

CS 31 Homework 3: IA32 Arithmetic

Due at the start of class Thursday, Sept. 30, 2021

Names and Roles:

1. Assume the CPU is executing a program and the state of some of its registers is given in the table below. Show how the registers would be updated by the sequence of IA32 instructions also listed below, i.e. fill in the **Final Value** column. Show your work by listing the intermediate values of the registers.

Register	Initial Value	Final Value
%eax	0	
%ebx	1	
%ecx	2	
%edx	3	

Here are the IA32 instructions:

```
addl    $20, %eax
addl    %eax, %ebx
subl    %ecx, %ebx
mull    $3, %edx
mull    %edx, %ecx
addl    %edx, %edx
decl    %edx
xorl    %eax, %eax
shrl    $4, %edx
andl    $0xffff, %ebx
orl     $0x0, %ecx
```

2. Assume the CPU is executing a function that has local variables `x`, `y`, and `z` allocated on the stack, and that `x` is allocated at the memory address that is `-12` bytes from the address value stored in register `%ebp`, or `-12(%ebp)`. Assume `y` is stored at `-8(%ebp)`, and `z` is at `-4(%ebp)`.

For the assembly code and register values listed below:

(1) Show the values that will be stored in the registers and in memory when execution of these instructions is complete. If the value is unknown, write “?”.

(2) Write a C code translation of the assembly code sequence. You may assume that `x`, `y`, and `z` have already been declared as `int` variables in the C code. You do not need to write the entire function, just the lines of C that might have generated the IA32 instructions. Hint: our solution is 5 lines of C code.

```
movl    $1, -12(%ebp)
movl    $2, -8(%ebp)
movl    -12(%ebp), %edx
movl    -8(%ebp), %eax
addl    %edx, %eax
movl    %eax, -4(%ebp)
incl    -12(%ebp)
sall    $1, -8(%ebp)
```

C Code Translation

Register	Initial Value	Final Value
<code>%eax</code>	4	
<code>%edx</code>	7	
<code>%ebp</code>	0xff44	

Memory Address	Final Value
0xff38	
0xff3c	
0xff40	
0xff44	
0xff48	