LAB5

Aniket sambher

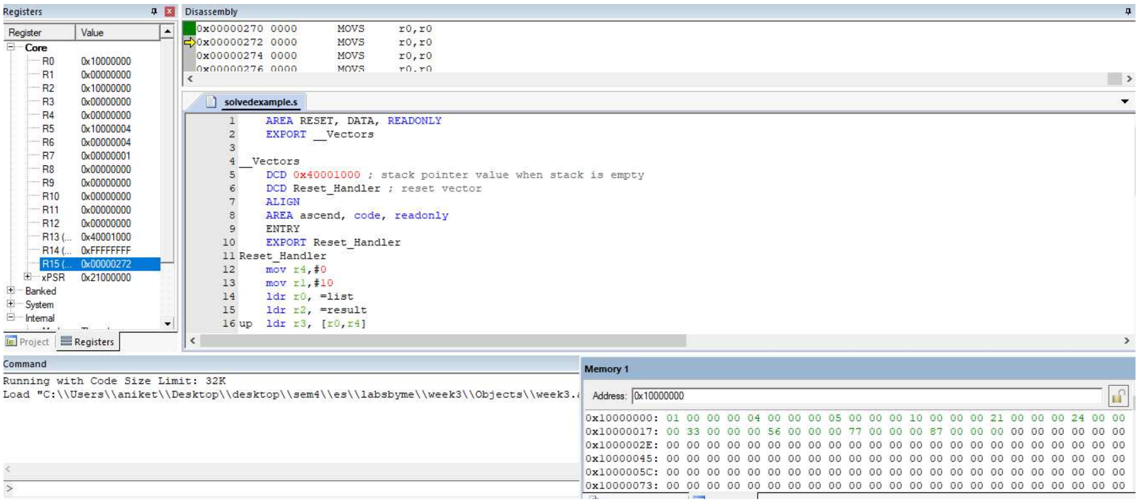
Reg no-190905466

Section -A

Roll no-58

SOLVED EXAMPLE

```
        AREA RESET, DATA, READONLY
        EXPORT __Vectors

__Vectors
        DCD 0x40001000 ; stack pointer value when stack is empty
        DCD Reset_Handler ; reset vector
        ALIGN
        AREA ascend, code, readonly
        ENTRY
        EXPORT Reset_Handler
Reset_Handler
        mov r4,#0
        mov r1,#10
        ldr r0, =list
        ldr r2, =result
up  ldr r3, [r0,r4]
        str r3, [r2,r4]
        add r4, #04
        sub r1,#01
        cmp r1,#00
        bhi up
        ldr r0, =result
        mov r3, #10 ; inner loop counter
        sub r3, r3, #1
        mov r9, r3 ; R9 contain no of passes
        ; outer loop counter
outer_loop
        mov r5, r0
        mov r4, r3 ; R4 contains no of comparison in a pass
inner_loop
        ldr r6, [r5], #4
        ldr r7, [r5]
        cmp r7, r6
        ; swap without swap instruction
        strls r6, [r5]
        strls r7, [r5, #-4]
        subs r4, r4, #1
        bne inner_loop
        sub r3, #1
        subs r9, r9, #1
        bne outer_loop
list dcd 0x10,0x05,0x33,0x24,0x56,0x77,0x21,0x04,0x87,0x01
        AREA data1, data, readwrite
result DCW 0,0,0,0,0,0,0,0,0,0
        end
```



Register	Value
Core	
R0	0x10000000
R1	0x00000000
R2	0x10000000
R3	0x00000000
R4	0x00000000
R5	0x10000004
R6	0x00000004
R7	0x00000001
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (...	0x40001000
R14 (...	0xFFFFFFFF
R15 (...	0x00000272
xPSR	0x21000000
Banked	
System	
Internal	

```
0x00000270 0000      MOVS     r0,r0
0x00000272 0000      MOVS     r0,r0
0x00000274 0000      MOVS     r0,r0
0x00000276 0000      MOVS     r0,r0
```

```
    AREA RESET, DATA, READONLY
    EXPORT __Vectors

__Vectors
    DCD 0x40001000 ; stack pointer value when stack is empty
    DCD Reset_Handler ; reset vector
    ALIGN
    AREA ascend, code, readonly
    ENTRY
    EXPORT Reset_Handler
Reset_Handler
    mov r4,#0
    mov r1,#10
    ldr r0, =list
    ldr r2, =result
up  ldr r3, [r0,r4]
```

```
Running with Code Size Limit: 32K
Load "C:\\Users\\aniket\\Desktop\\desktop\\sem4\\es\\labsbyme\\week3\\Objects\\week3.
>
```

Address: 0x10000000

```
0x10000000: 01 00 00 00 04 00 00 00 05 00 00 00 10 00 00 00 21 00 00 00 24 00 00
0x10000017: 00 33 00 00 00 56 00 00 00 77 00 00 00 87 00 00 00 00 00 00 00 00 00
0x1000002E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000045: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1000005C: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000073: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

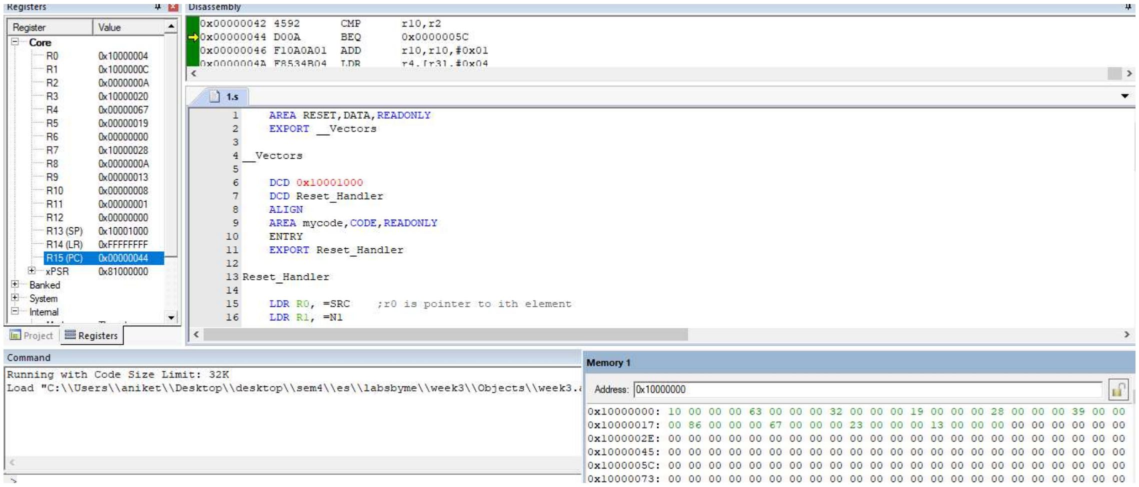
Q1 Selection Sort

```
        AREA RESET,DATA,READONLY
        EXPORT __Vectors

__Vectors

        DCD 0x10001000
        DCD Reset_Handler
        ALIGN
        AREA mycode,CODE,READONLY
        ENTRY
        EXPORT Reset_Handler

Reset_Handler

        LDR R0, =SRC    ;r0 is pointer to ith element
        LDR R1, =N1
        LDR R2,[r1]             ;r2 stores number of elements
        LDR R7, =DST
        MOV R8,#0
up      CMP R8,R2
        BEQ out
        ADD R8,#1
        LDR R9,[R0],#4
        STR R9,[R7],#4
        B up
out     LDR R0,=DST
        MOV R1, R0              ;r1 is pointer to element to swap
        MOV R3,R0               ;r3 is pointer to jth element
        MOV R10,#0              ;r10 is counter for inner(j) loop
        MOV R11,#0              ;r11 is counter for outer(i) loop
```

```
lp1     CMP R11, R2             ;comparing i<10
        BEQ exit
        ADD R3,R0,#4    ;sets jth pointer to A[i+1]
        MOV R1,R0               ;sets swap element to A[i]
        ADD R10,R11,#1          ;j=i+1
lp2     CMP R10,R2              ;j<10
        BEQ oif
        ADD R10,#1              ;j++
        LDR R4,[R3],#4
        LDR R5,[R1]
        CMP R5,R4
        BLT lp2
        MOV R1,R3
        SUB R1,#4
        B lp2
oif     ADD R11,#1
        LDR R4,[R0]
        LDR R5,[R1]
        STR R4,[R1]
        STR R5,[R0],#4
        B lp1
exit


STOP
        B STOP


N1 DCD 0xA
SRC DCD 0x32,0x63,0x10,0x19,0x28,0x39,0x86,0x67,0x23,0x13


        AREA mydata,DATA,READWRITE
```

DST DCD 0,0,0,0,0,0,0,0,0,0

END



Register	Value
Core	
R0	0x10000004
R1	0x1000000C
R2	0x0000000A
R3	0x10000020
R4	0x00000067
R5	0x00000019
R6	0x00000000
R7	0x10000028
R8	0x0000000A
R9	0x00000013
R10	0x00000008
R11	0x00000001
R12	0x00000000
R13 (SP)	0x10001000
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x00000044
xPSR	0x81000000
Banked	
System	
Internal	

```
0x00000042 4592      CMP      r10,r2
0x00000044 D00A      BEQ      0x0000005C
0x00000046 F10A0A01  ADD      r10,r10,#0x01
0x0000004A F8534B04  LDR      r4,[r3],#0x04
```

```
    AREA RESET,DATA,READONLY
    EXPORT __Vectors

__Vectors

    DCD 0x10001000
    DCD Reset_Handler
    ALIGN
    AREA mycode,CODE,READONLY
    ENTRY
    EXPORT Reset_Handler

Reset_Handler

    LDR R0, =SRC    ;r0 is pointer to ith element
    LDR R1, =N1
```

```
Running with Code Size Limit: 32K
Load "C:\\Users\\aniket\\Desktop\\desktop\\sem4\\es\\labsbyme\\week3\\Objects\\week3.
```

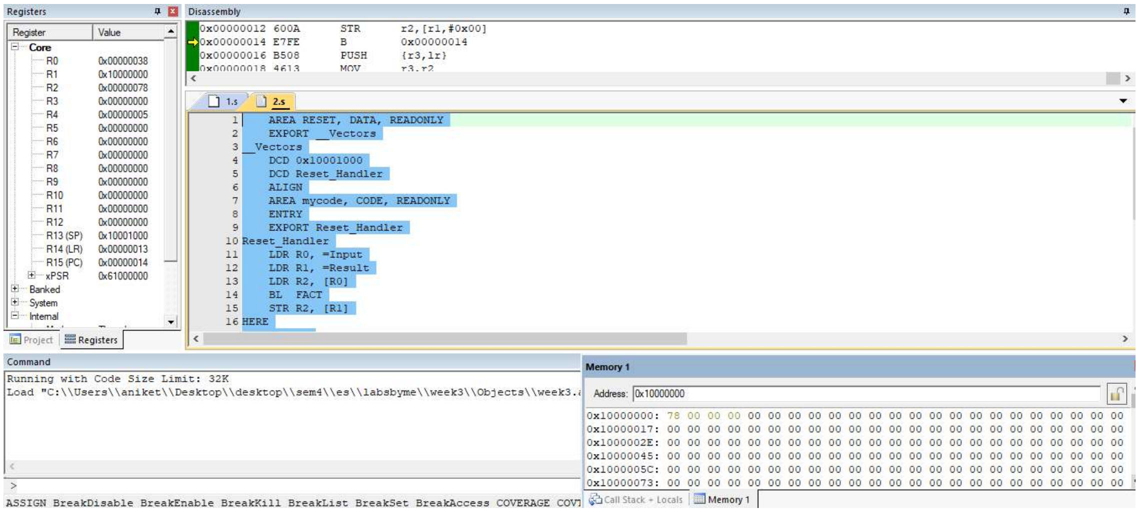
Address: 0x10000000

```
0x10000000: 10 00 00 00 63 00 00 00 32 00 00 00 19 00 00 00 28 00 00 00 39 00 00
0x10000017: 00 86 00 00 00 67 00 00 00 23 00 00 00 13 00 00 00 00 00 00 00 00 00
0x1000002E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000045: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1000005C: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000073: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Q2.Factorial using recursion

```
        AREA RESET, DATA, READONLY

        EXPORT __Vectors

__Vectors

        DCD 0x10001000

        DCD Reset_Handler

        ALIGN

        AREA mycode, CODE, READONLY

        ENTRY

        EXPORT Reset_Handler

Reset_Handler

        LDR R0, =Input

        LDR R1, =Result

        LDR R2, [R0]

        BL      FACT

        STR R2, [R1]

HERE

        B HERE

FACT PUSH{R3, LR}

        MOV R3, R2

        CMP R2, #0

        BNE DOWN

        MOV R2, #1

        B DOWN1

DOWN SUB R2, #1

        BL FACT

        MOV R4, R3

        MUL R2, R4

DOWN1 POP{R3, LR}

        BX LR

Input DCD 5
```

```
        AREA mydata, DATA, READWRITE
Result DCD 0
        END
```



Registers

Register	Value
Core	
R0	0x00000038
R1	0x10000000
R2	0x00000078
R3	0x00000000
R4	0x00000005
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x10001000
R14 (LR)	0x00000013
R15 (PC)	0x00000014
xPSR	0x61000000
Banked	
System	
Internal	

Disassembly

```
0x00000012 600A      STR      r2,[r1,#0x00]
0x00000014 E7FE      B        0x00000014
0x00000016 B508      PUSH     {r3,lr}
0x00000018 4613      MOV      r3,r2
```

2.s

```
    AREA RESET, DATA, READONLY
    EXPORT __Vectors
__Vectors
    DCD 0x10001000
    DCD Reset_Handler
    ALIGN
    AREA mycode, CODE, READONLY
    ENTRY
    EXPORT Reset_Handler
Reset_Handler
    LDR R0, =Input
    LDR R1, =Result
    LDR R2, [R0]
    BL  FACT
    STR R2, [R1]
HERE
```

Command

```
Running with Code Size Limit: 32K
Load "C:\\Users\\aniket\\Desktop\\desktop\\sem4\\es\\labsbyme\\week3\\Objects\\week3.
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE COV
```

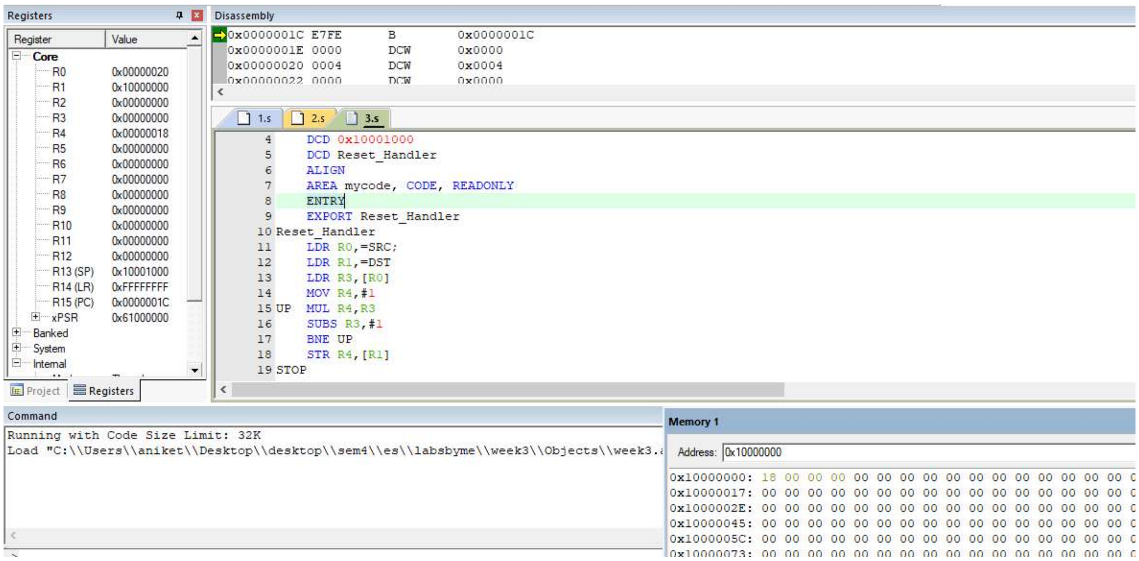
Memory 1

Address: 0x10000000

```
0x10000000: 78 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000017: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1000002E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000045: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1000005C: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000073: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Q3.Factorial iteratively

```
        AREA RESET, DATA, READONLY

        EXPORT __Vectors

__Vectors

        DCD 0x10001000

        DCD Reset_Handler

        ALIGN

        AREA mycode, CODE, READONLY

        ENTRY

        EXPORT Reset_Handler

Reset_Handler

        LDR R0,=SRC;

        LDR R1,=DST

        LDR R3,[R0]

        MOV R4,#1

UP  MUL R4,R3

        SUBS R3,#1

        BNE UP

        STR R4,[R1]

STOP

        B STOP


SRC DCD 4


        AREA mydata, DATA, READWRITE

DST DCD 0

        END
```



Registers

Register	Value
Core	
R0	0x00000020
R1	0x10000000
R2	0x00000000
R3	0x00000000
R4	0x00000018
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x10001000
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x0000001C
xPSR	0x61000000
Banked	
System	
Internal	

Disassembly

```
0x0000001C E7FE       B         0x0000001C
0x0000001E 0000       DCW       0x0000
0x00000020 0004       DCW       0x0004
0x00000022 0000       DCW       0x0000
```

```
    DCD 0x10001000
    DCD Reset_Handler
    ALIGN
    AREA mycode, CODE, READONLY
    ENTRY
    EXPORT Reset_Handler
Reset_Handler
    LDR R0,=SRC;
    LDR R1,=DST
    LDR R3,[R0]
    MOV R4,#1
UP  MUL R4,R3
    SUBS R3,#1
    BNE UP
    STR R4,[R1]
STOP
```

Command

```
Running with Code Size Limit: 32K
Load "C:\\Users\\aniket\\Desktop\\desktop\\sem4\\es\\labsbyme\\week3\\Objects\\week3.
```

Memory 1

Address: 0x10000000

```
0x10000000: 18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000017: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1000002E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000045: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1000005C: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000073: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```