Lab/Experiment No. 7

Aim: WRITE A PROGRAM USING 8085 & VERIFY FOR:

1. TO FIND THE NUMBER OF EVEN AND ODD NUMBERS IN A DATA ARRAY.
2. TO FIND THE SUM OF EVEN NUMBERS AND ODD NUMBERS IN A DATA ARRAY.
3. TO FIND THE FACTORIAL OF A NUMBER.
4. TO FIND THE FIRST TEN ELEMENTS OF A FIBONACCI SERIES.
5. TO FIND THE SUM OF UPPER AND LOWER NIBBLE OF AN 8-BIT NUMBER.
6. ADDITION OF TWO 8-BIT NUMBERS USING A SUBROUTINE.

ALP

```
; Main Program
LXI H, 3000H    ; Load H-L pair with address 3000H
MOV A, M        ; Load first 8-bit number from memory into Accumulator
INX H           ; Increment H-L pair to point to the next memory location
MOV B, M        ; Load second 8-bit number into register B
CALL ADD_SUB    ; Call the subroutine to add A and B
STA 3002H       ; Store the result at memory location 3002H
MOV A, C        ; Move the carry flag (in C) to A for storing
STA 3003H       ; Store the carry at memory location 3003H

HLT             ; Halt the program

; Subroutine to add two 8-bit numbers with carry
ADD_SUB: ADD B          ; Add the contents of register B to A (result in A)
MOV C, 00H      ; Clear register C to store the carry
JNC NO_CARRY    ; If no carry, jump to NO_CARRY
INR C           ; If carry, increment C (C = 01H)

NO_CARRY: RET           ; Return to the calling program
```

ALP to find the sum of series using a subroutine

```
; Main Program
MVI D, 00H      ; Clear D register to store the higher byte of the sum (for carry)
MVI E, 00H      ; Clear E register to store the lower byte of the sum
LXI H, 3000H    ; Load H-L pair with the starting address of the series
MOV C, M        ; Load the count of numbers in the series into register C
INX H           ; Point to the first number in the series

SUM_LOOP: MOV A, M        ; Load the current number into Accumulator
CALL ADD_TO_SUM  ; Call the subroutine to add the number to the sum
INX H           ; Move to the next number in the series
DCR C           ; Decrement the count
JNZ SUM_LOOP    ; If count is not zero, repeat the loop

STA 3100H       ; Store the lower byte of the sum at memory location 3001H
```

```
MOV A, D         ; Move the higher byte of the sum (D) into Accumulator
STA 3101H        ; Store the higher byte of the sum at memory location 3002H

HLT              ; Halt the program

; Subroutine to add a number to the current sum
ADD_TO_SUM: ADD E            ; Add the number in A to the lower byte of the sum (E)
MOV E, A         ; Move the result back to E
JNC NO_CARRY     ; If no carry, skip to NO_CARRY
INR D            ; Increment the higher byte (D) if there's a carry

NO_CARRY: RET              ; Return to the calling program
```

Please write the ALP for the above problems and bring it to the lab session.