cs154: Introduction to Computer Systems **Homework 2**
Spring 2020 (Assigned Apr 13)
Due Apr 20 11:59pm

Submit your answers by adding and committing one file with your answers into the `hw2` directory of your `CNETID-cs154-spr-20` svn repository[1]. The file should be named either `hw2.txt` or `hw2.pdf` for answers written in a plain ASCII text file or PDF file, respectively. PDFs of scanned hand-written pages must not exceed 5 megabytes. No other file formats or filenames are acceptable, and no files besides `hw2.txt` or `hw2.pdf` will be graded. Not following directions may result in losing points.

Note: Some of the operands and instructions below use the scaled indexed addressing mode. As the caption for Figure 3.3 in the book explains, for x86 the valid scaling factors are 1, 2, 4, or 8. For this homework and this homework only, however, you should assume that there are no restrictions on scaling factors.

(Q1. 12 points + 2 bonus points) The following values are stored at the indicated memory addresses and registers:

Address	Value	Register	Value
`0x210`	`0xAB`	`%rax`	`0x210`
`0x218`	`0xBC`	`%rcx`	`0x220`
`0x220`	`0x09`	`%rdx`	`0x1`
`0x228`	`0x42`	`%rsi`	`0x8`
`0x230`	`0x54`	`%rdi`	`0x100`

Assume that the Values occupy eight bytes of storage, but the endianness of storage is not specified. Each part of this question concerns one item in the following list of operands. For the operands, give the Value of the operand, and the addressing Mode used (copying from the "Name" column of Figure 3.3 in the textbook). If the facts given here don't provide sufficient information to answer the question unambiguously, for Value say "unknown", and for Mode give an explanation of what information is missing, without which there isn't an unambiguous answer. **Give all Values in hex** (unsigned), starting with "`0x`".

Part	Operand	Value	Mode
e.g.	`%rax`	`0x210`	Register
A.	`(%rcx)`		
B.	`560`		
C.	`(%rax,%rsi)`		
D.	`0x108(%rdi,%rdx,0x20)`		
E.	`$251`		
F.	`(,%rsi,0x45)`		
(Bonus) G.	`0x14(%rax)`		

[1] see https://classes.cs.uchicago.edu/current/15400-1/svn.html

(Q2. 20 points) Assume the same information about addresses, registers, and values as in the previous question. The table is reprinted here for convenience:

Address	Value	Register	Value
0x210	0xAB	%rax	0x210
0x218	0xBC	%rcx	0x220
0x220	0x09	%rdx	0x1
0x228	0x42	%rsi	0x8
0x230	0x54	%rdi	0x100

Each part of this question concerns a single instruction. Each instruction is to be considered in isolation (the answer for one instruction does not affect any other instruction). The argument ordering for the instructions is the same ATT format (not Intel format) as used in the book. Negative values are represented with two's complement.

For each part, give the Destination for the instruction (as either "0xADDR" for address ADDR or as '%reg" for register reg), and the (unsigned) Value (**in hex**, starting with "0x") stored in the Destination. If the facts given don't provide sufficient information to answer the question unambiguously, give "unknown" for Destination, and a terse explanation why for Value.

Part	Instruction	Destination	Value
e.g.	`addq %rdx,(%rax)`	0x210	0xAC
A.	`incq %rsi`		
B.	`decq (%rax)`		
C.	`notq %rdx`		
D.	`shlq %rsi,(%rcx)`		
E.	`subq $0x100,%rcx`		
F.	`imulq %rdi,(%rax,%rsi)`		
G.	`orq (%rcx),%rax`		
H.	`shrq %rdx,(%rdi,%rsi,37)`		
I.	`xorq %rbx,%rbx`		
J.	`subq 0x220,%rdi`		

Note: For this homework, it is fine to use a calculator or some other program to aid with arithmetic.