Ayush Goyal

190905522 CSE D 62

ES Lab-5 (Week-5) – Sorting and Searching Programs

1) **Write an assembly program to sort an array using selection sort.**

CODE:

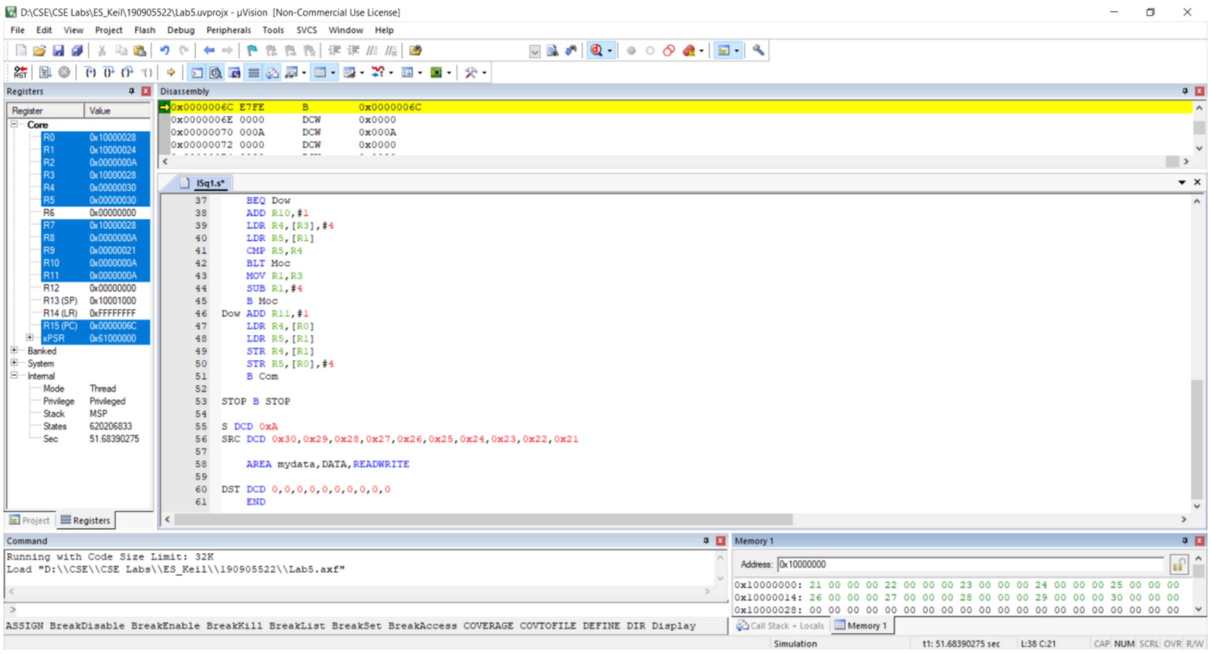
```
        AREA RESET,DATA,READONLY
        EXPORT __Vectors
__Vectors
        DCD 0x10001000
        DCD Reset_Handler
        ALIGN
        AREA mycode,CODE,READONLY
        ENTRY
        EXPORT Reset_Handler
Reset_Handler
        LDR R0, =SRC
        LDR R1, =S
        LDR R2,[R1]
        LDR R7, =DST
        MOV R8,#0
Up      CMP R8,R2
        BEQ Wod
        ADD R8,#1
        LDR R9,[R0],#4
        STR R9,[R7],#4
        B Up
Wod     LDR R0,=DST
        MOV R1, R0
        MOV R3,R0
        MOV R10,#0
        MOV R11,#0
Com     CMP R11, R2
        BEQ STOP
        ADD R3,R0,#4
        MOV R1,R0
        ADD R10,R11,#1
Moc     CMP R10,R2
        BEQ Dow
        ADD R10,#1
        LDR R4,[R3],#4
        LDR R5,[R1]
        CMP R5,R4
        BLT Moc
        MOV R1,R3
        SUB R1,#4
        B Moc
Dow     ADD R11,#1
        LDR R4,[R0]
        LDR R5,[R1]
        STR R4,[R1]
        STR R5,[R0],#4
        B Com

STOP B STOP
S DCD 0xA
SRC DCD 0x30,0x29,0x28,0x27,0x26,0x25,0x24,0x23,0x22,0x21

        AREA mydata,DATA,READWRITE

DST DCD 0,0,0,0,0,0,0,0,0,0
        END
```

OUTPUT:



D:\CSE\CSE Labs\ES_Keil\190905522\Lab5.uvprojx - µVision [Non-Commercial Use License]

Register	Value
Core	
R0	0x10000028
R1	0x10000024
R2	0x0000000A
R3	0x10000028
R4	0x00000030
R5	0x00000030
R6	0x00000000
R7	0x10000028
R8	0x0000000A
R9	0x00000021
R10	0x0000000A
R11	0x0000000A
R12	0x00000000
R13 (SP)	0x10001000
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x0000006C
xPSR	0x61000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	620206833
Sec	51.68390275

Disassembly

```
0x0000006C E7FE      B        0x0000006C
0x0000006E 0000      DCW      0x0000
0x00000070 000A      DCW      0x000A
0x00000072 0000      DCW      0x0000
```

l5q1.s*

```
    BEQ Dow
    ADD R10,#1
    LDR R4,[R3],#4
    LDR R5,[R1]
    CMP R5,R4
    BLT Moc
    MOV R1,R3
    SUB R1,#4
    B Moc
Dow ADD R11,#1
    LDR R4,[R0]
    LDR R5,[R1]
    STR R4,[R1]
    STR R5,[R0],#4
    B Com

STOP B STOP

S DCD 0xA
SRC DCD 0x30,0x29,0x28,0x27,0x26,0x25,0x24,0x23,0x22,0x21

    AREA mydata,DATA,READWRITE

DST DCD 0,0,0,0,0,0,0,0,0,0
    END
```

Command

```
Running with Code Size Limit: 32K
Load "D:\\CSE\\CSE Labs\\ES_Keil\\190905522\\Lab5.axf"
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE COVTOFILE DEFINE DIR Display
```

Memory 1

Address: 0x10000000

```
0x10000000: 21 00 00 00 22 00 00 00 23 00 00 00 24 00 00 00 25 00 00 00
0x10000014: 26 00 00 00 27 00 00 00 28 00 00 00 29 00 00 00 30 00 00 00
0x10000028: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Simulation   t1: 51.68390275 sec   L:38 C:21

2) Write an assembly program to find the factorial of an unsigned number using recursion.

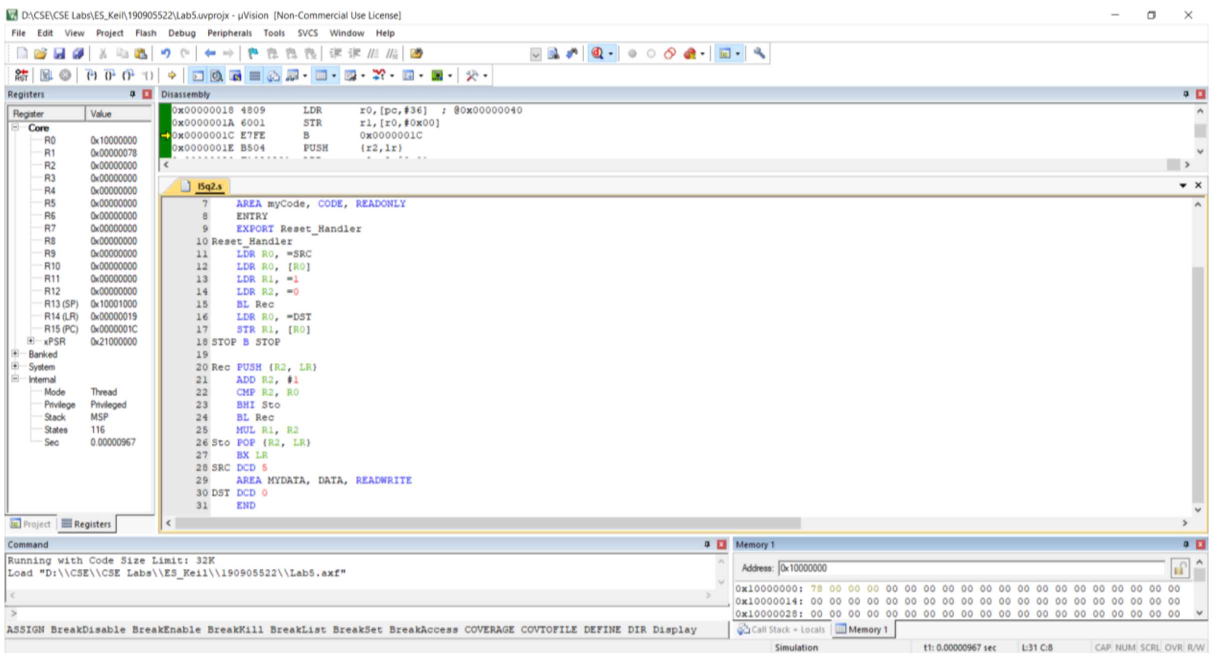
CODE:

```
        AREA RESET, CODE, READONLY
        EXPORT __Vectors
__Vectors
        DCD 0x10001000
        DCD Reset_Handler

        AREA myCode, CODE, READONLY
        ENTRY
        EXPORT Reset_Handler
Reset_Handler
        LDR R0, =SRC
        LDR R0, [R0]
        LDR R1, =1
        LDR R2, =0
        BL Rec
        LDR R0, =DST
        STR R1, [R0]
STOP B STOP

Rec     PUSH {R2, LR}
        ADD R2, #1
        CMP R2, R0
        BHI Sto
        BL Rec
        MUL R1, R2
Sto     POP {R2, LR}
        BX LR
SRC DCD 5
        AREA MYDATA, DATA, READWRITE
DST DCD 0
        END
```

OUTPUT:



D:\CSE\CSE Labs\ES_Keil\190905522\Lab5.uvprojx - µVision [Non-Commercial Use License]

Registers

Register	Value
Core	
R0	0x10000000
R1	0x00000078
R2	0x00000000
R3	0x00000000
R4	0x00000000
R5	0x00000000
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x10001000
R14 (LR)	0x00000019
R15 (PC)	0x0000001C
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	116
Sec	0.00000967

Disassembly

```
0x00000018 4809      LDR     r0,[pc,#36]  ; @0x00000040
0x0000001A 6001      STR     r1,[r0,#0x00]
0x0000001C E7FE      B       0x0000001C
0x0000001E B504      PUSH    {r2,lr}
```

l5q2.s

```
    AREA myCode, CODE, READONLY
    ENTRY
    EXPORT Reset_Handler
Reset_Handler
    LDR R0, =SRC
    LDR R0, [R0]
    LDR R1, =1
    LDR R2, =0
    BL Rec
    LDR R0, =DST
    STR R1, [R0]
STOP B STOP

Rec PUSH {R2, LR}
    ADD R2, #1
    CMP R2, R0
    BHI Sto
    BL Rec
    MUL R1, R2
Sto POP {R2, LR}
    BX LR
SRC DCD 5
    AREA MYDATA, DATA, READWRITE
DST DCD 0
    END
```

Command

```
Running with Code Size Limit: 32K
Load "D:\\CSE\\CSE Labs\\ES_Keil\\190905522\\Lab5.axf"
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE COVTOFILE DEFINE DIR Display
```

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
0x10000014: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000028: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Simulation   t1: 0.00000967 sec   L:31 C:8

3) **Write an assembly program to search an element in an array of ten 32-bit numbers using linear search.**

CODE:

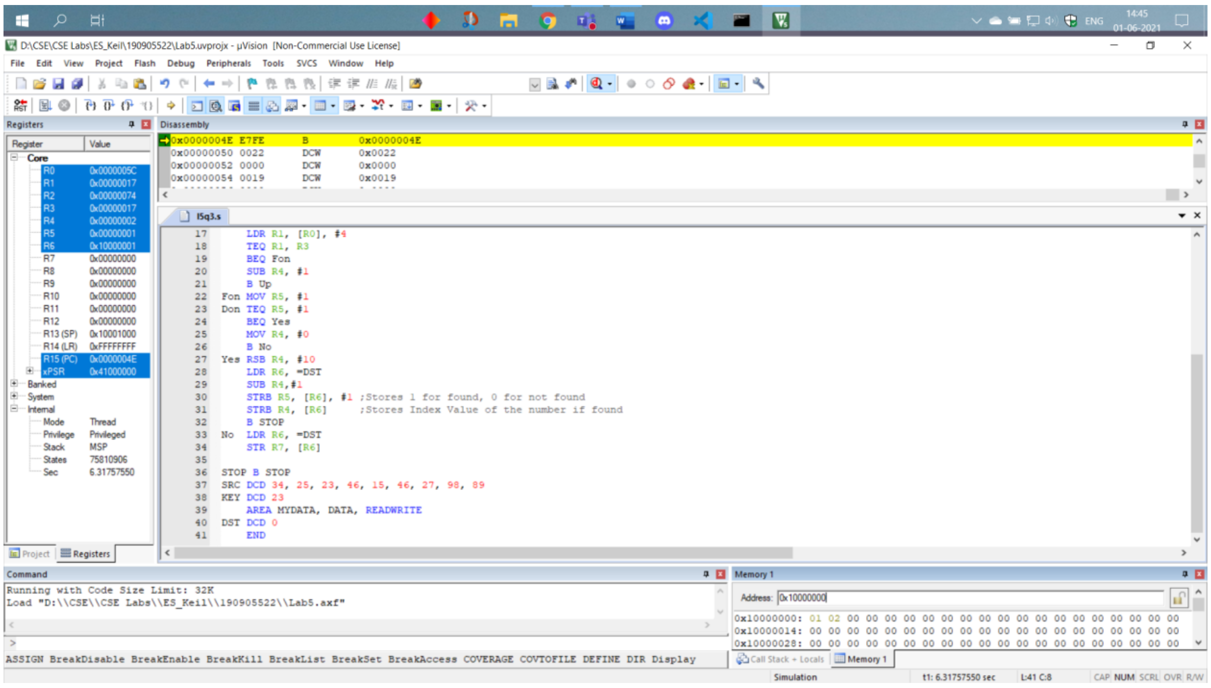
```
        AREA RESET, CODE, READONLY
        EXPORT __Vectors
__Vectors
        DCD 0x10001000
        DCD Reset_Handler

        AREA myCode, CODE, READONLY
        ENTRY
        EXPORT Reset_Handler
Reset_Handler
        LDR R0, =SRC
        LDR R2, =KEY
        MOV R4, #9
        LDR R3, [R2]
```

```
Up      TEQ R4, #0
        BEQ Don
        LDR R1, [R0], #4
        TEQ R1, R3
        BEQ Fon
        SUB R4, #1
        B Up
Fon     MOV R5, #1
Don     TEQ R5, #1
        BEQ Yes
        MOV R4, #0
        B No
Yes     RSB R4, #10
        LDR R6, =DST
        SUB R4,#1
        STRB R5, [R6], #1 ;Stores 1 for found, 0 for not found
        STRB R4, [R6]     ;Stores Index Value of the number if found
        B STOP
No      LDR R6, =DST
        STR R7, [R6]

STOP B STOP
SRC     DCD 34, 25, 23, 46, 15, 46, 27, 98, 89
KEY DCD 23
        AREA MYDATA, DATA, READWRITE
DST DCD 0
        END
```

OUTPUT:



D:\CSE\CSE Labs\ES_Keil\190905522\Lab5.uvprojx - µVision [Non-Commercial Use License]

```
Register     Value
Core
  R0         0x0000005C
  R1         0x00000017
  R2         0x00000074
  R3         0x00000017
  R4         0x00000002
  R5         0x00000001
  R6         0x10000001
  R7         0x00000000
  R8         0x00000000
  R9         0x00000000
  R10        0x00000000
  R11        0x00000000
  R12        0x00000000
  R13 (SP)   0x10001000
  R14 (LR)   0xFFFFFFFF
  R15 (PC)   0x0000004E
  xPSR       0x41000000
Banked
System
Internal
  Mode       Thread
  Privilege  Privileged
  Stack      MSP
  States     75810906
  Sec        6.31757550
```

```
Disassembly
0x0000004E E7FE      B         0x0000004E
0x00000050 0022      DCW       0x0022
0x00000052 0000      DCW       0x0000
0x00000054 0019      DCW       0x0019
```

```
l5q3.s
     LDR R1, [R0], #4
     TEQ R1, R3
     BEQ Fon
     SUB R4, #1
     B Up
Fon MOV R5, #1
Don TEQ R5, #1
     BEQ Yes
     MOV R4, #0
     B No
Yes RSB R4, #10
     LDR R6, =DST
     SUB R4,#1
     STRB R5, [R6], #1 ;Stores 1 for found, 0 for not found
     STRB R4, [R6]     ;Stores Index Value of the number if found
     B STOP
No   LDR R6, =DST
     STR R7, [R6]

STOP B STOP
SRC DCD 34, 25, 23, 46, 15, 46, 27, 98, 89
KEY DCD 23
     AREA MYDATA, DATA, READWRITE
DST DCD 0
     END
```

```
Command
Running with Code Size Limit: 32K
Load "D:\\CSE\\CSE Labs\\ES_Keil\\190905522\\Lab5.axf"
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE COVTOFILE DEFINE DIR Display
```

```
Memory 1
Address: 0x10000000
0x10000000: 01 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000014: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x10000028: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Simulation t1: 6.31757550 sec L:41 C:8

4) **Assume that ten 32-bit numbers are stored in registers R1-R10. Sort these numbers in the empty ascending stack using selection sort and store the sorted array back into the registers. Use STM and LDMDB instructions wherever necessary.**

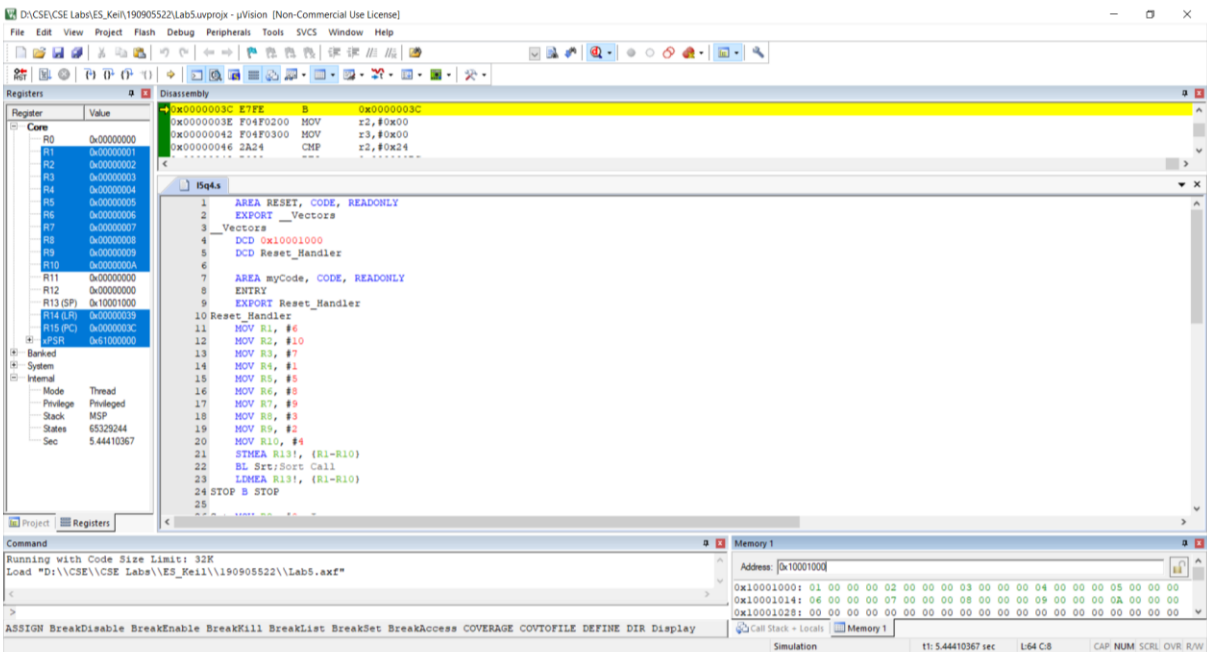
CODE:

```
        AREA RESET, CODE, READONLY
        EXPORT __Vectors
__Vectors
        DCD 0x10001000
        DCD Reset_Handler

        AREA myCode, CODE, READONLY
        ENTRY
        EXPORT Reset_Handler
Reset_Handler
        MOV R1, #6
        MOV R2, #10
        MOV R3, #7
        MOV R4, #1
        MOV R5, #5
        MOV R6, #8
        MOV R7, #9
        MOV R8, #3
        MOV R9, #2
        MOV R10, #4
        STMEA R13!, {R1-R10}
        BL Srt ; Sort Call
        LDMEA R13!, {R1-R10}
STOP B STOP

Srt     MOV R2, #0 ; I
        MOV R3, #0 ; J
Top     CMP R2, #36
        BEQ Sto
        SUB R8, R13, R2
        SUB R8, #4
        LDR R4, [R8]
        MOV R7, R2
        MOV R3, R2
        ADD R3, #4
UP      CMP R3, #40
        BEQ Hel
        SUB R8, R13, R3
        SUB R8, #4
        LDR R5, [R8]
        CMP R4, R5
        BCS Leh
        SUB R8, R13, R3
        SUB R8, #4
        LDR R4, [R8]
        MOV R7, R3
Leh     ADD R3, #4
        B UP
Hel     SUB R8, R13, R7
        SUB R8, #4
        LDR R4, [R8]
        SUB R8, R13, R2
        SUB R8, #4
        LDR R5, [R8]
        SUB R8, R13, R7
        SUB R8, #4
        STR R5, [R8]
        SUB R8, R13, R2
        SUB R8, #4
        STR R4, [R8]
        ADD R2, #4
        B Top
Sto     BX LR
        END
```

OUTPUT:


D:\CSE\CSE Labs\ES_Keil\190905522\Lab5.uvprojx - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Disassembly
l5q4.s
Project
Registers
Command
Memory 1
Call Stack + Locals
Simulation
t1: 5.44410367 sec
L:64 C:8
CAP NUM SCRL OVR R/W

```
Register      Value
Core
  R0          0x00000000
  R1          0x00000001
  R2          0x00000002
  R3          0x00000003
  R4          0x00000004
  R5          0x00000005
  R6          0x00000006
  R7          0x00000007
  R8          0x00000008
  R9          0x00000009
  R10         0x0000000A
  R11         0x00000000
  R12         0x00000000
  R13 (SP)    0x10001000
  R14 (LR)    0x00000039
  R15 (PC)    0x0000003C
  xPSR        0x61000000
Banked
System
Internal
  Mode        Thread
  Privilege   Privileged
  Stack       MSP
  States      65329244
  Sec         5.44410367
```

```
0x0000003C E7FE      B         0x0000003C
0x0000003E F04F0200  MOV       r2,#0x00
0x00000042 F04F0300  MOV       r3,#0x00
0x00000046 2A24      CMP       r2,#0x24
```

```
    AREA RESET, CODE, READONLY
    EXPORT __Vectors
__Vectors
    DCD 0x10001000
    DCD Reset_Handler

    AREA myCode, CODE, READONLY
    ENTRY
    EXPORT Reset_Handler
Reset_Handler
    MOV R1, #6
    MOV R2, #10
    MOV R3, #7
    MOV R4, #1
    MOV R5, #5
    MOV R6, #8
    MOV R7, #9
    MOV R8, #3
    MOV R9, #2
    MOV R10, #4
    STMEA R13!, {R1-R10}
    BL Srt;Sort Call
    LDMEA R13!, {R1-R10}
STOP B STOP

```

```
Running with Code Size Limit: 32K
Load "D:\\CSE\\CSE Labs\\ES_Keil\\190905522\\Lab5.axf"
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE COVTOFILE DEFINE DIR Display
```

```
Address: 0x10001000
0x10001000: 01 00 00 00 02 00 00 00 03 00 00 00 04 00 00 00 05 00 00 00
0x10001014: 06 00 00 00 07 00 00 00 08 00 00 00 09 00 00 00 0A 00 00 00
0x10001028: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

THE END