Microprocessor and Computer Architecture

UE24CS251B

4th Semester, Academic Year 2025-26

Date:

Name: Harshit Chandak	SRN: PES2UG24CS185	Section C

LAB #____3_____ Program Number: ____1___

Title of the Program

Write an ALP to perform Addition for of two numbers of sizes 64 bit and 128 bit and save the result in register (reuse the register to store the result)

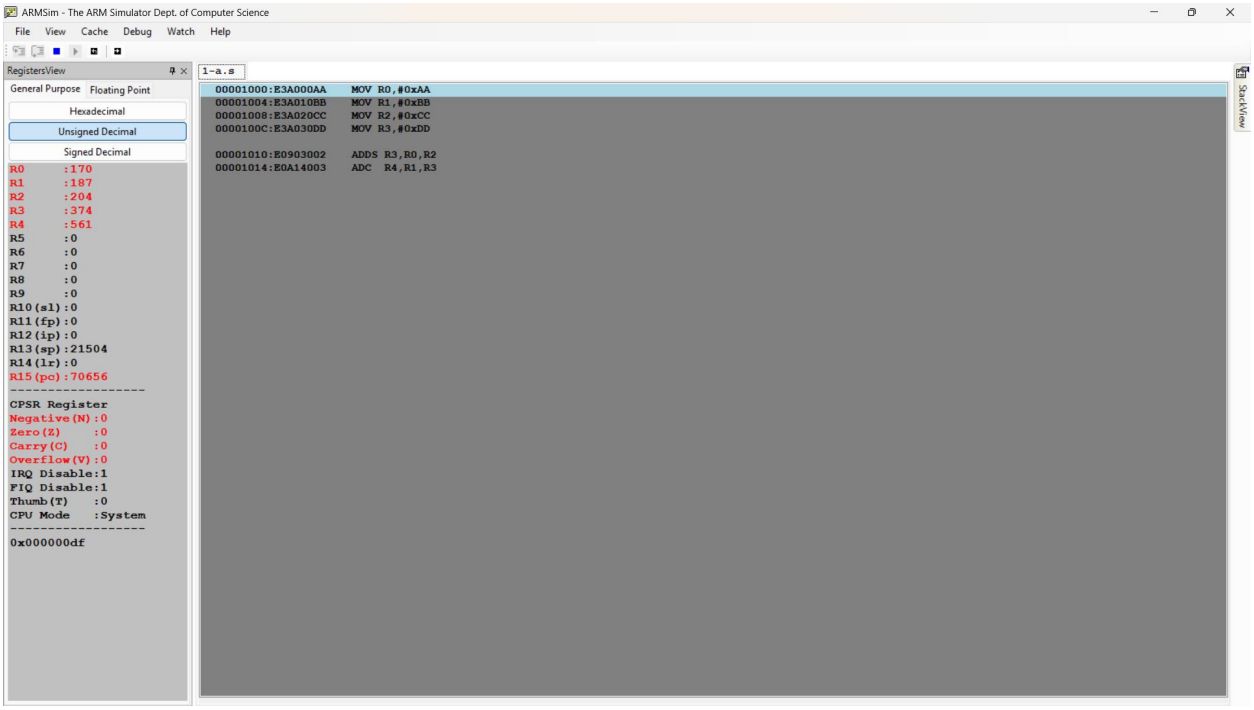
Hint:

R3 R2 R1 R0

\+ R5 R4

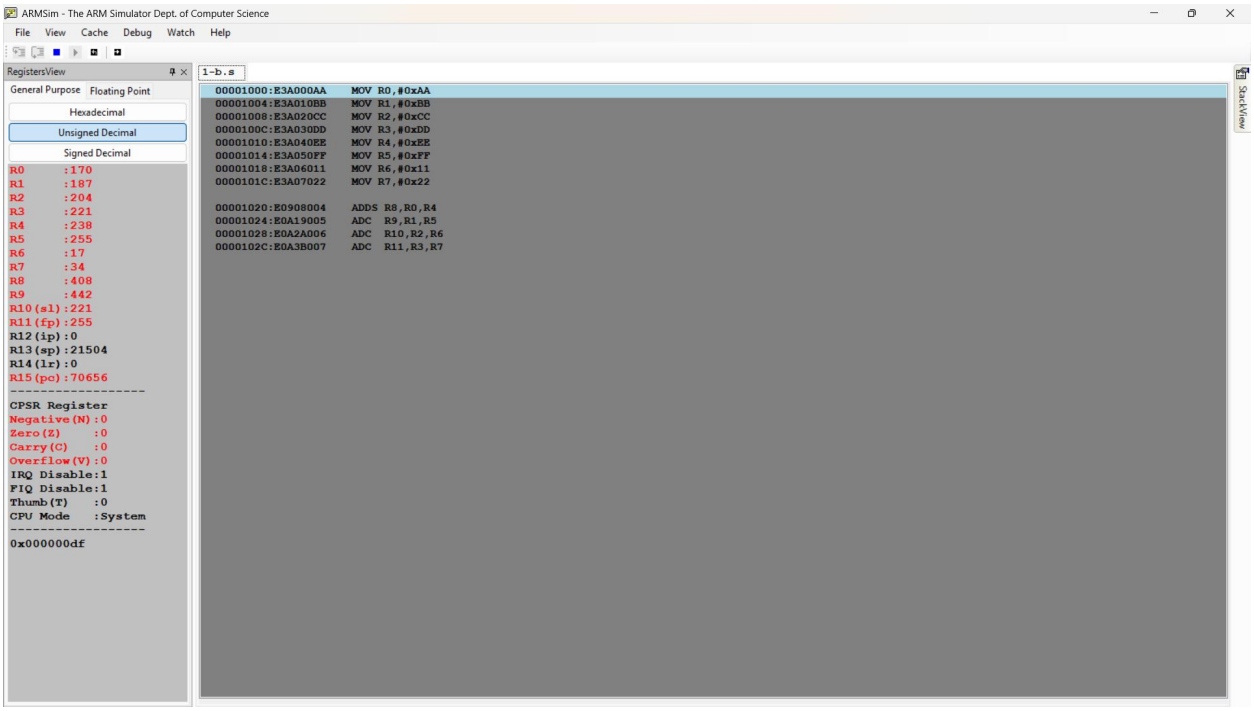
I. Typed ARM Assembly Code
II. Three Output Screen Shots (one for each case)
(Screenshot including Register Window,Output Window and Code Window)

```
MOV R0,#0xAA
MOV R1,#0xBB
MOV R2,#0xCC
MOV R3,#0xDD

ADDS R3,R0,R2
ADC  R4,R1,R3
```




```
MOV R0,#0xAA
MOV R1,#0xBB
MOV R2,#0xCC
MOV R3,#0xDD
MOV R4,#0xEE
MOV R5,#0xFF
MOV R6,#0x11
```

```
MOV R7,#0x22

ADDS R8,R0,R4
ADC  R9,R1,R5
ADC  R10,R2,R6
ADC  R11,R3,R7
```



Microprocessor and Computer Architecture

UE24CS251B

4th Semester, Academic Year 2025-26

Date:

Name: Harshit Chandak	SRN:PES2UG24CS185	Section

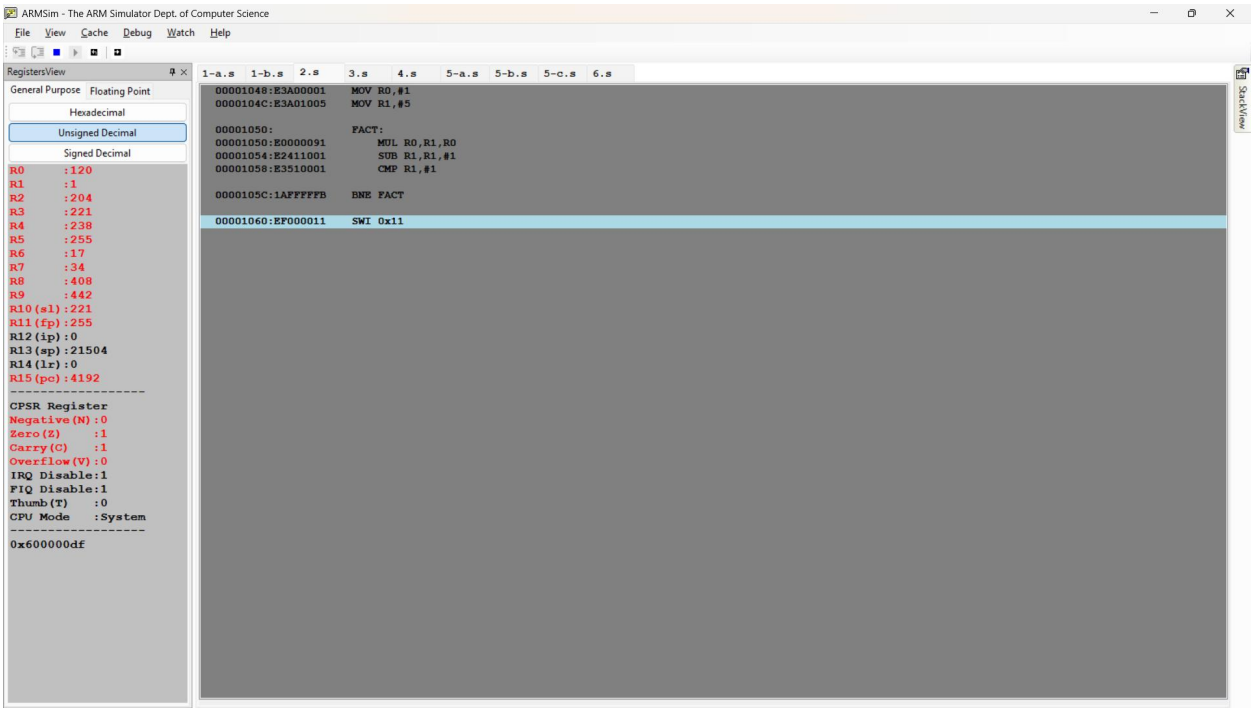
LAB #____3_____ Program Number: ____2___

Title of the Program

Write a program to find the factorial of a given number.

(One Screenshot including Register Window,Output Window and Code Window,Memory Window)

```
MOV R0,#1
MOV R1,#5

FACT:
    MUL R0,R1,R0
    SUB R1,R1,#1
    CMP R1,#1

BNE FACT

SWI 0x11
```



Microprocessor and Computer Architecture

UE24CS251B

4th Semester, Academic Year 2025-26

Date:

Name: Harshit Chandak	SRN: PES2UG24CS185	Section C

LAB #____3_____ Program Number: ____3___

Title of the Program

Write an ALP to find the number of zeroes, positive and negative numbers in a given array

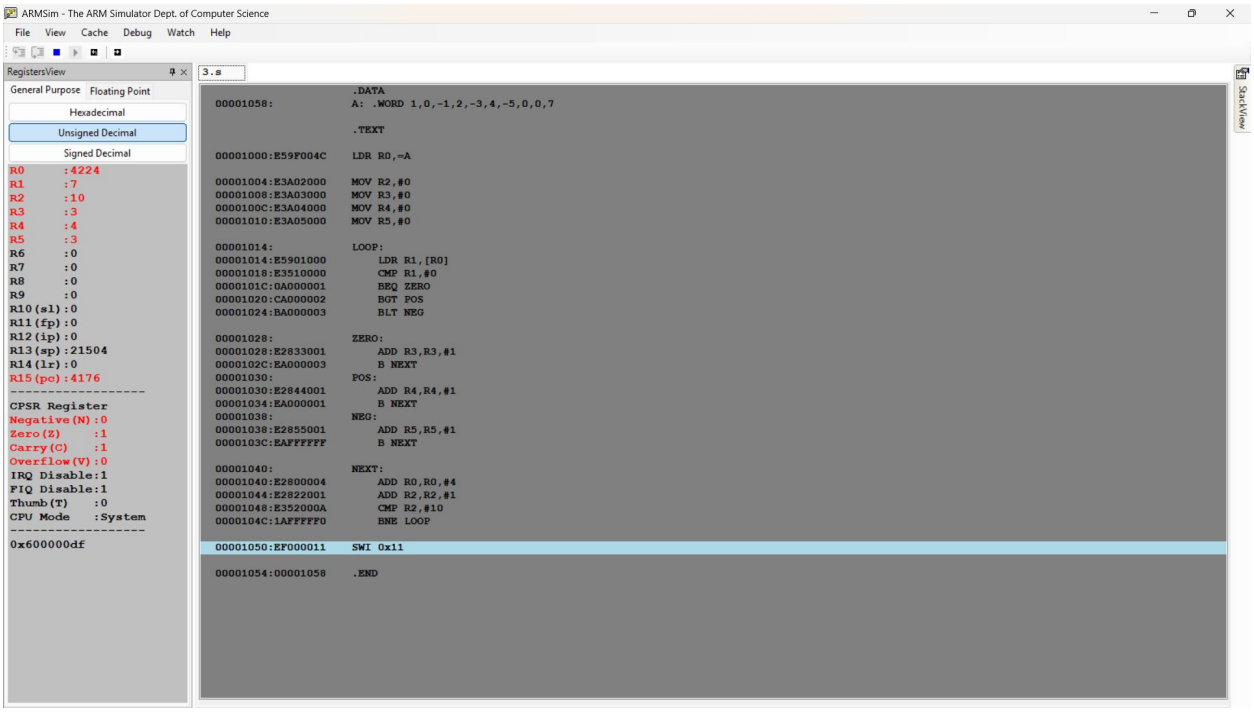
I. ARM Assembly Code

II. Output Screen Shots

(One Screenshot including Register Window,Memory Window and Code Window)

```
.DATA
A: .WORD 1,0,-1,2,-3,4,-5,0,0,7

.TEXT

LDR R0,=A

MOV R2,#0
MOV R3,#0
MOV R4,#0
MOV R5,#0

LOOP:
    LDR R1,[R0]
    CMP R1,#0
```

```
    BEQ ZERO
    BGT POS
    BLT NEG

ZERO:
    ADD R3,R3,#1
    B NEXT
POS:
    ADD R4,R4,#1
    B NEXT
NEG:
    ADD R5,R5,#1
    B NEXT

NEXT:
    ADD R0,R0,#4
    ADD R2,R2,#1
    CMP R2,#10
    BNE LOOP

SWI 0x11

.END
```



Microprocessor and Computer Architecture

UE22CS251B

4th Semester, Academic Year 2023-24

Date:

Name: Harshit Chandak	SRN: PES2UG24CS185	Section C

LAB #____3_____ Program Number: ____4___

Title of the Program

Write a program to perform 3X3 matrix addition.

I. ARM Assembly Code

II. Output Screen Shots

(One Screenshot including Register Window,Memory Window and Code Window)

```
.DATA
A: .WORD 1,2,3,4,5,6,7,8,9
B: .WORD 1,2,3,4,5,6,7,8,9
C: .WORD 0,0,0,0,0,0,0,0,0

.TEXT
```

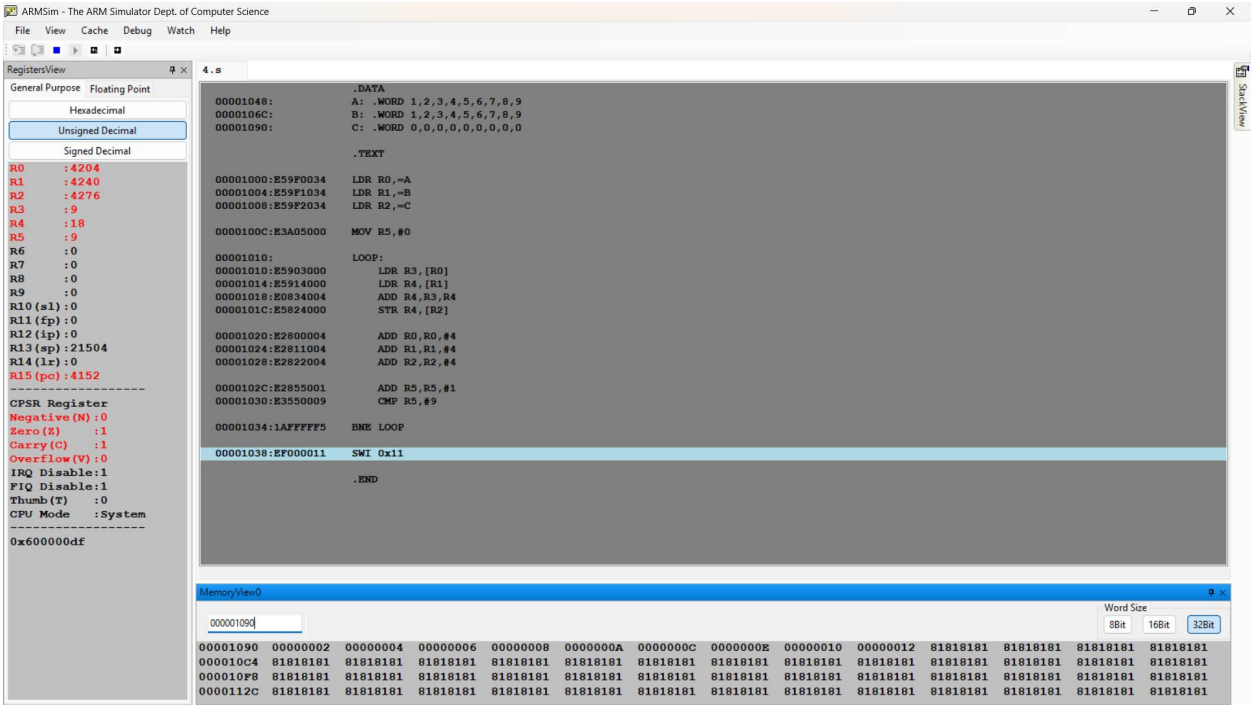
```
LDR R0,=A
LDR R1,=B
LDR R2,=C
```

```
MOV R5,#0
```

```
LOOP:
    LDR R3,[R0]
    LDR R4,[R1]
    ADD R4,R3,R4
    STR R4,[R2]
```

```
    ADD R0,R0,#4
    ADD R1,R1,#4
    ADD R2,R2,#4
```

```
    ADD R5,R5,#1
    CMP R5,#9

BNE LOOP

SWI 0x11

.END
```



Microprocessor and Computer Architecture

UE22CS251B

4th Semester, Academic Year 2023-24

Date:

Name: Harshit Chandak	SRN: PES2UG24CS185	Section C

LAB #____3_____ Assignment Question 1

Title of the Program

I. Write a program in ARM7TDMI-ISA to find the sum of N data items at alternate [odd or even positions] locations in the memory. Store the result in the memory location.
   a) Use Pre-indexing addressing mode
   b) Use post-indexing addressing mode
   c) Use Auto-indexing addressing mode

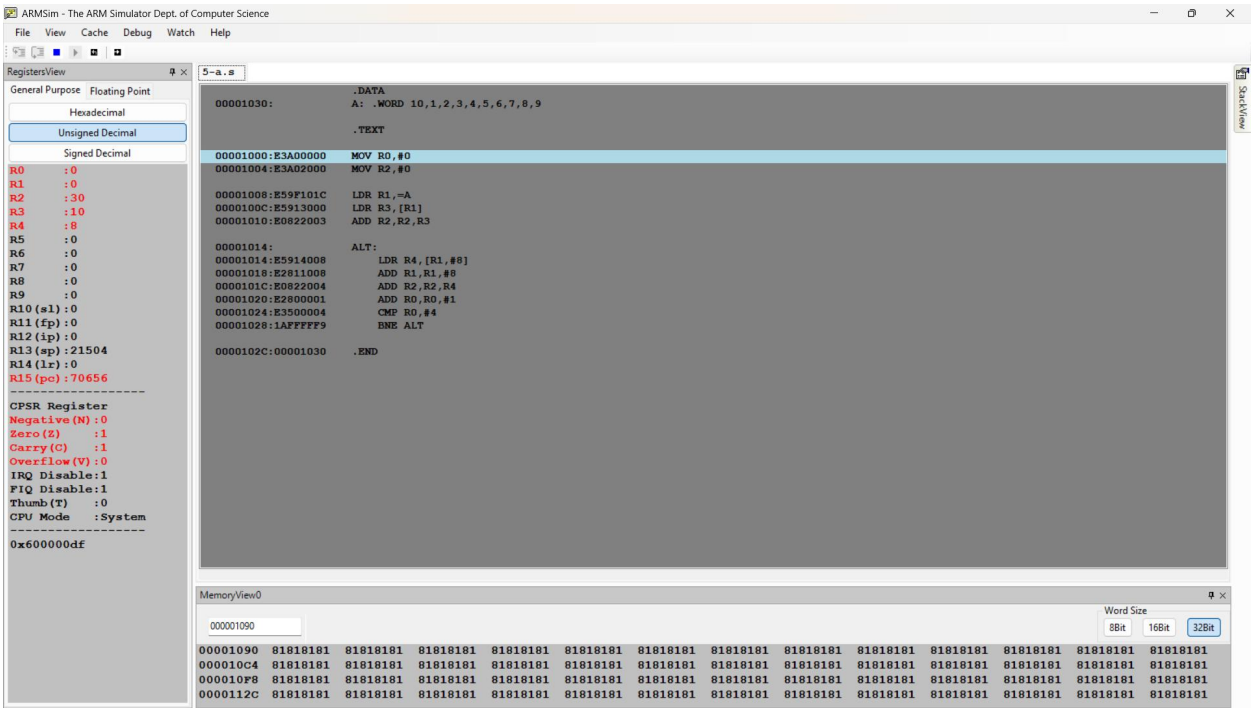
II. ARM Assembly Code

III. Output Screen Shots

(One Screenshot for each case including Register Window,Memory Window and Code Window)

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

```
.TEXT

MOV R0,#0
MOV R2,#0

LDR R1,=A
LDR R3,[R1]
ADD R2,R2,R3

ALT:
    LDR R4,[R1,#8]
    ADD R1,R1,#8
    ADD R2,R2,R4
    ADD R0,R0,#1
    CMP R0,#4
    BNE ALT

.END
```


ARMSim - The ARM Simulator Dept. of Computer Science
File View Cache Debug Watch Help
RegistersView
General Purpose Floating Point
Hexadecimal
Unsigned Decimal
Signed Decimal
R0 :0
R1 :0
R2 :30
R3 :10
R4 :8
R5 :0
R6 :0
R7 :0
R8 :0
R9 :0
R10(sl):0
R11(fp):0
R12(ip):0
R13(sp):21504
R14(lr):0
R15(pc):70656
CPSR Register
Negative(N):0
Zero(Z) :1
Carry(C) :1
Overflow(V):0
IRQ Disable:1
FIQ Disable:1
Thumb(T) :0
CPU Mode :System
0x600000df
5-a.s
00001030: .DATA
A: .WORD 10,1,2,3,4,5,6,7,8,9
.TEXT
00001000:E3A00000 MOV R0,#0
00001004:E3A02000 MOV R2,#0
00001008:E59F101C LDR R1,=A
0000100C:E5913000 LDR R3,[R1]
00001010:E0822003 ADD R2,R2,R3
00001014: ALT:
00001014:E5914008 LDR R4,[R1,#8]
00001018:E2811008 ADD R1,R1,#8
0000101C:E0822004 ADD R2,R2,R4
00001020:E2800001 ADD R0,R0,#1
00001024:E3500004 CMP R0,#4
00001028:1AFFFFF9 BNE ALT
0000102C:00001030 .END
StackView
MemoryView0
000001090
Word Size
8Bit 16Bit 32Bit
00001090 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010C4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010F8 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000112C 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

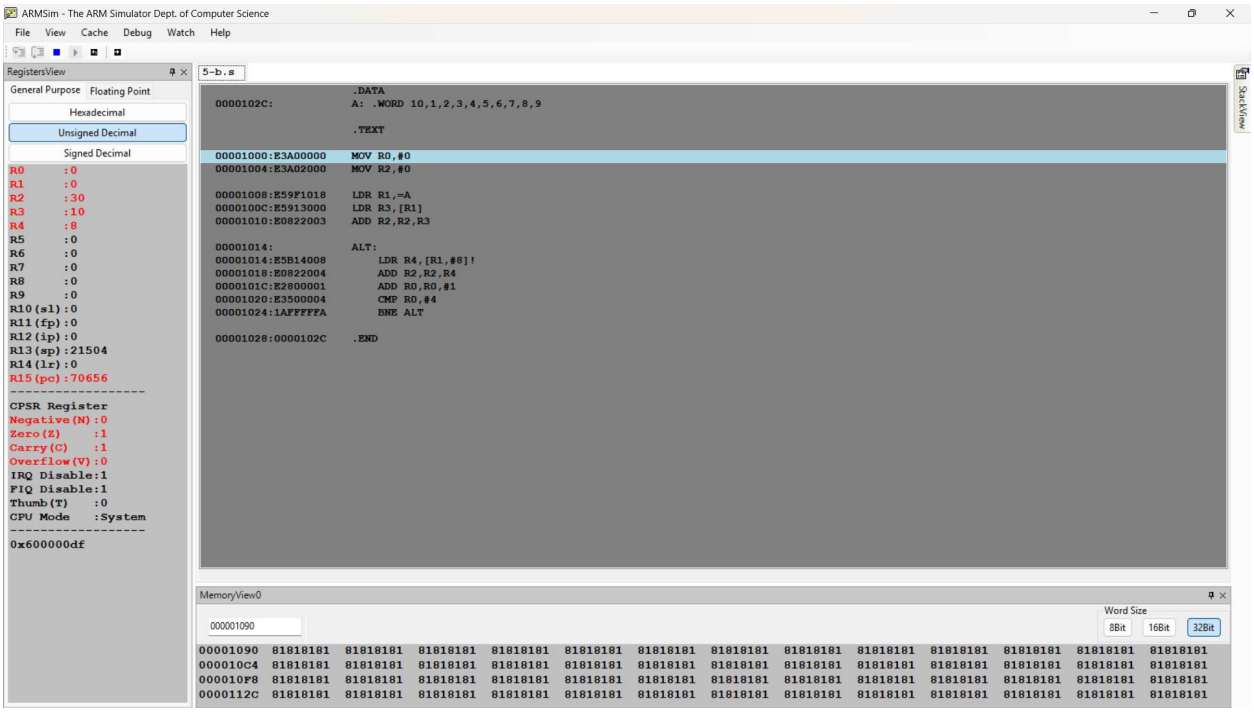
```
.DATA
A: .WORD 10,1,2,3,4,5,6,7,8,9

.TEXT
```

```
MOV R0,#0
MOV R2,#0
```

```
LDR R1,=A
LDR R3,[R1]
ADD R2,R2,R3
```

```
ALT:
    LDR R4,[R1,#8]!
    ADD R2,R2,R4
    ADD R0,R0,#1
    CMP R0,#4
    BNE ALT
```

```
.END
```




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

.TEXT
```

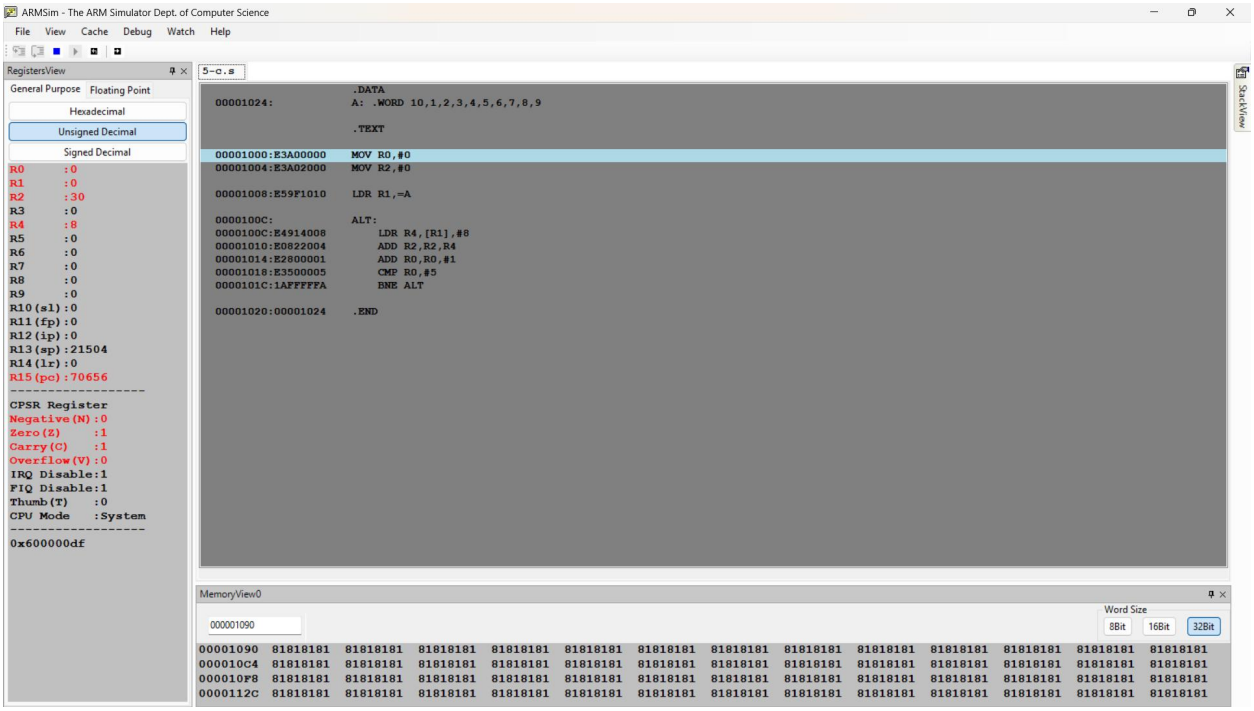
```
MOV R0,#0
MOV R2,#0
```

```
LDR R1,=A
```

```
ALT:
    LDR R4,[R1],#8
    ADD R2,R2,R4
    ADD R0,R0,#1
    CMP R0,#5
```

```
    BNE ALT
```

```
.END
```



Microprocessor and Computer Architecture

UE22CS251B

4th Semester, Academic Year 2023-24

Date:

Name: Harshit Chandak	SRN: PES2UG24CS185	Section C

LAB #____3_____

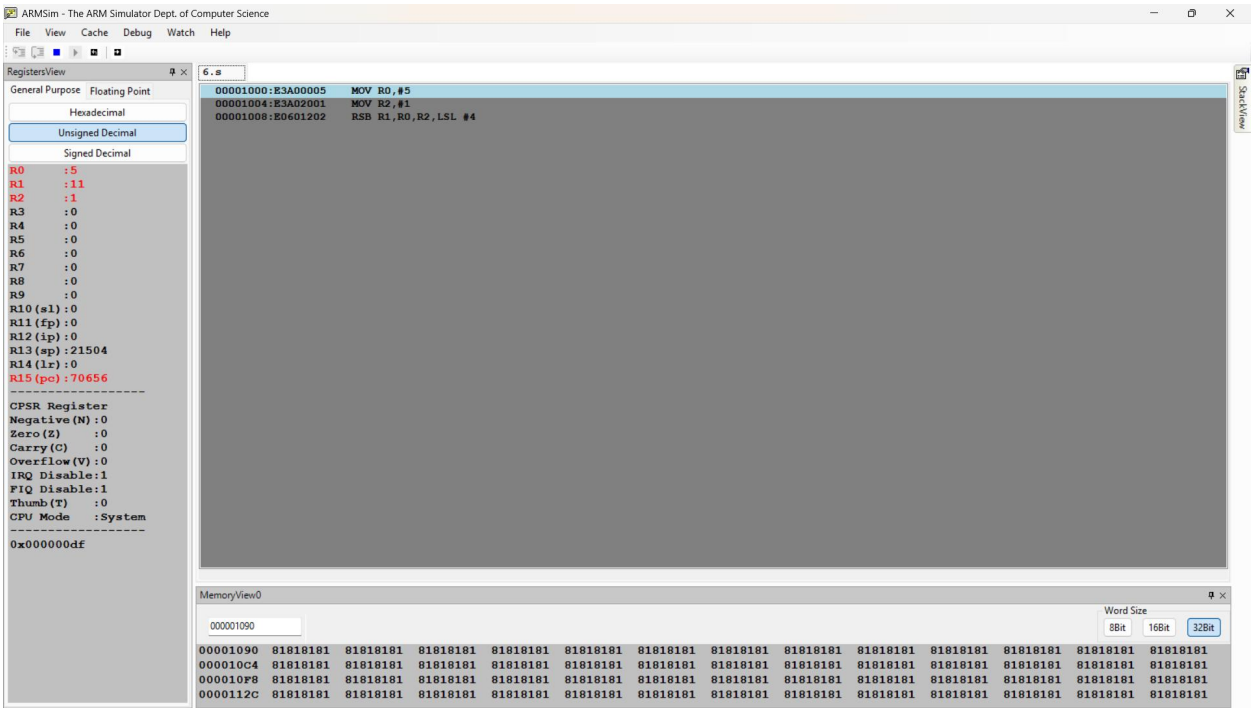
Assignment Question 2

Title of the Program

i. Write an ALP to perform 2's complement, Only using MOV and RSB instruction.
   I. ARM Assembly Code
   II. Output Screen Shots

   (One Screenshot including Register Window,Memory Window and Code Window)

```
MOV R0,#5
MOV R2,#1
RSB R1,R0,R2,LSL #4
```



Disclaimer:

- The programs and output submitted is duly written, verified and executed by me.
- I have not copied from any of my peers nor from the external resource such as internet.
- If found plagiarized, I will abide with the disciplinary action of the University.

Signature:

Name:

SRN:

Section:

Date: