CSE 370 – Introduction to Digital Logic Design
Spring 2010
Quiz #9

For maximum credit, show all your work. Please raise your hand if you have a question.
Write a program for our x370 processor that applies Euclid's original algorithm to 2 numbers to find the greatest common divisor. For those of you who have forgotten Euclid's original algorithm, a C/Java program that finds the GCD of x and y, with the result in y, is shown below.
Hint: Write your program first to the right as shown (using whatever notation you like as long as I can understand it.) And then turn it into binary on the left.
[Note: The > is a typo – it should have been >= : the solution is for the code as written.]

```
x = 12;
y =  20;
while (x != 0) {
   if (x > y) x = x − y;
  else  y = y − x;
}
```

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	1	0	1	1	1	0	0	0	0	0	0	0	1	1	0	0	LDI r0 <- 12 // x = 12
1	1	0	1	1	1	0	0	1	0	0	0	1	0	1	0	0	LDI r1 <- 20 // y = 20
2	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	START: BRZ R0, END
3	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	1	SUB R2 <- R0, R1 (R2 = y - x)
4	0	0	0	1	0	0	0	0	0	0	1	1	1	0	1	0	BRN R2, X>Y
5	1	0	0	1	1	0	0	1	0	0	0	1	0	0	0	0	PASSA R1 <- R2
6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	BR START (we could branch to 3)
7	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	X>Y: SUB R0 <- R1, R0 (x = x - y)
8	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	BR START
9	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	END: BR END
A																	
B																	
C																	
D																	
E																	