8085 PROGRAMMING

ADD A SERIES OF NUMBERS

Q1) Add a series of ten 8-bit numbers stored from 2000H. Store Result at 200AH and 200BH.

Solution:

```
        LXI H, 2000H    ; HL = 2000H; M = 1st number
        MVI C, 0AH      ; Count
        MVI A, 00H      ; Sum
        MVI B, 00H      ; Carry
BACK:   ADD M           ; Add current number
        JNC SKIP        ; Check for carry
        INR B           ; Increment B register if Carry Flag = 1
SKIP:   INX H           ; Point to next number
        DCR C           ; Decrement count
        JNZ BACK        ; Loop
        MOV M, A        ; Store Sum
        INX H           ; Point to next location
        MOV M, B        ; Store Carry
        HLT             ; End of program
```

Special Note:
Please refer our videos for full explanation of the programs and the diagram of the logic.

FIND HIGHEST NUMBER

Q2) Find the highest in a series of ten 8-bit numbers stored from 2000H. Store Result at 200AH.

Solution:

```
        LXI H, 2000H    ; HL = 2000H; M = 1st number
        MVI C, 0AH      ; Count
        MVI A, 00H      ; Highest number

BACK:   CMP M           ; Compare current number
        JNC SKIP        ; Check for carry
        MOV A, M        ; If carry is 1 then A ← M
SKIP:   INX H           ; Point to next number
        DCR C           ; Decrement count
        JNZ BACK        ; Loop
        MOV M, A        ; Store Result
        HLT             ; End of program
```

FIND EVEN AND ODD NUMBERS

Q3) Find the number of even and odd numbers in a series of ten 8-bit numbers stored from 2000H. Store Result at 200AH and 200BH.

Solution:

```
        LXI H, 2000H    ; HL = 2000H; M = 1st number
        MVI C, 0AH      ; Count
        MVI D, 00H      ; Even count
        MVI B, 00H      ; Odd count
BACK:   MOV A, M        ; Get number into A
        RRC             ; Check LSB
        JC ODD          ; If carry, its Odd
        INR D           ; Increment Even count
        JMP SKIP        ; Move ahead
ODD:    INR B           ; Increment Odd count
SKIP:   INX H           ; Point to next number
        DCR C           ; Decrement count
        JNZ BACK        ; Loop
        MOV M, D        ; Store Even count
        INX H           ; Move to next location
        MOV M, B        ; Store Odd count
        HLT             ; End of program
```

FIND NUMBER OF ONES

Q4) Find the number of Ones in an 8-bit number stored at 2000H. Store Result at 2001H.

Solution:

```
        LDA 2000H     ; A = given number
        MVI B, 00H    ; Number of 1s
        MVI C, 08H    ; Loop count

BACK:   RRC           ; Rotate right
        JNC SKIP      ; If no carry, move ahead
        INR B         ; Increment 1s count
SKIP:   DCR C         ; Decrement loop count
        JNZ BACK      ; Loop
        MOV A, B      ; A ← 1s count from B
        STA 2001H     ; Storfe result
        HLT           ; End of program
```