

# ICS Homework 6

October 31, 2019

## 1 Arithmetic and Logical Operations

Suppose a 64-bit little endian machine has the following memory and register status:

Address	Value	Register	Value
0x100	0x0000000000002019	%rax	0x2121
0x108	0xffffffffaabb8922	%rbx	0x100
0x110	0x1212121212121212	%rcx	0x2
0x118	0x1300130013001300	%rdx	0x9

Each operation take effect on the status of memory and register, please fill in the blanks in the following table:

Operation	Destination	Value
subq (%rbx),%rax		
incq -8(%rax)		
decq %rdx		
imulq \$4,0x100(%rdx,%rcx,4)		
shrq \$4,%rax		
imulq 0x10		
notw (%rax,%rdx)		
andq 0x10(%rax,%rcx,4),%rax		
leaq 9(%rax,%rcx,8),%rdx		

## 2 Imple C codes from Assembly

You are given prototypes of three functions and assembly codes when they are compiled. Please write C code for the functions that will have equivalent effect as the assembly code shown. Function parameters *a*, *b*, *c*, and *d* are stored in registers %rdi, %rsi, %rdx, and %rcx, respectively.

Function prototypes:

```
1 long f1(long a, long b, long c, long d);
2 long f2(long a, long b, long c);
3 long f3(long a, long b);
```

Assembly codes:

```

1  f1:
2      leaq    (%rsi,%rsi,2), %rax
3      leaq    (%rax,%rdi,2), %rax
4      leaq    (%rdx,%rdx,2), %rdx
5      addq    %rdx, %rax
6      leaq    (%rcx,%rcx,2), %rdx
7      addq    %rdx, %rax
8      ret
9  f2:
10     leaq    -1(%rsi), %rcx
11     imulq   %rcx, %rdx
12     leaq    (%rdx,%rdi,2), %rax
13     imulq   %rsi, %rax
14     xorq    %rdx, %rdx
15     divq    $2
16     ret
17 f3:
18     movq    %rsi, %rax
19     cmpq    %rsi, %rdi
20     jg      .L5
21 .L4:
22     rep ret
23 .L5:
24     movq    %rdi, %rax
25     jmp     .L4

```

Please implement  $f1$ ,  $f2$ , and  $f3$ :