(150 pts) CS3843 Computer Organization Exam #2 Name/abc123:_______________

(100 pts) Part 1

Fast Answer: (45 pts, 3 pts each)

1. There are two instructions that efficiently set a register to zero. List one of them. ________

2. After executing either of those instructions, what is the value of the SIGN flag? ________

3. When a value is popped from the stack, the data is read (**before, after**) esp is (**incremented, decremented**).

4. Given that cl = 0xEA, show the value of eax after this instruction: `movsx eax,cl`

5. Given that cl = 0xB5, show the value of eax after this instruction: `movzx eax,cl`

6. A malloc returns the address of memory in (heap, stack, program) memory. ______

7. The following instruction reserves stack space for (**parameters, local variables, global variables**).

   ```
   sub esp, 0x0C
   ```

8. Given the stack frame set up we have used and discussed in class, mov eax, [ebp + 12] will move what into eax? (**global variable, parameter, local variable**)

9. Given that same stack frame set up, what will be in edx after executing this instruction:

   `mov edx, [ebp + 4]`? __________ Hint: It is NOT any of the selections above.

10. In which one of the 3 types of memory are global variables stored? (heap, stack, program)

11. A "pop ebp" instruction uses which register implicitly? _______

12. What is the range of signed displacement values for a 4 byte offset (Express as a power of 2)? ______

13. The stack is always aligned to a _____ byte boundary.

14. What is the difference between a CMP instruction and a SUB instruction? _______

15. What is the difference between a TEST instruction and an AND instruction? _______

Short Answer (30 pts)

16. (6 pts) Given the memory shown below and esp = 0x12F458. What is value of eax after executing a “pop eax” instruction? What is esp after executing the pop instruction?

```
0012F454    FE 22 B0 CA EF BE CE D1 12 58      eax = __________    esp = __________
```

17. (24 pts) Given that [ebp + 0x08] refers to “tmpi” = 0x92EC5, ebp = 0x128F0, esp=0x128E8.

```
CODE A:                                 CODE B:                 .
mov ecx, [ebp + 0x08]       vs.         lea ecx, [ebp + 0x08]
push ecx                                push ecx
call func1                              call func1
```

a. (4 pts) Briefly describe the difference between the CODE A instructions and the CODE B instructions.

b. (4 pts) For each code sample, show the value of ecx as it appears on the stack.

Stack	CODE A:		CODE B:
0x128E4			
0x128E8			
0x128EC			
0x128F0	Prior ebp		Prior ebp
0x128F4			
0x128F8			

c. (4 pts) What is the stack address that will hold the return address of func1?

d. (3 pts) Show tmpi as it appears on the stack prior to executing the code above.

e. (3 pts) Assuming that any local variables in this function are integers, at what addresses are they stored on this stack?

f. (3 pts) What assembly instruction should be immediately after the call above?

g. (3 pts) At what address is the return address for the function with the code that calls func1?

Reading Code (25 pts)

18. (25 pts) Examine the following assembly instructions, and answer the subsequent questions.

```
400020 mov eax, [ebp + 0x0C]     ; value here = 0x0000006C
400023 mov cl,  [ebp + 0x08]     ; value here = 0xFF
400026 cmp al, cl
400028 jl  label1          ; label is at address 40003C, jl <jmp less> is signed
...
Label1:
40003C sub al, cl          ; al = al - cl
40003E jae label2          ; jae <jmp above or equal> is unsigned
...
Label2:                    ; note these addresses are less than those above
400010 neg cl
400011 mov edx,0x40D004
400016 mov edi,edx
400018 mov esi,[edx]
```

a. (5 pts) Given the current values, is the jl taken? (YES, NO)

b. (5 pts) Assuming al and cl remain unchanged at address 40003C, is the jae taken? (YES, NO)

c. (5 pts) What is the value of cl after executing the NEG instruction at address 400010?

d. (5 pts) Describe the difference in values between edi and esi after executing the instructions at addresses 400016 and 400018

e. (5 pts) Both jl and jae are conditional jumps that use relative addressing. The address is relative to what?

f. (8 pts) BONUS: What are the offsets for jl and jae? Show how to determine for partial credit.