

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 2 Program Number: 1

Title of the Program:

Based on the value of the number in R0, Write an ALP to store 1 in R1 if R0 is zero, Store 2 in R1 if R0 is positive, Store 3 in R1 if R0 is negative. (Program shown in class)

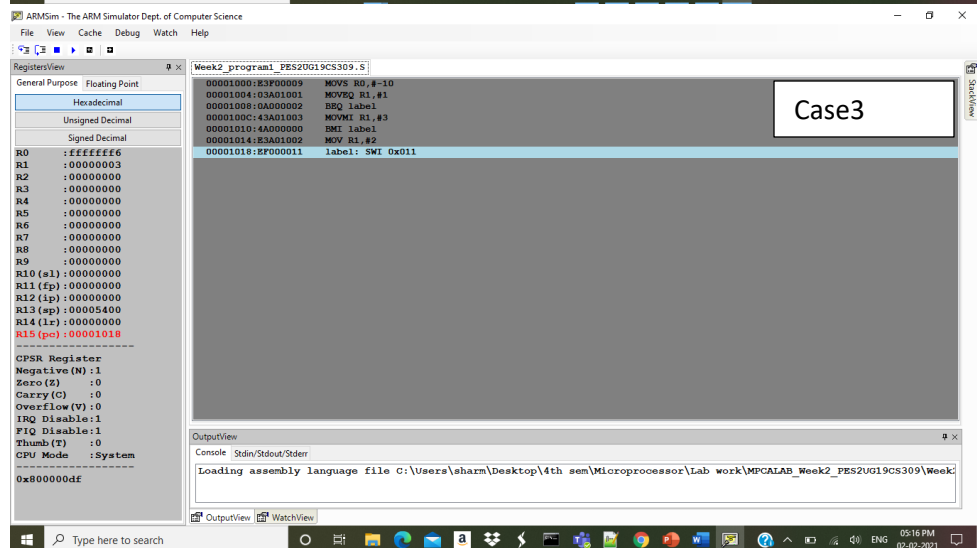
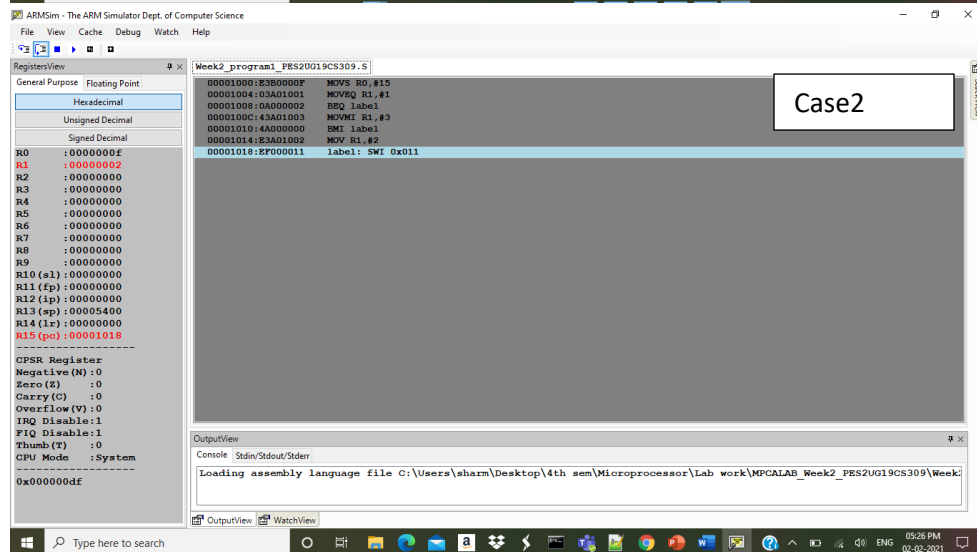
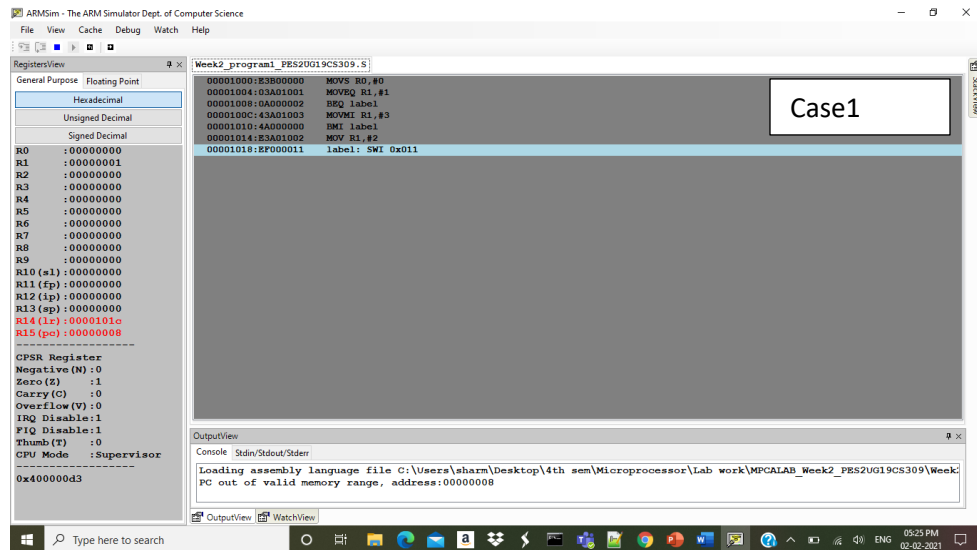
I. ARM Assembly Code for each program

```
1 .text
2 MOVS R0,#0
3 MOVEQ R1,#1
4 BEQ label
5 MOVMI R1,#3
6 BMI label
7 MOV R1,#2
8 label: SWI 0x011
9 .end
```

```
1 .text
2 MOVS R0,#15
3 MOVEQ R1,#1
4 BEQ label
5 MOVMI R1,#3
6 BMI label
7 MOV R1,#2
8 label: SWI 0x011
9 .end
```

```
1 .text
2 MOVS R0,#-10
3 MOVEQ R1,#1
4 BEQ label
5 MOVMI R1,#3
6 BMI label
7 MOV R1,#2
8 label: SWI 0x011
9 .end
```

II. Output Screen Shot



III. Input -Output Table for each program

CASE 1	R0		0x00
	R1	After compare	1
CASE 2	R0		0x0a
	R1	After compare	2
CASE 3	R0		0xFF
	R1	After compare	3

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

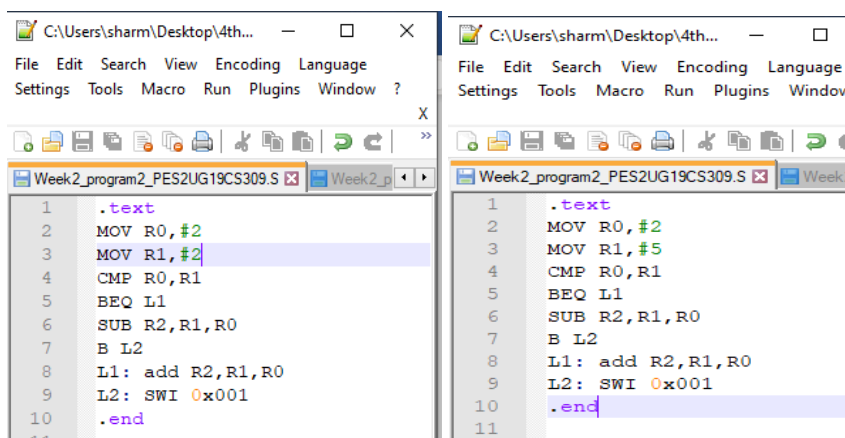
Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# ____2____ Program Number: ____2____

Title of the Program

Write an ALP to compare the value of R0 and R1, add if R0 = R1, else subtract (Program shown in class)

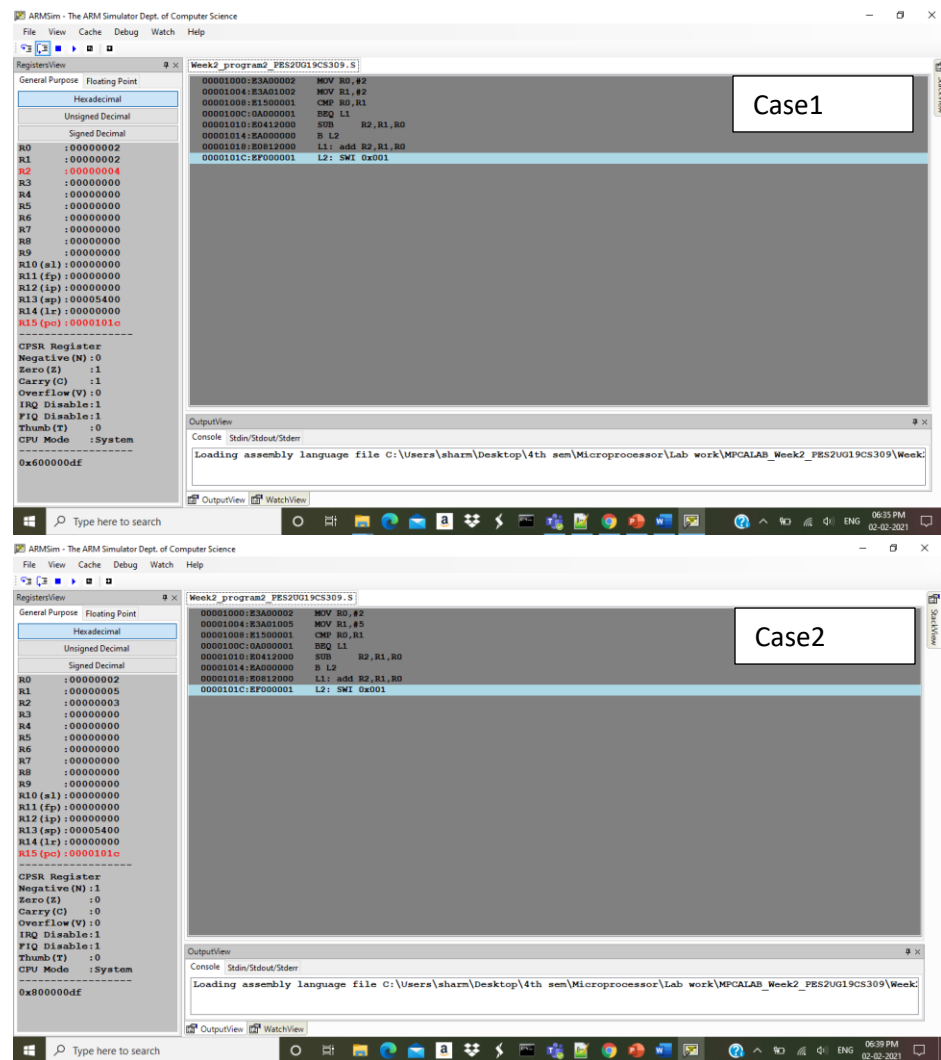
I. ARM Assembly Code for each program



```
1 .text
2 MOV R0,#2
3 MOV R1,#2
4 CMP R0,R1
5 BEQ L1
6 SUB R2,R1,R0
7 B L2
8 L1: add R2,R1,R0
9 L2: SWI 0x001
10 .end
11
```

```
1 .text
2 MOV R0,#2
3 MOV R1,#5
4 CMP R0,R1
5 BEQ L1
6 SUB R2,R1,R0
7 B L2
8 L1: add R2,R1,R0
9 L2: SWI 0x001
10 .end
11
```

II. Output Screen Shot



III. Input -Output Table for each program

CASE 1	R1=0x02, R0=0x02		
	R2=R1+R0=0x04		
CASE 2	R1=0x05, R0=0x02		
	R2=R1-R0=0x03		

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

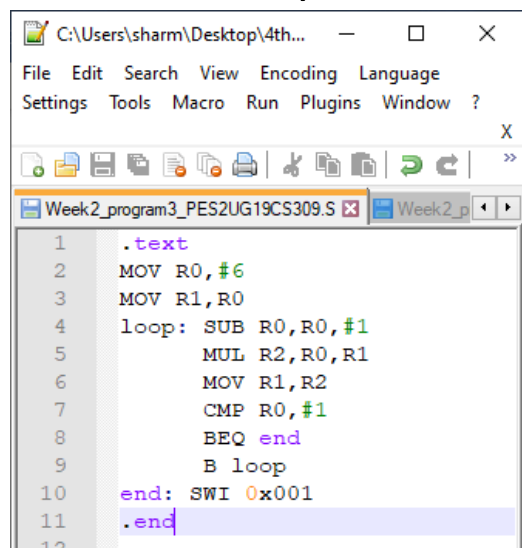
Week# ____2____ Program Number: ____3____

Title of the Program

Write an ALP to find the factorial of a number stored in R0.

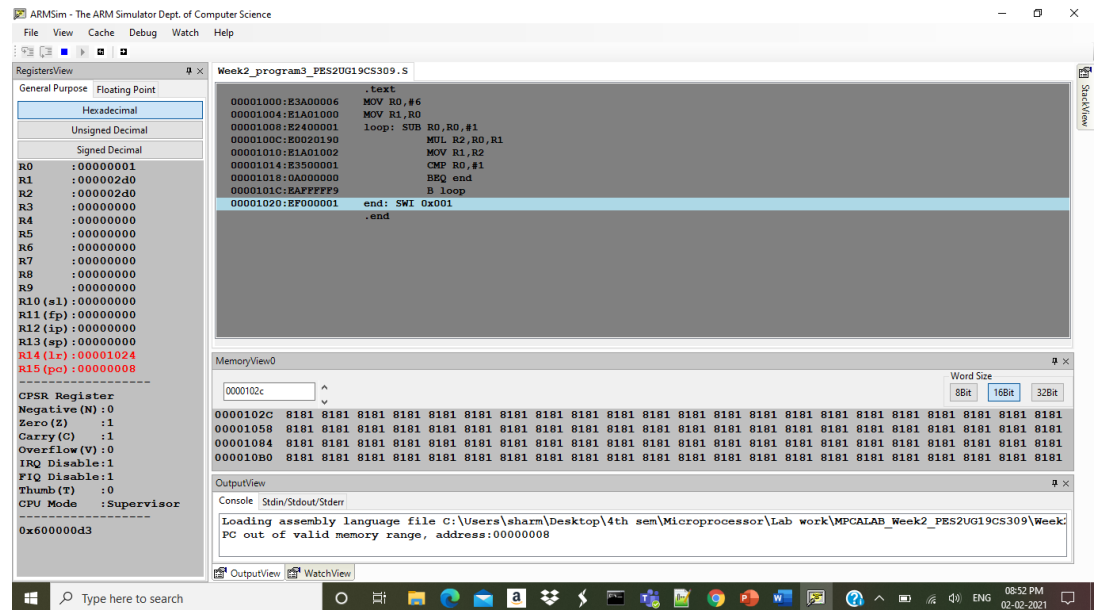
Store the value in R1 (without using LDR and STR instructions).Use only registers.(Program shown in class)

I. ARM Assembly Code for each program



```
1  .text
2  MOV R0,#6
3  MOV R1,R0
4  loop: SUB R0,R0,#1
5         MUL R2,R0,R1
6         MOV R1,R2
7         CMP R0,#1
8         BEQ end
9         B loop
10 end: SWI 0x001
11 .end
12
```

II. Output Screen Shot



III. Input -Output Table for each program

1 st Iteration	R1=0x06
	R0=0x05
	R2=0x1e=Decimal 30
2 nd Iteration	R1=0x1e
	R0=0x04 R2=0x78=Decimal 120
3 rd Iteration	R1=0x78
	R0=0x03 R2=0x168=Decimal 360
4 th Iteration	R1=0x168
	R0=0x02 R2=0x2d0=Decimal 720
5 th Iteration	R1=0x2d0
	R0=0x01 R2=0x2d0=Decimal 720

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

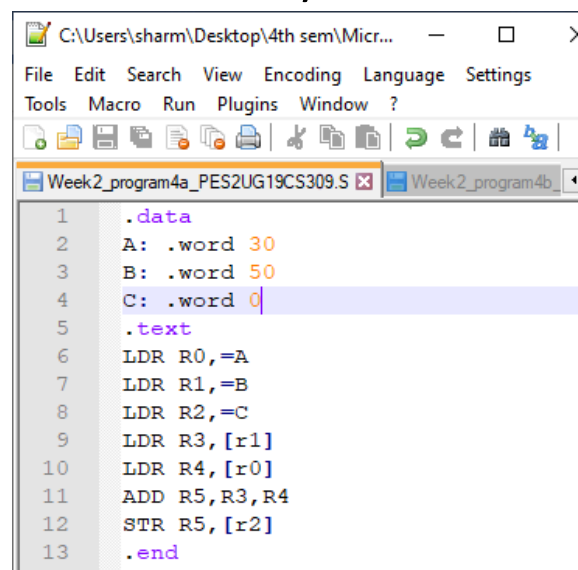
Week# ____2____

Program Number: ____4a____

Title of the Program

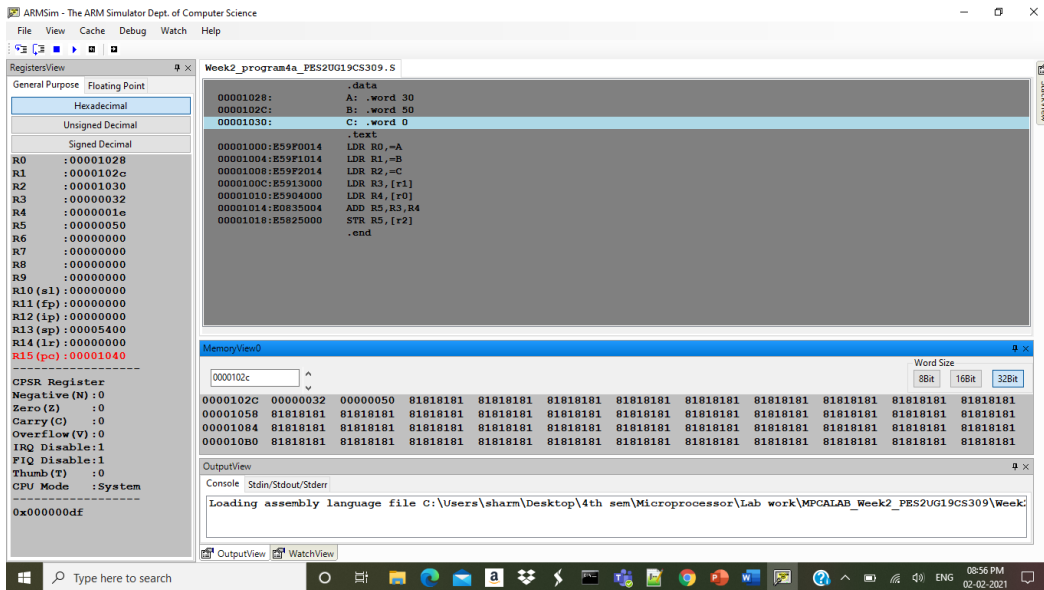
Write an ALP to add two 32 bit numbers loaded from memory and store the result in memory.

I. ARM Assembly Code for each program



```
1  .data
2  A: .word 30
3  B: .word 50
4  C: .word 0
5  .text
6  LDR R0,=A
7  LDR R1,=B
8  LDR R2,=C
9  LDR R3,[r1]
10 LDR R4,[r0]
11 ADD R5,R3,R4
12 STR R5,[r2]
13 .end
```


II. Output Screen Shot



III. Input -Output Table for each program

	A=0x1E, B=0x32
R0	Address of A
R1	Address of B
R2	Address of C
R3	0x32=Decimal 50 =Content of Location B
R4	0x1E=Decimal 30= Content of Location A
R5	0x50=Decimal 80
Location C	0x50=Decimal 80

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# ____2____ Program Number: ____4b____

Title of the Program

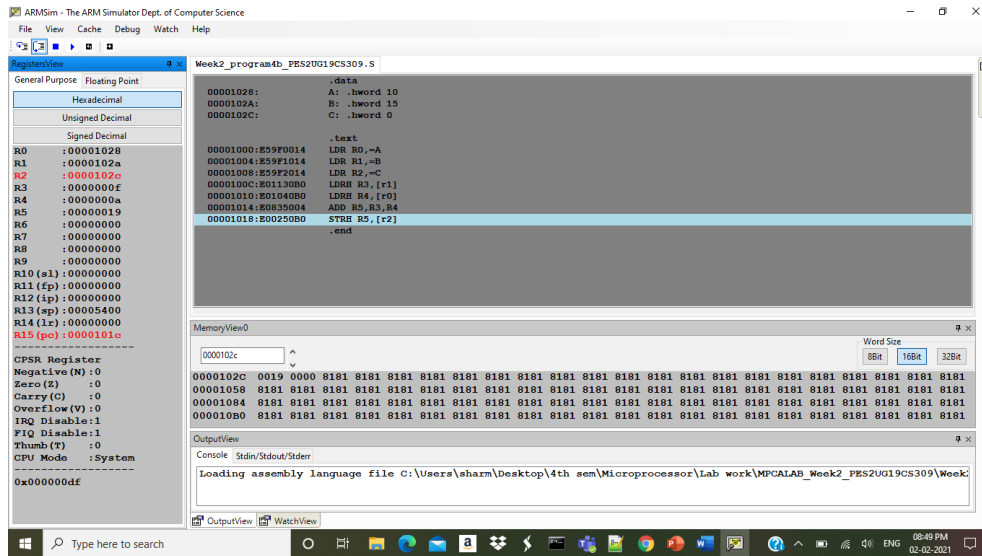
Write an ALP to add two 16 bit numbers loaded from memory and store the result in memory.

I. ARM Assembly Code for each program

```
C:\Users\sharm\Desktop\4th sem\Microprocessor\Lab work\MPC
File Edit Search View Encoding Language Settings Tools

Week2_program4a_PES2UG19CS309.S Week2_program4b_PES:
1  .data
2  A: .hword 10
3  B: .hword 15
4  C: .hword 0
5
6  .text
7  LDR R0,=A
8  LDR R1,=B
9  LDR R2,=C
10 LDRH R3,[R1]
11 LDRH R4,[R0]
12 ADD R5,R3,R4
13 STRH R5,[R2]
14 .end
15
```

II. Output Screen Shot



Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 2 Program Number: 5a

Title of the Program

Write an ALP to find GCD of two numbers (without using LDR and STR instructions). Both numbers are in registers.
Use only registers.

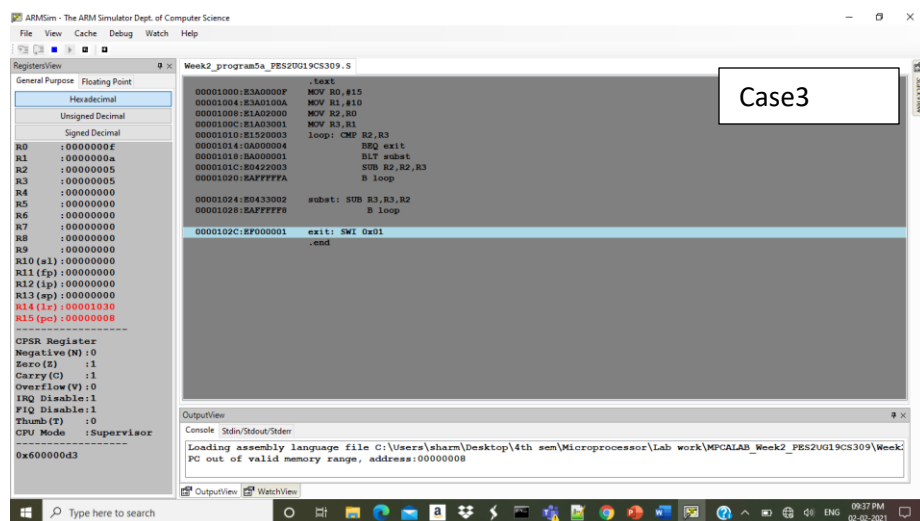
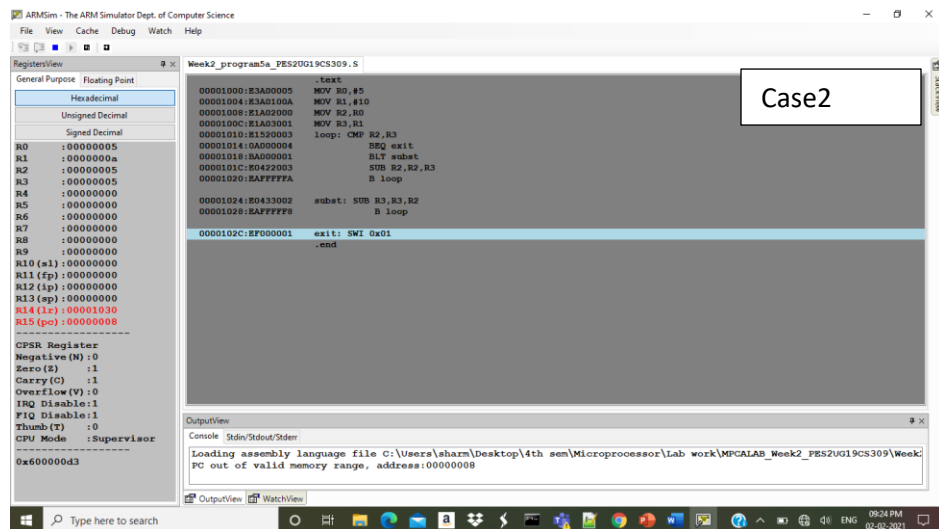
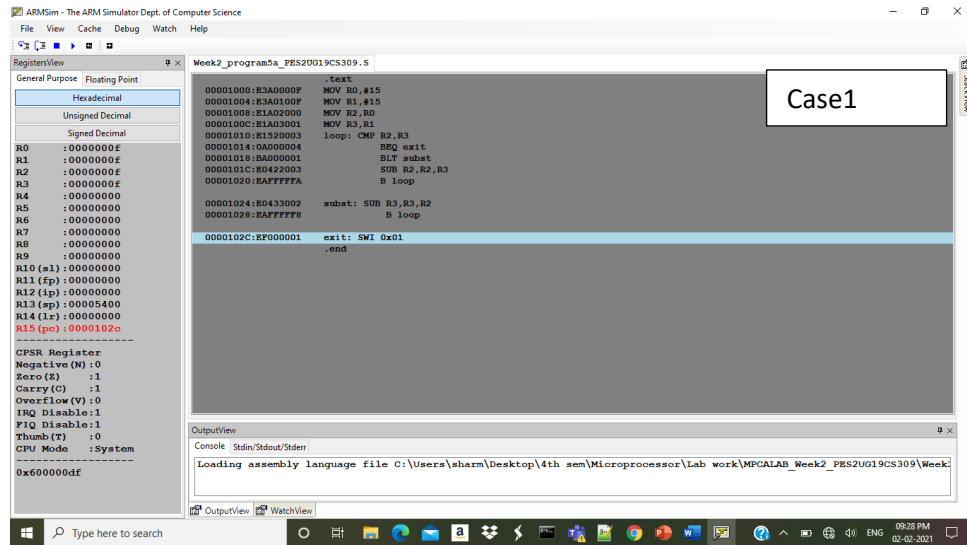
I. ARM Assembly Code for each program

```
1 .text
2 MOV R0,#15
3 MOV R1,#15
4 MOV R2,R0
5 MOV R3,R1
6 loop: CMP R2,R3
7       BEQ exit
8       BLT subst
9       SUB R2,R2,R3
10      B loop
11
12 subst: SUB R3,R3,R2
13        B loop
14
15 exit: SWI 0x01
16 .end
```

```
1 .text
2 MOV R0,#5
3 MOV R1,#10
4 MOV R2,R0
5 MOV R3,R1
6 loop: CMP R2,R3
7       BEQ exit
8       BLT subst
9       SUB R2,R2,R3
10      B loop
11
12 subst: SUB R3,R3,R2
13        B loop
14
15 exit: SWI 0x01
16 .end
```

```
1 .text
2 MOV R0,#15
3 MOV R1,#10
4 MOV R2,R0
5 MOV R3,R1
6 loop: CMP R2,R3
7       BEQ exit
8       BLT subst
9       SUB R2,R2,R3
10      B loop
11
12 subst: SUB R3,R3,R2
13        B loop
14
15 exit: SWI 0x01
16 .end
```

II. Output Screen Shot



III. Input -Output Table for each program

CASE 1	R0	0x0f
	R1	0x0f
1 st iteration	R2=0x0f R3=0x0f R3=0x0f-0x0f=0x00 GCD=15	
CASE 2	R0	0x05
	R1	0x0a
1 st iteration	R2=0x05 R3=0x0a R3=0x0a-0x05=0x05	
2 nd iteration	R2=0x05 R3=0x05 R2=0x05-0x05=0x00 GCD=5	

CASE 3	R0	0x0f
	R1	0x0a
1 st iteration	R2=0x0f R3=0x0a $R2=0x0f-0x0a=0x05$	
2 nd iteration	R2=0x05 R3=0x0a $R3=0x0a-0x05=0x05$	
3 rd iteration	R2=0x05 R3=0x05 GCD=5	

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 2 Program Number: 5b

Title of the Program

Write an ALP to find the GCD of given numbers (both numbers in memory).Store result in memory.

I. ARM Assembly Code for each program

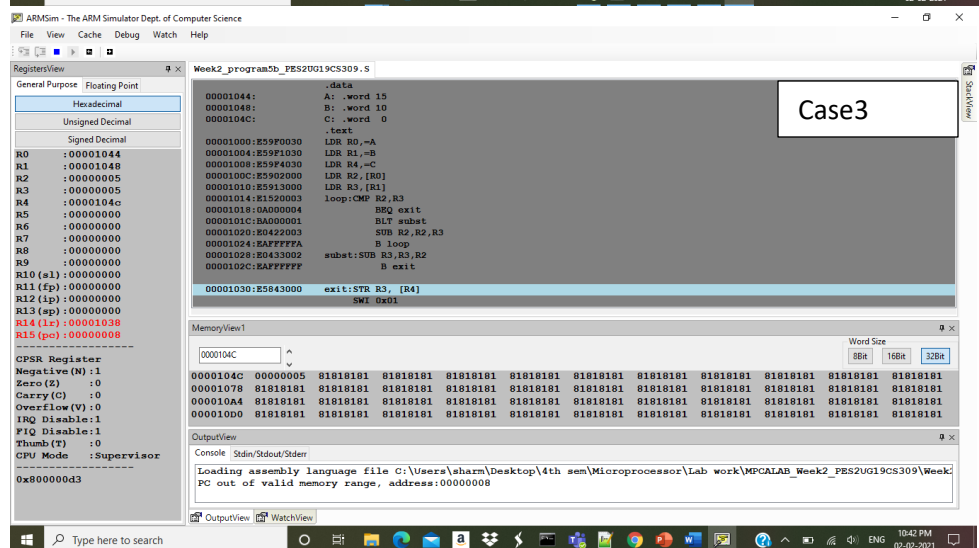
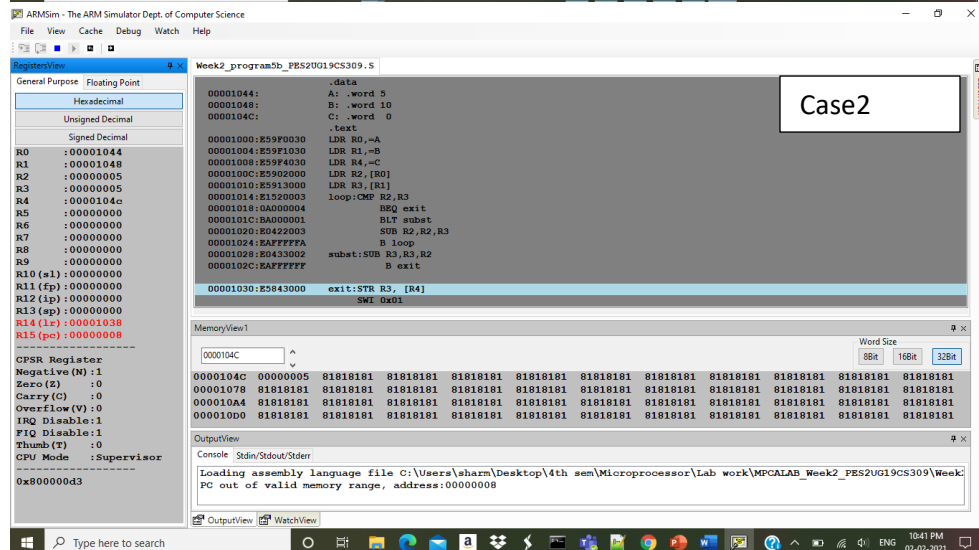
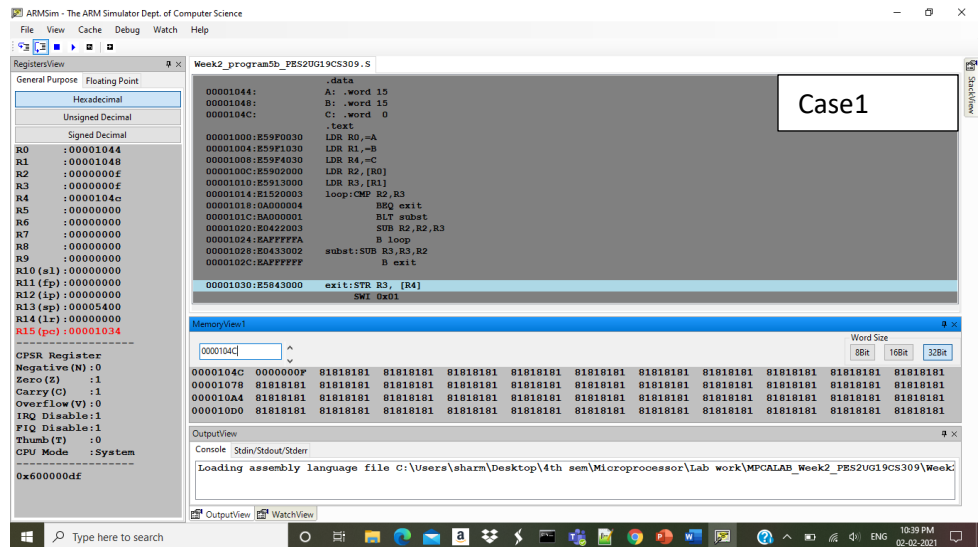
The image displays three side-by-side screenshots of an ARM assembly code editor. Each window shows a different assembly program for finding the GCD of two numbers stored in memory. The programs use a loop and conditional instructions to calculate the GCD and store the result in memory.

```
1 .data
2 A: .word 15
3 B: .word 15
4 C: .word 0
5 .text
6 LDR R0,=A
7 LDR R1,=B
8 LDR R4,=C
9 LDR R2,[R0]
10 LDR R3,[R1]
11 loop: CMP R2,R3
12 BEQ exit
13 BLT subst
14 SUB R2,R2,R3
15 B loop
16 subst: SUB R3,R3,R2
17 B exit
18
19 exit: STR R3, [R4]
20 SWI 0x01
21
```

```
1 .data
2 A: .word 5
3 B: .word 10
4 C: .word 0
5 .text
6 LDR R0,=A
7 LDR R1,=B
8 LDR R4,=C
9 LDR R2,[R0]
10 LDR R3,[R1]
11 loop: CMP R2,R3
12 BEQ exit
13 BLT subst
14 SUB R2,R2,R3
15 B loop
16 subst: SUB R3,R3,R2
17 B exit
18
19 exit: STR R3, [R4]
20 SWI 0x01
21
```

```
1 .data
2 A: .word 15
3 B: .word 10
4 C: .word 0
5 .text
6 LDR R0,=A
7 LDR R1,=B
8 LDR R4,=C
9 LDR R2,[R0]
10 LDR R3,[R1]
11 loop: CMP R2,R3
12 BEQ exit
13 BLT subst
14 SUB R2,R2,R3
15 B loop
16 subst: SUB R3,R3,R2
17 B exit
18
19 exit: STR R3, [R4]
20 SWI 0x01
21
```


II. Output Screen Shot



III. Input -Output Table for each program

CASE 1	R0	0x0f
	R1	0x0f
1 st iteration	R2=0x0f R3=0x0f R3=0x0f-0x0f=0x00 GCD=15	

CASE 2	R0	0x05
	R1	0x0a
1 st iteration	R2=0x05 R3=0x0a R3=0x0a-0x05=0x05	
2 nd iteration	R2=0x05 R3=0x05 R2=0x05-0x05=0x00 GCD=5	

CASE 3	R0	0x0f
	R1	0x0a
1 st iteration	R2=0x0f R3=0x0a R2=0x0f-0x0a=0x05	
2 nd iteration	R2=0x05 R3=0x0a R3=0x0a-0x05=0x05	
3 rd iteration	R2=0x05 R3=0x05 GCD=5	

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

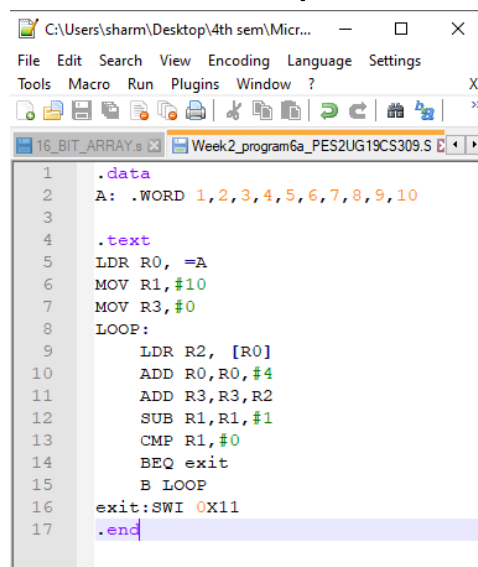
Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 2 Program Number: 6a

Title of the Program

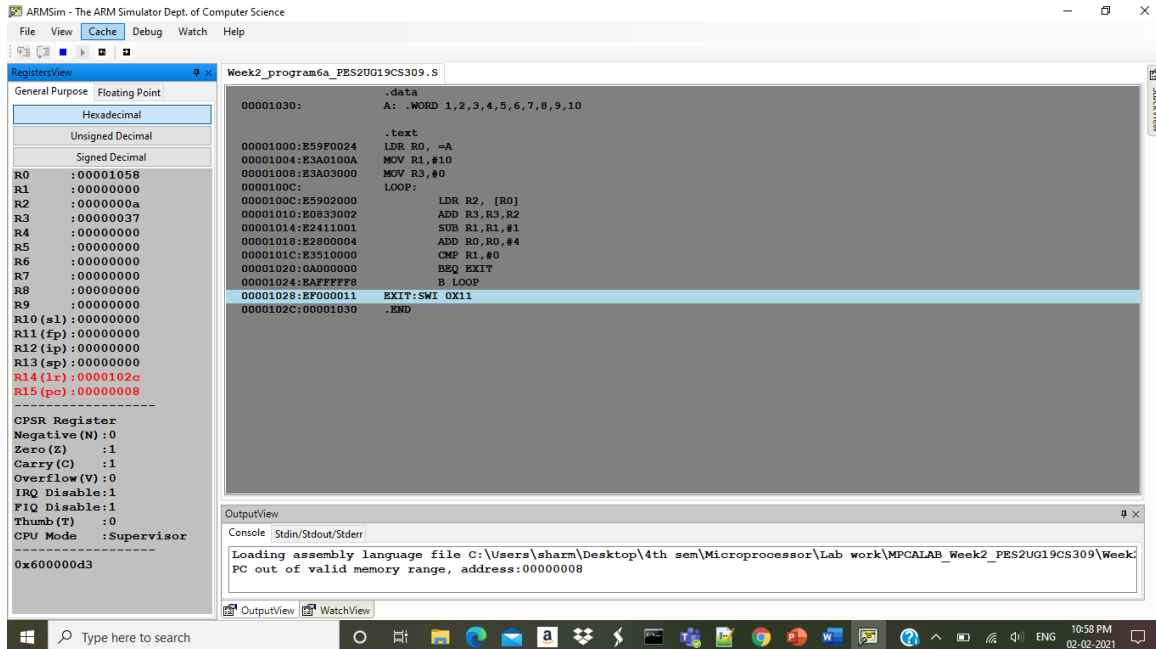
Write an ALP to add an array of ten 32 bit numbers from memory

I. ARM Assembly Code for each program



```
1  .data
2  A: .WORD 1,2,3,4,5,6,7,8,9,10
3
4  .text
5  LDR R0, =A
6  MOV R1, #10
7  MOV R3, #0
8  LOOP:
9      LDR R2, [R0]
10     ADD R0, R0, #4
11     ADD R3, R3, R2
12     SUB R1, R1, #1
13     CMP R1, #0
14     BEQ exit
15     B LOOP
16  exit: SWI 0x11
17  .end
```

II. Output Screen Shot



III. Input -Output Table for each program

A: word 1,2,3,4,5,6,7,8,9,10										
R1	10	9	8	7	6	5	4	3	2	1
R0	A	A+4	A+8	A+12	A+16	A+20	A+24	A+28	A+32	A+36
R2	1	2	3	4	5	6	7	8	9	10
R3	0	1	3	6	10	15	21	28	36	45
R3(After Execution)	1	3	6	10	15	21	28	36	45	55
Values in hex	0x01	0x03	0x06	0x0A	0x0F	0x15	0x1C	0x24	0x2D	0x37

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

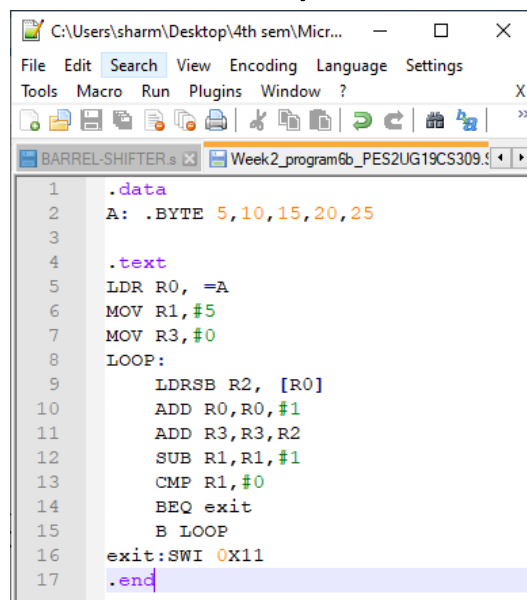
Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 2 Program Number: 6b

Title of the Program

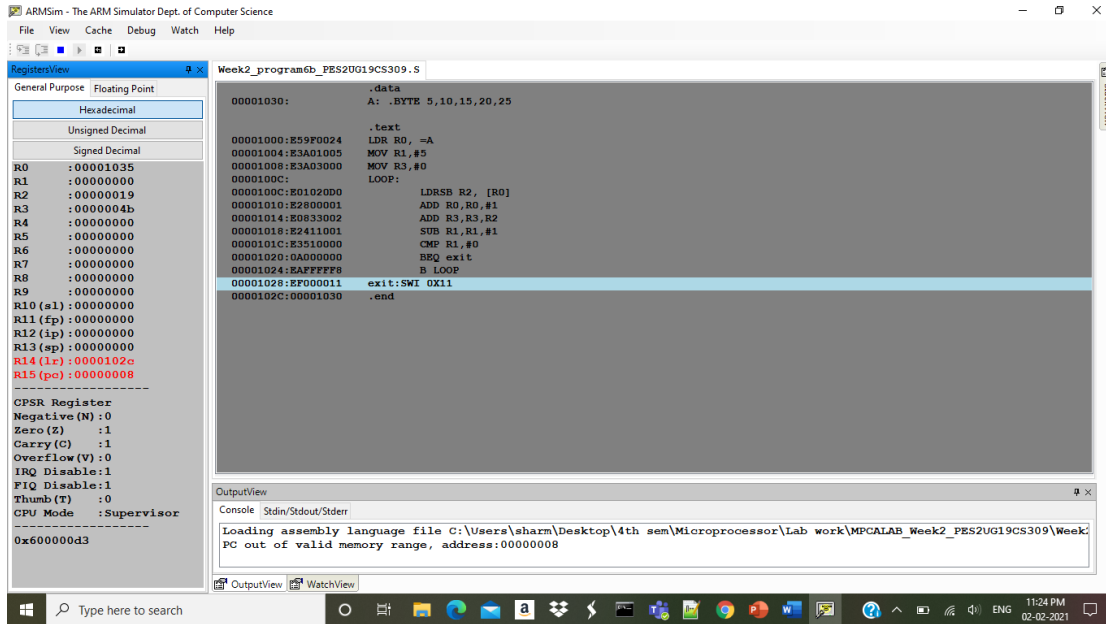
Write an ALP to add an array of five 8 bit numbers from memory(use .byte to store the data instead of .word)

I. ARM Assembly Code for each program



```
1  .data
2  A: .BYTE 5,10,15,20,25
3
4  .text
5  LDR R0, =A
6  MOV R1, #5
7  MOV R3, #0
8  LOOP:
9      LDRSB R2, [R0]
10     ADD R0, R0, #1
11     ADD R3, R3, R2
12     SUB R1, R1, #1
13     CMP R1, #0
14     BEQ exit
15     B LOOP
16 exit: SWI 0X11
17 .end
```

II. Output Screen Shot



III. Input -Output Table for each program

A:.byte 5,10,15,20,25					
R1	5	4	3	2	1
R0	A	A+1	A+2	A+3	A+4
R3	0	5	15	30	50
R2	5	10	15	20	25
R3(After Execution)	5	15	30	50	75
Values in hex	0x05	0x0F	0x1E	0x32	0x4B

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

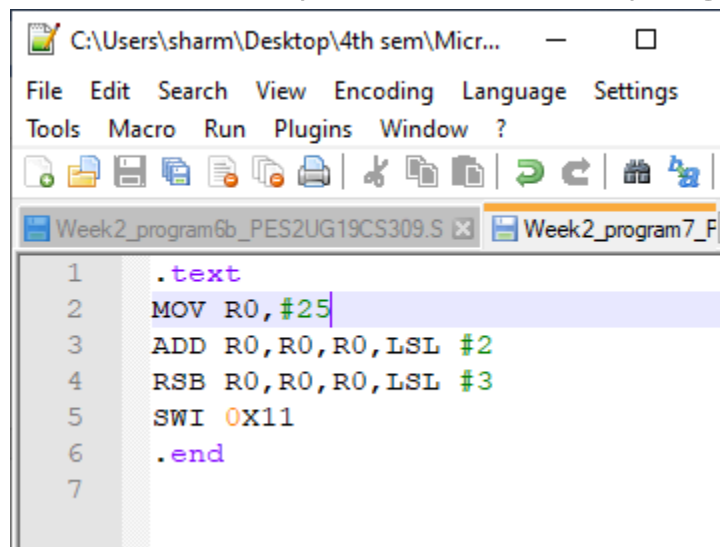
Week# 2 Program Number: 7

Title of the Program

Write an ALP to multiply $35 \times R0$

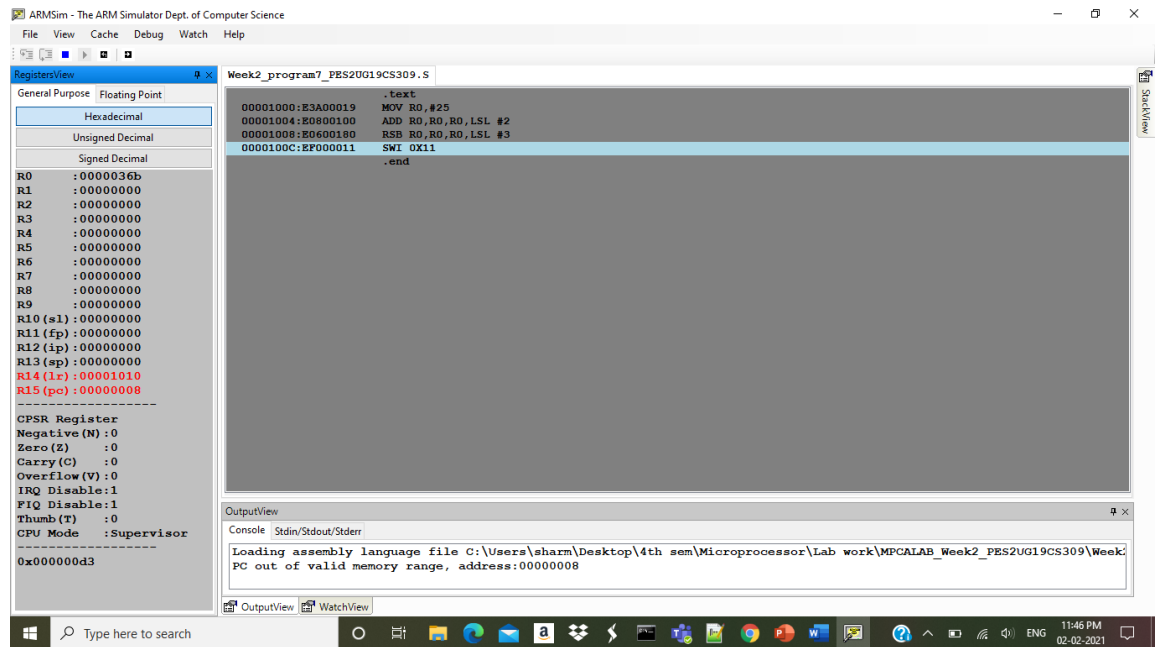
*Use LSL instruction for multiplication

I. ARM Assembly Code for each program



```
1  .text
2  MOV R0, #25
3  ADD R0, R0, R0, LSL #2
4  RSB R0, R0, R0, LSL #3
5  SWI 0X11
6  .end
7
```


II. Output Screen Shot



III. Input -Output Table for each program

	Decimal	Hexadecimal
R0(1 st line of the Program)	25	19
R0(2 nd line of the Program)	4R0=100	
	4R0+R0=	7d
	5R0=125	
R0(3 rd line of the Program)	8R0=25*8=200	
	8R0-R0=175	36b

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:1/2/2021

Name: R Sharmila	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 2 Program Number: 8

Title of the Program

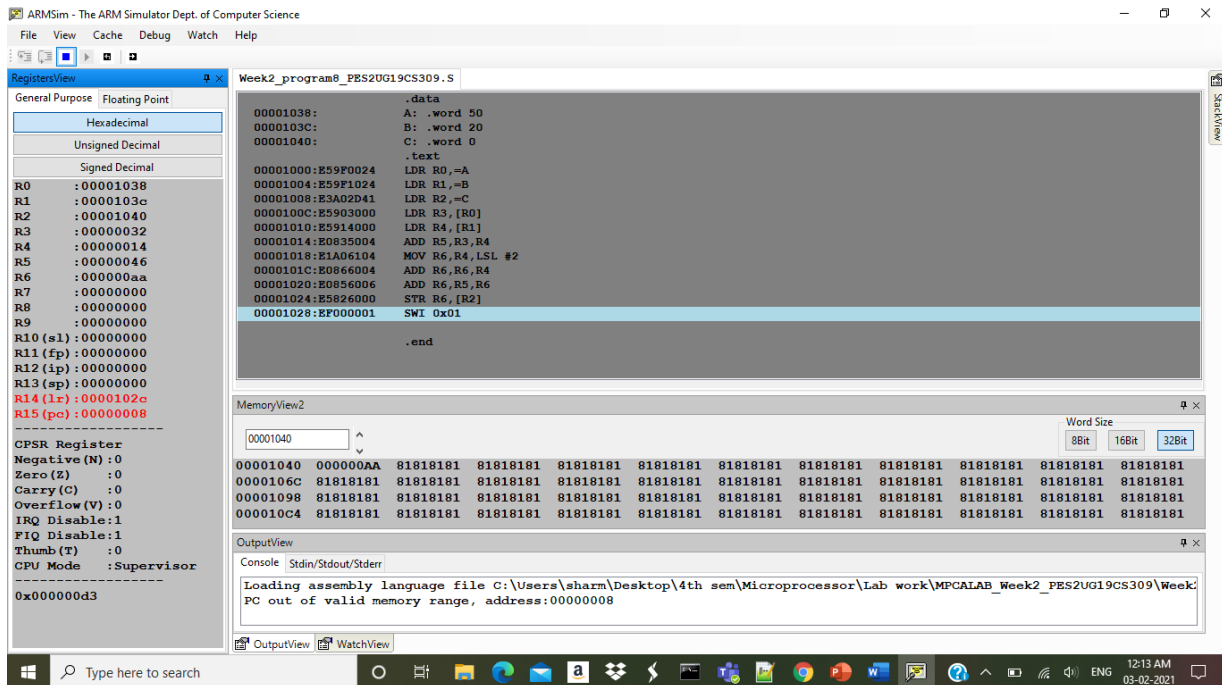
Write an ALP to evaluate the expression $(A+B) + (5*B)$, where A and B are available in memory location. 9.Store the final result in memory Location C

* Use LSL instruction for multiplication

I. ARM Assembly Code for each program

```
C:\Users\sharm\Desktop\4th sem\Microprocessor\Lab work\
File Edit Search View Encoding Language Settings
Window ?
Week2_program8_PES2UG19CS309.S
1  .data
2  A: .word 50
3  B: .word 20
4  C: .word 0
5  .text
6  LDR R0,=A
7  LDR R1,=B
8  LDR R2,=C
9  LDR R3,[R0]
10 LDR R4,[R1]
11 ADD R5,R3,R4
12 MOV R6,R4,LSL #2
13 ADD R6,R6,R4
14 ADD R6,R5,R6
15 STR R6,[R2]
16 SWI 0x01
17
18 .end
```

II. Output Screen Shot



III. Input -Output Table for each program

A=Decimal 50,B=Decimal 19		
	Decimal	Hexadecimal
R3	50	32
R4	19	13
R5=R3+R4=A+B	69	45
Calculate 4*B	19*4=76	4C
Calculate 5*B	76+19=95	5F
Calculate (A+B)+5*B	69+95=164	A4

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: Sharmila

Name: R Sharmila

SRN: PES2UG19CS309

Section: E

Date: 1/2/2021