Lab-7:

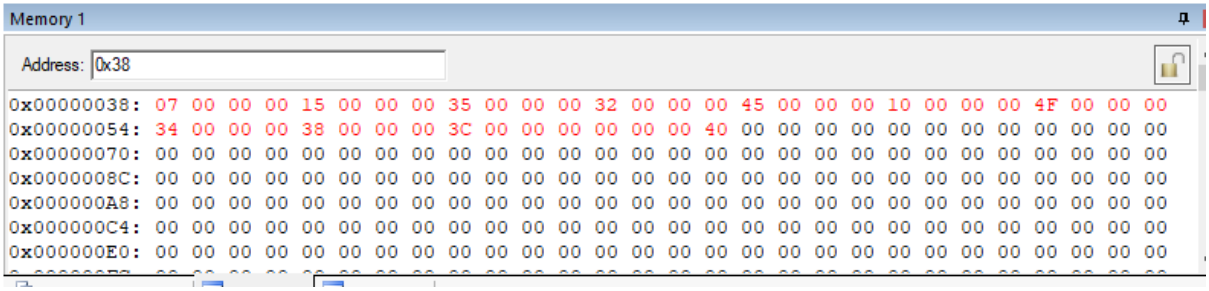
Objective 1: **Find the Largest number from a given array of size N using ARM assembly Language.**

Program:

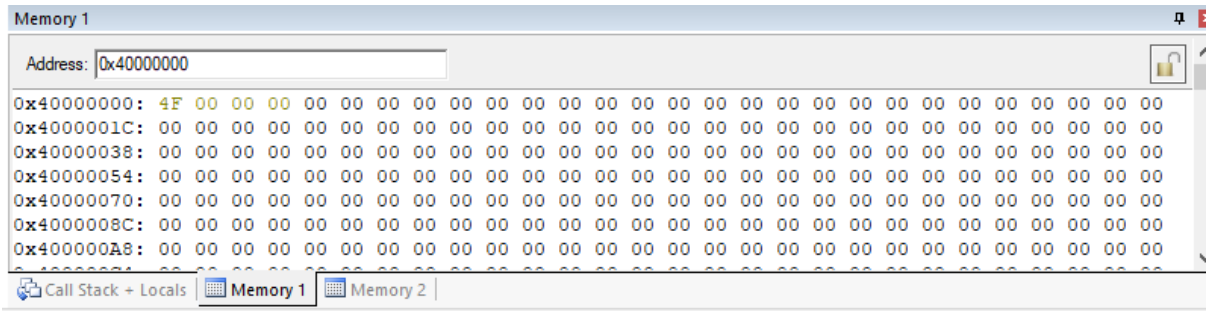
```
        AREA PROG1, CODE,READONLY
        ENTRY
START
        ;Largest number from a given array
        ldr r0,=count
        ldr r1,[r0]       ; r1= array size
        ldr r2,=array
   ldr r3, [r2],#4
back
        subs r1,r1,#01
        beq fwd
        ldr r4,[r2],#4
        cmp r3,r4
        bgt back
        mov r3,r4
        b back
fwd
   ldr r5,=RESULT
        str r3,[r5]
exit b exit

count DCD 0x07
array
                DCD 0x15 ; DCD= Define Constant Double-words(32-bit)
                DCD 0x35 ; DCD directive allocates one or more words of memory, aligned
                         ; on 4-byte boundaries
                DCD 0x32
                DCD 0x45
                DCD   0x10
                DCD 0x4f
                DCD 0x34
        AREA DATA2,DATA,READWRITE ; TO STORE RESULT IN GIVEN ADDRESS
lARGEST  DCD 0X0
        END
```

Result:

Input Location:



Output Locations:



Objective-2 : Separate Even numbers and odds numbers in an array of size N using ARM Assembly language.

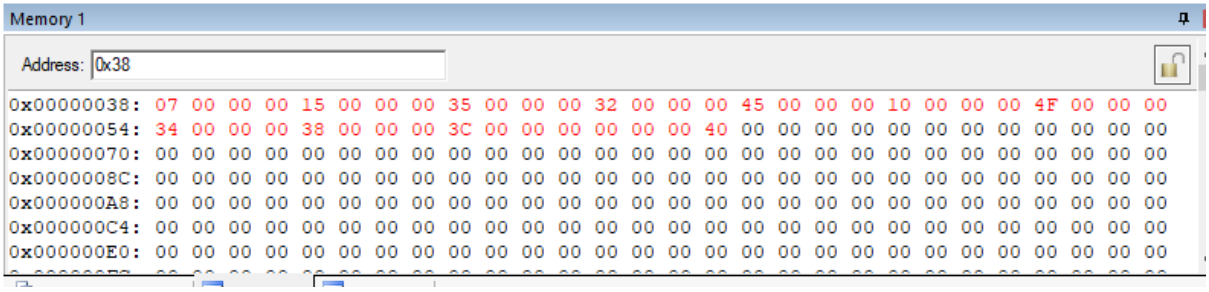
Program:

```
        AREA prog2, CODE, READONLY
        ENTRY      ;Mark first instruction to execute
START
        ldr r0,=count
        ldr r1,[r0]
        ldr r3,=array ; r3 = base address of array=array[0]
        ldr r4,=even  ; r4=base address of even data locations as constant = even[0]
                                                                    ; = 0x40000000
        ldr r5,=odd   ; r5=base address of odd data locations as constant = odd[0]
                                                                    ; = 0x4000001c
back
        ldr r6, [r3],#4
        ands r7,r6,#1
        beq fwd
        str r6,[r5],#4
        b fwd1
fwd
        str r6,[r4],#4
```

```
fwd1
        subs r1,r1,#01
        bne back

exit b exit
; Array declaration
count DCD 0x07
array
                DCD 0x15
                DCD 0x35
                DCD 0x32
                DCD 0x45
                DCD   0x10
                DCD 0x4f
                DCD 0x34
        AREA DATA2,DATA,READWRITE ; TO STORE RESULT IN GIVEN ADDRESS
even
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
odd
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        DCD 0X0
        END
```

Result:

Input Location:



Output Location: