TOY PROGRAMMING WORKSHEET

1. TOY has 16 registers, numbered R[0] through R[F]

2. One of these registers is special. Which is it? R[0] What makes it special?

   R[0] always has the value 0.

3. TOY has 256 memory locations, addressed 00 through FF

4. One of these memory locations is special. Which is it? FF What makes it special?

   We write to FF to use StdOut and read from FF for StdIn.

5. TOY has 16 op codes, numbered 0 through F

6. TOY has one PC. What does "PC" stand for? Program Counter

7. What does the PC do?

   The PC keeps track of which memory address is being executed.

8. Use the TOY cheatsheet to fill in the following blanks:

MEM	CODE	PSEUDO-CODE	HUMAN-READABLE COMMENTS
10:	7101	R[1] <- 01	R[1] gets the constant 1
11:	4222	R[2] <- R[2] ^ R[2]	Initialize R[2] to 0
12:	7301	R[3] <- 01	R[3] gets the constant 1
13:	85FF	R[5] <- mem[FF]	Read N from StdIn (R[5])
14:	1423	R[4] <- R[2] + R[3]	Keep a running sum in R[4]
15:	1203	R[2] <- R[0] + R[3]	Copy R[3] value into R[2]
16:	1304	R[3] <- R[0] + R[4]	Copy R[4] value into R[3]
17:	2551	R[5] <- R[5] - R[1]	Subtract 1 from N (R[5])
18:	D514	if (R[5]>0) pc<-14	N > 0? Do another iteration.
19:	94FF	mem[FF] <- R[4]	Send the sum in R[4] to StdOut
1A:	0000	halt	When we get here, we're done!

9. List 6 TOY instructions, each having a different op code, that copy the value of R[B] into R[A].

1AB0, 2AB0, 3ABB, 4AB0, 5AB0, 6AB0

10. List 7 TOY instructions, each having a different op code, that put 0000 in R[A].

1A00, 2A00, 3A00, 4A00, 5A00, 6A00, 7A00

11. Write a TOY program that sets R[B] to the absolute value of R[A]. (Can be done in 4 or 5 lines.)

MEM	CODE	PSEUDO-CODE	HUMAN-READABLE COMMENTS
10:	DA13	if (R[A]>0) pc<-13	if R[A]>0 go to R[B]<-R[A]
11:	2B0A	R[B] <- R[0] - R[A]	otherwise, R[B] gets -R[A]
12:	C014	if (R[0]==0) pc<-14	then go directly to halt
13:	1BA0	R[B] <- R[A] + R[0]	R[B] gets copy of R[A]
14:	0000	halt	All done!

Can you trace this four line solution?

```
10: 1BA0
11: DB13
12: 2B0B
13: 0000
```

12. Consider the following TOY program. What is the value of R[3] when it halts? 128

Stored as 0080 in memory, because all TOY data is stored in hex, not decimal.

MEM	CODE	PSEUDO-CODE	HUMAN-READABLE COMMENTS
10:	7101	R[1] <- 01	R[1] initialized to 1
11:	7207	R[2] <- 07	R[2] initialized to 7
12:	7301	R[3] <- 01	R[3] initialized to 1
13:	1333	R[3] <- R[3] + R[3]	double R[3]
14:	2221	R[2] <- R[2] - R[1]	decrement R[2]
15:	D213	if (R[2]>0) pc<-13	if (R[2]>0) go to 13
16:	0000	halt	Done!