LAB NO : 5 DATE : 05/02/2025

Title : PROGRAMS ON SORTING

# Lab Exercise 1: Write an ARM assembly language program to sort an array using selection sort.

## Code :

## AREA RESET, DATA, READONLY

## EXPORT \_\_Vectors

## \_\_Vectors

## DCD 0x10001000 ; stack pointer value when stack is empty

## DCD Reset\_Handler ; reset vector

## ALIGN

## AREA mycode, CODE, READONLY

## ENTRY

## EXPORT Reset\_Handler

## Reset\_Handler

## LDR R0, =ARR

## MOV R1, #0 ;loop counter i

## ADD R7, R0, #36 ;index for last element

## 

## ;storing data in array (10 to 1 descending order)

## MOV R9, #10

## loop STR R9, [R0], #4

## SUBS R9, #1

## BNE loop

## LDR R0, =ARR

## 

## ;selection sort starts here

## loopI LDR R3, [R0]

## MOV R4, R0

## MOV R6, R0

## loopJ ADD R6, #4

## CMP R6, R7

## BGT exitloopJ

## LDR R5, [R6]

## CMP R5, R3

## BGT skip

## MOV R3, R5

## MOV R4, R6 ;storing index for smallest found

## skip B loopJ

## exitloopJ LDR R8, [R0]

## STR R3, [R0]

## STR R8, [R4]

## ADD R0, #4

## CMP R0, R7

## BEQ STOP

## B loopI

## STOP B STOP

## AREA mydata, DATA, READWRITE

## ARR DCD 1

## END

## Output :

# Lab Exercise 2: Write an ARM assembly language program to find the factorial of an unsigned number using recursion

## 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, =N

LDR R1, [R0]

MOV R2, #1

recur

BL fact

CMP R1, #0

BEQ exit

B recur

exit

LDR R5, =FACT

STR R2, [R5]

B stop

fact

MUL R2, R2, R1

SUB R1, #1

CMP R1, #0

BNE fact

BX LR

stop

B stop

N DCD 3

AREA mydata, DATA, READWRITE

FACT DCD 1

END

## Output :

# Lab Exercise 3: Assume that ten 32-bit numbers are stored in registers R1-R10. Write an ARM assembly language program to 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, DATA, READONLY

EXPORT \_\_Vectors

\_\_Vectors

DCD 0x40001000 ; stack pointer value when stack is empty

DCD Reset\_Handler ; reset vector

ALIGN

AREA mycode, CODE, READONLY

ENTRY

EXPORT Reset\_Handler

Reset\_Handler

LDR R0, =ARR

;storing data in array (10 to 1 descending order)

MOV R9, #10

loop

STR R9, [R0], #4

SUBS R9, #1

BNE loop

LDR R0, =ARR

LDM R0, {R1-R10}

STM R13!, {R1-R10}

MOV R0, R13

SUB R0, #40

ADD R7, R0, #36 ;index for last element

;selection sort starts here

loopI

LDM R0, {R3}

MOV R4, R0

MOV R6, R0

loopJ

ADD R6, #4

CMP R6, R7

BGT exitloopJ

LDM R6, {R5}

CMP R5, R3

BGT skip

MOV R3, R5

MOV R4, R6 ;storing index for smallest found

skip

B loopJ

exitloopJ

LDM R0, {R8}

STM R0, {R3}

STM R4, {R8}

ADD R0, #4

CMP R0, R7

BEQ exit

B loopI

exit

SUB R0, #36

LDM R0, {R1-R10}

STOP

B STOP

AREA mydata, DATA, READWRITE

ARR DCD 1

END

## **Output :**

