Problem 1

Following the style of Practice Problem 3.1 (page.182), assume the following values are stored at the indicated memory addresses and registers:

Address	Value	--	Register	Value
0x1000	0xAA	--	%rdi	0x1000
0x1004	0xBB	--	%rsi	0x1
0x1008	0xCC	--	%rdx	0x2
0x100C	0xDD	--	%rcx	0x4

Fill in the following table showing the values for the indicated operands:

Operand	Value
%rdi	0x1000
0x1004	0xBB
$0x1008	0x1008
(%rdi)	0xAA
4(%rdi)	0xBB
8(%rdi,%rcx)	0xDD
0x1002(%rdx,%rcx)	0xCC
-4(%rdi,%rsi,4)	0xAA
(%rdi,%rdx,4)	0xCC

Problem 2

Using the Address and Register table in Problem 1, fill in the table below showing the values for the indicated operands (similar to Practice Problem 3.8 of page 194):

Address	Value	--	Register	Value
0x1000	0xAA	--	%rdi	0x1000
0x1004	0xBB	--	%rsi	0x1
0x1008	0xCC	--	%rdx	0x2
0x100C	0xDD	--	%rcx	0x4

Instruction	Destination	Value
addq (%rdi),%rsi	%rsi	0xAB
andq %rsi,%rdi	%rdi	0x0000
subq %rsi,(%rdi)	0x1000	0xA9
incq %rsi	%rsi	0x2
decq %rdx	%rdx	0x1
xorq (%rdi,%rdx,4),%rcx	%rcx	0xC8
orq 0x1002(%rdx,%rcx),%rsi	%rsi	0xCD

Problem 3

For the unknown assembly shown below, fill in the missing return statement in the C function unknown.

```
unknown:
        imulq   %rdx, %rsi
        leaq    (%rsi,%rdi), %rax
        ret

 long unknown(long x, long y, long z) {
      return (x*y+z);
 }

unknown:
        movq    %rdi, %rax
        salq    $3, %rax
        addq    %rdi, %rax
        ret

 long unknown(long x) {
      return  9*x;
 }
```

Problem 4

1. Do problem 3.58 which says "Write C code for the assembly shown in the problem."

For a function with prototype
long decode2(long x, long y, long z);
gcc generates the following assembly code:

```
 Decode2:

        subq   %rdx, %rsi
        Imulq  %rsi, %rdi
        Movq   %rsi, %rax
        Salq   $63, %rax
        Sarq   $63, %rax
        Xorq   %rdi, %rax
        Ret
```

Parameters x, y, and z are passed in registers %rdi, %rsi, and %rdx. The code stores the return value in register %rax.
Write C code for decode2 that will have an effect equivalent to the assembly code shown.

```
long decode2(long x, long y, long z){
        y = y-z;
        x = y*x;
        return ((x<<63)>>63) ^ x;
}
```

Problem 5

Write an assembly program to perform the C function below using the following register assignment:
x:rdi; y:rsi, z:rdx, t1:rcx, t2:rdi, t3:rsi, t4:rax
No credit will be given to the ones with stack addresses.

```
long arith(long x, long y, long z) {
  long t1,t2,t3,t4;
  t1 = x + y;
  t2 = t1 - z;
  t3 = t1 & t2;
  t4 = t2 * t3;

  return t4;
}
```

```
arith:
 leaq (%rdi,%rsi), %rax
 leaq %rax, rcx        # t1
 subq %rdx, %rax
 leaq %rax , %rdi      # t2
 andq %rdi, %rcx
 Leaq %rcx, %rsi      #t3
 imulq %rdi, %rsi
 leaq   %rsi, rax       #t4
 ret
```