Lab- 6:

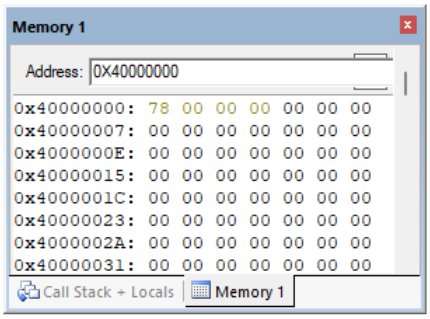
Objective - 1: Find the factorial of a given 8-bit number.

Program:

```
        AREA FACTORIAL, CODE, READONLY
        ENTRY
START
        MOV R0, #5        ;n
        MOV R1, #1        ;result
fact_loop
        CMP R0, #0
        BEQ end_loop      ; If R0 = 0, branch to end_loop
        MUL R2, R1, R0
        MOV  R1, R2        ; R1 = R1 * R0
        SUBS R0, R0, #1    ; R0 = R0 - 1
        B fact_loop       ; Repeat the loop
end_loop
        LDR R2, =RESULT    ; Store the result in memory
        STR R1, [R2]
EXT B EXT
        AREA RES, DATA, READWRITE
RESULT DCD 0X0
        END

;       C-CODE FOR FACTORIAL
;if (n == 0) {
;       return 1;
;  }
;   return n * factorial(n - 1);
; }
```

```
    AREA fACTORIAL,CODE,READONLY
    ENTRY
START
    MOV R0, #5          ;n =5
    MOV R1, #1          ;RESULT INITIALIZE TO 1
fact_loop
    CMP R0, #0
    BEQ end_loop        ; If R0 = 0, branch to end_loop
    MUL R2,R1,R0
    MOV  R1,R2          ; R1 = R1 * R0
    SUBS R0, R0, #1     ; R0 = R0 - 1
    B fact_loop         ; Repeat the loop
end_loop
    LDR R2, =RESULT     ; Store the result in memory
    STR R1, [R2]
```



Objective - 2: Find the Fibonacci Series up to n digits.

Program:

```
        AREA fibnacci, CODE, READONLY
        ENTRY
START
        MOV R1, #01
        MOV R2, #00
        MOV R3, #00
        LDR R0, =COUNT
        LDR R6, =FIB_SERIES
        LDRB R5, [R0]
        CMP R5, #1
        BLE STOP1
        STRB R2, [R6], #01
        SUBS R5, R5, #01
        STRB R1,[R6],#01
BACK
        SUBS R5,R5,#01
        BEQ STOP
        ADD R3,R1,R2
        STRB R3,[R6],#01
        MOV R2,R1
        MOV R1,R3
        B BACK
STOP1  STRB R3,[R6]
STOP    B  STOP
COUNT DCB 0X0A     ; N =10
        AREA DATA1, DATA, READWRITE
```

FIB_SERIES DCB 0X0,0X0,0X0,0X0,0X0,0X0,0X0,0X0,0X0,0X0,0X0

END

RESULT:

```
1     AREA fibnacci,CODE,READONLY
2     ENTRY
3   START
4           mov r1,#01
5           mov r2,#00
6           mov r3,#00
7           ldr r0,=count
8           ldr r6,=fib_series
9           ldrb r5,[r0]
10          cmp r5,#1
11          ble stop1
12          strb r2,[r6],#01
13          subs r5,r5,#01
14          strb r1,[r6],#01
15  back    subs r5,r5,#01
16          beq stop
```