax
    cmpq    %rdi, %rsi
    jl      .L1
    addq    %rdi, %rax
    ret
.L1:
    cmpq     %rdi, %rsi
    jg      .L3
    cmpq     %rdi, %rax
    jge     .L2
    ret
.L2:
    addq     %rdi, %rax
    ret
.L3:
    addq     %rsi, %rax
    ret
```

(a) For each variable, indicate which register that variable is stored in.

- x: %rdi
- y: %rsi
- z: %rax

(b) Based on the assembly code, fill in the blanks in the C source code.

```
int mystery(int x, int y) {
    int z = 47;

    if (y-z >= 0){
        return x + z;
    } else if ( y-z == 0)
        if(47 - x >= 0){
             return x + z
        } else {
            return 47;
        }
    } else {
        return y + z;
    }
}
```