

Homework 5

1. Indicate the status (0, 1 or unchanged) of the following flag registers after the instructions being executed. Assume 5 in %eax and -7 in %ebx. The following instructions execute independently and would **NOT** affect each other.

Ins	CF	ZF	SF	OF
leal (%edx,%edx,2),%edx				
addl %eax, %ebx				
xorl %eax, %eax				
cmpl %ebx, %eax				

2. Translate the following assembly into C codes. Note that you can name local variables represented by 28(%esp), 24(%esp)... freely as you like.

```
    movl $1, 28(%esp)
    movl $1, 24(%esp)
    movl $10, 16(%esp)
    movl $3, 20(%esp)
    jmp  L2
L3:
    movl 28(%esp), %eax
    movl %eax, 12(%esp)
    movl 24(%esp), %eax
    movl %eax, 28(%esp)
    movl 12(%esp), %eax
    addl %eax, 24(%esp)
    incl    20(%esp)
L2:
    cmpl $10, 20(%esp)
    jle  L3
    (Following code omitted...)
```

3. Translate the following switch statements into assembly using jump table.

```
int x = <some value>;
int result = 0;
switch (x) {
    case 64:
        result = x + x;
        break;
    case 66:
        result = x + 10;
        break;
    case 68:
```

```
        result = x * 2;  
        // Notice: there is no break here!  
case 69:  
    result = result + 1;  
    break;  
default:  
    result = 3;  
    break;  
}
```

4. Draw a stack for each step when the following function is being called and indicate where %ebp and %esp points. Here is an example where the first step takes place.
int foo(int a, int b, int c);

