

WEEK 1

Microprocessor and Computer Organisation Laboratory

UE21CS251B

4th Semester, Academic Year 2022-23

Date: 24-01-2023

Name: Nandan N	SRN: PES1UG21CS361	Section: F
----------------	--------------------	------------

Week# ____1____ Program Number: ____1____

TITLE:

- I. Code Screenshot
- II. Output Screen Shot (Register set)
- III. Output Screen Shot (memory set)

WEEK 1

1. Sum of n numbers

CODE:

```
.DATA
A: .WORD 10,20,30,40,50,60,70,80,90,100
SUM: .WORD 0

.TEXT
LDR R1, =A
LDR R2, =SUM
MOV R4, #0      ;INITIASATION
MOV R5, #1      ;COUNT REGISTER

LOOP:  LDR R3, [R1]
        ADD R4, R4, R3      ;ADD NEXT ELEMENT INT HE ARRAY

        ADD R1, R1, #4      ;INCREMENT THE POINTER TO THE NEXT DATA
        ADD R5, R5, #1      ;INCREMENT THE COUNT REGISTER
        CMP R5, #11        ;CHECK WHEATHER ALL NUMBERS ARE ADDED

        BNE LOOP           ;IF (R5 != 11), GO TO LOOP
        STR R4, [R2]
        SWI 0X011
```

Output:

The screenshot shows the ARMSim ARM Simulator interface. The main window displays the assembly code being executed. The RegistersView window on the left shows the state of registers R0 through R15. The MemoryView window on the right shows the memory contents, including the data array and the sum. The OutputView window at the bottom is empty.

RegistersView:

Register	Value
R0	:00000000
R1	:00001060
R2	:00001060
R3	:00000064
R4	:00000226
R5	:0000000b
R6	:00000000
R7	:00000000
R8	:00000000
R9	:00000000
R10 (s1)	:00000000
R11 (fp)	:00000000
R12 (ip)	:00000000
R13 (sp)	:00005400
R14 (lr)	:00000000
R15 (pc)	:0000102c

MemoryView:

Address	Value
00001060	00000226
00001084	81818181
000010A8	81818181
000010CC	81818181

WEEK 1

2. Block

CODE:

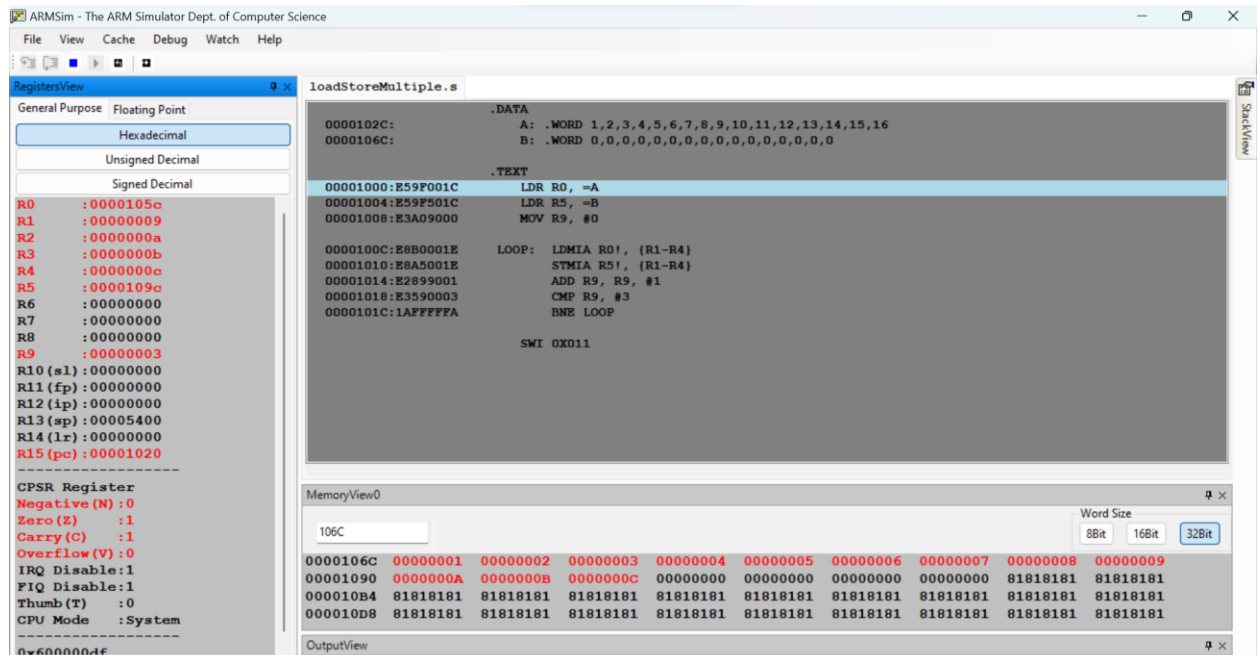
```
.DATA
A: .WORD 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16
B: .WORD 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

.TEXT
LDR R0, =A
LDR R5, =B
MOV R9, #0

LOOP: LDMIA R0!, {R1-R4}
      STMIA R5!, {R1-R4}
      ADD R9, R9, #1
      CMP R9, #3
      BNE LOOP

SWI 0X011
```

OUTPUT:



WEEK 1

3. Factorial

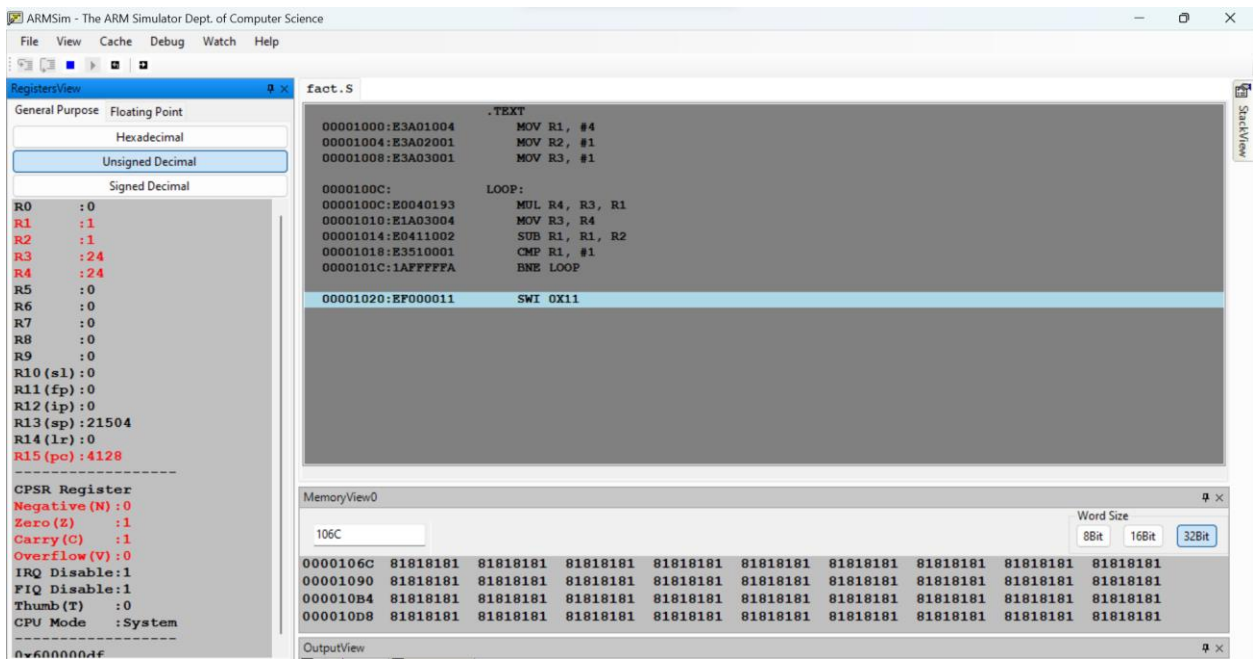
CODE:

```
.TEXT
MOV R1, #4      ;number to find factorial of
MOV R2, #1
MOV R3, #1

LOOP:
MUL R4, R3, R1  ;R4 = R3 * R1
MOV R3, R4      ;R3 = R4 ; R3 and R4 stores the factorial
SUB R1, R1, R2
CMP R1, #1
BNE LOOP

SWI 0X11
```

OUTPUT:



WEEK 1

4. Search

```
.DATA
ARR: .WORD 1,2,3,4,5,6,7,8,9,10
KEY: .WORD 11

.TEXT
LDR R1, =ARR
LDR R2, =KEY
LDR R3, [R2]
MOV R4, #0 ;COUNTER
MOV R5, #99 ;RESULT ;99 MEANS NOT FOUND

SUB R1, R1, #4
LOOP:
LDR R6, [R1, #4]! ;ADD 4 TO R1 AND LOAD THE CONTENTS AT THAT LOCATION TO
R6. R1 IS ALSO UPDATED
CMP R6, R3
BEQ FOUND

ADD R4, R4, #1
CMP R4, #9 ;LENGTH OF ARRAY IS 10
BNE LOOP

SWI 0X11

FOUND:
MOV R5, R4 ;STORE THE INDEX OF THE KEY IN R5
SWI 0X11
```

OUTPUT:

