CSE 321a: Computer Organization (I)
Third Year, Computer & Systems Engineering

Assignment #3

Due date: **Thursday, December 21st, 2017**

1. **Consider an instruction set architecture with the following characteristics: …**
   (a) Translate the following C-language snippet to the assembly language of this instruction set

```
      MOV R1, #2
FOR:  BRN R1, OUT /* BRN R1, @7 */
      ASL x(R1), #3
      DEC R1
      BR FOR /* BR @-10 */
OUT:
```

   (b) Fill up the following table with the rest of the machine program …

Address (Hexadecimal)	Contents (Hexadecimal)
5B	A4
5C	04
5D	02
5E	D4
5F	07
60	07
61	BA
62	E4
63	03
64	F5
65	01
66	EF
67	8A

(c) Show, using the table below, the execution trace of the machine program (from part 2)

Instruction	PC	R1	Memory Locations		
			2E	2F	30
Initially	5B	C5	0F	85	03
A40402	5E	02	0F	85	03
D40707	61	02	0F	85	03
BAE403	64	02	0F	85	18
F501	66	01	0F	85	18
EF8A	5E	01	0F	85	18
D40707	61	01	0F	85	18
BAE403	64	01	0F	A8	18
F501	66	00	0F	A8	18
EF8A	5E	00	0F	A8	18
D40707	61	00	0F	A8	18
BAE403	64	00	78	A8	18
F501	66	81	78	A8	18
EF8A	5E	81	78	A8	18
D40707	68	81	78	A8	18

2. A CPU implements a stack-based (*i.e.*, zero-address) instruction set ..

(a) `NEGATE`

t_1: MAR ← [SP]
t_2: MBR ← Memory
t_3: MBR ← - [MBR]
t_4: Memory ← [MBR]

(b) **`EXCHANGE`** /* ***Hint****: Assume CPU has special-purpose registers for holding temporary data* */

t_1: MAR ← [SP]
t_2: MBR ← Memory
t_3: Temp ← [MBR], MAR ← [SP] + 1
t_4: MBR ← Memory, MAR ← [SP]
t_5: Memory ← [MBR]
t_6: MBR ← [Temp], MAR ← [SP] + 1
t_7: Memory ← [MBR]

(c) **`PUSH 100(R1)+`**

t_1: MAR ← [IR_Address] + [R1], SP ← [SP] – 1
t_2: MBR ← Memory, R1 ← [R1] + 1, MAR ← [SP]
t_3: Memory ← [MBR]

3. A pipelined processor has the following features: …

(a) Draw a timing diagram to show instr. pipelining during the first 3 iterations of the loop.

Suppose the three instructions located after "**JNZ**" in memory are: I_1, I_2, and I_3 respectively.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
INC	FI	DI	FO	EI											
JNZ		FI	DI		FO	EI									
I_1			FI		DI	FO									
I_2					FI	DI									
I_3						FI									
INC							FI	DI	FO	EI					
JNZ								FI	DI		FO	EI			
INC									FI		DI	FO	EI		
JNZ											FI	DI		FO	EI
	Iteration #1 (6 Cycles)						Iteration #2 (6 Cycles)						Iteration #3 (3 Cycles)		

(b) How many clock cycles are needed to execute this program?

Number of clock cycles = 6 + 6 + 98 * 3 = 306 cycles