Name: _____

Directions: Work only on this sheet (on both sides, if needed); do not turn in any supplementary sheets of paper. There is actually plenty of room for your answers, as long as you organize yourself BEFORE starting writing. In order to get full credit, SHOW YOUR WORK.

1. (15) On typical machines, including Intel, stacks grow toward _____ and thus the stack pointer _____ as the stack grows. (The second blank should contain either "gets smaller" or "gets larger.")

2. (10) In the output of **as -a** on p.3 of the machine-language unit in the printed lecture notes, suppose we had forgotten the **decl** instruction in line 31. Then what would the value 75F8 in line 32 change to?

3. (10) Look at the **call** instruction on p.3, and consider the two bulleted actions at the top of that page. Which one of the following is true?

(i) The two actions will be done during Step A of the call.

(ii) The two actions will be done during Step B of the call.

(iii) The two actions will be done during Step C of the call.

(iv) The two actions will be done during different steps of the call.

4. (5) In class, it was mentioned that some machines allow an immediate operand to be specified in a **ret** instruction, e.g

```
ret $8
```

The goal of this is to eliminate the need to _____.

5. Here is (most of) the output of **as -a** on a certain .s file:

```
 2                      .data
 3 0000 0C000000        x:  .long 12
 4 0004 05000000        y:  .long 5
 5
 6                      .text
 7                      .global _start
 8                      _start:
 9 0000 68000000           pushl $x
 9      00
10 0005 68040000           pushl $y
10      00
11 000a E8030000           call trade
11      00
12                      done:
13 000f 83EC08             subl $8, %esp
14
15                      trade:
16 0012 8B442404           movl 4(%esp),%eax
17 0016 8B5C2408           movl 8(%esp),%ebx
18 001a ****               **********
19 001c ****               **********
20 001e 8910               movl %edx, (%eax)
21 0020 890B               movl %ecx, (%ebx)
22 0022 C3                 ret
```

The subroutine **trade** does what it says, i.e. swaps two words. After the call, x will contain 5 and y will contain 12.

Suppose the .data and .text segments being at 0x1000 and 0x2000, respectively.

(a) (15) Lines 18 and 19 have been censored here. Line 18 deals with the EAX register; state what the assembly-language portion of line 18 is.

(b) (5) Based on the above information and our discussion in class, it would appear that the Intel engineers anticipated that among all the instructions used in this program, _____ would be used most frequently in general usage.

(c) (10) Suppose we have a direct-mapped cache, with line size 256 bytes (not counting the "extra word"), with 64 lines. Then if y is in the cache, which line will it be in?

(d) (5) Infer from the above listing what the general format of a **pushl** instruction is (analogous to our notation that an immediate-to-register move instruction has the format 10111DDDIMM4).

(e) (10) State what values, if any, will be in the MAR and MDR during Step C of the **ret** instruction.

6. Suppose the addone() function in the example in our notes were to return an **int** value, consisting of the old (i.e. pre-incremented) value in the memory location being incremented.

(a) (5) Give a single assembly language instruction to add to addone.s to accomplish this change.

(b) (10) Suppose in TryAddOne.c the call to addone() was done as

```
printf("%d\n",addone(&x));
```

Show the additional line(s) which would be generated by **gcc** in TryAddOne.s.

Solutions:

1. 0; gets smaller

2. 75F9

3. (iii)

4. Clean up the stack after the call.

5.a

```
movl (%eax), %ecx
```

5.b ret (shortest instruction)

5.c (0x1000+0x0004) / 0x100 = 0x10 = 16; 16 mod 64 = 16

5.d 01101000IMM4

5.e MDR: 0x2000+0xf = 0x200f; MAR unknown with data given

6.a

```
movl (%ebx), %eax
```

6.b

```
pushl %eax
pushl $.LC0
```