NAME:- KRUNAL RANK

ADM. NO:- U18C0081

B TECH 3RD YEAR

U18CO081

31/8/20

MIT Tutorial 2

Ans 1:

2:

Line No.	Mnemonics
1	LXI SP, 0400H
2	LXI B, 2055H
3	LXI H, 22FFH
4	LXI D, 2090H
5	PUSH H
6	PUSH B
7	MOV A, L
↓	↓
20	POP H

a) SP pair stores 0400H after execution of first line.

b) The location of ~~data~~ first byte stored in stack will be 03FEH.

c) When line 5 is executed, 22FFH is stored at 03FEH.

d) The address of stack pointer is 03FEH and the value stored there is 2055H.

e) The contents of register pair HL will be the contents of the stack value popped. If no push or pop is done between line 8 to 19, the value of HL pair will be 2055H.

Ans 3:

Location	Mnemonics
2000	LXI SP, 20CD H
2003	LXI H, 0008 H
2006	MVI B, OF H
2008	CALL 2060H
200B	OUT 01H
	DCR B
	↓ CONTD
2060	PUSH H
2061	PUSH B
	MVI B, 05H
	LXI H, COUNT
	↓
	POP B
	POP H
	RET

a) Content at 20CCH = 0008H 20H
Content at 20CBH = OF00H OBH
Content at SP = 20CBH
The program counter will have the same address as the Pointer

b) Stack location after execution of PUSH H = 20C9H 20C9H
Stack pointer location after execution of PUSH B = 20C8H 20C7H

c) Content of stack pointer = 20CBH 20C7H

d) Content of stack pointer = 20CBH

Ans 8)

a) ;< Question 17>
jmp start
;data

;code
start: lxi SP, XX99H ; XX can be replaced with 00-FF
mvi A, 0FH ; sets 0FH to accumulator
lxi H, XX90H ; initialises HL pair
loop: cmp B ; compares accumulator with B
jz next ; jump on zero at next
dcr A ; decrement accumulator
mvi M, 00H ; move 00H to memory location
inx H ; increment HL pair
jmp loop ; jump to loop

next: lxi B, 0237H ; initialise BC pair
lxi D, 1242H ; initialise DE pair
lxi H, 4087H ; initialise HL pair
PUSH B ; push B to stack
push D ; push D to stack
push H ; push H to stack

The value of memory locations are:-

2093H	35	(Decimal)
2094H	64	
2095H	66	
2096H	18	Stack pointer: 2093H
2097H	55	
2098H	2	

```
9;  ;<question2>
    jmp start

;data

;code
start:  mvi B, 00h
        inr B               ; clears parity, zero and aux flags
        stc
        cmc                 ; clears carry flag
        mvi A, 00ffh        ; Moves FFH to A
        adi 0001h           ; Adds 1 into A
        mvi B, 00h          ;
        inr B               ; Marks other flags to 0
        jnc exit            ; If not carry, jump to exit
        mvi A, 01h          ; Move 1 to A
        out 00H             ; Display A to 00H O/P
exit:   hlt
```

The same is done by replacing adi instruction with inr instruction.

In the latter case, carry flag was not set to 1.

This can be explained as follows:-

INR Instruction is an increment instruction that follows wrap around on overflow.
ADI instruction is an addition instruction that does not preserve the carry flag.

```
;<Question 3>
jmp start

;data
;code
start: push PSW
       pop H
       mov C,H
       mov H,B
       mov B,C
       push H
       pop PSW
hlt
```