Microprocessor and Computer Architecture Laboratory
COURSE CODE :UE19CS256
4th Semester, Academic Year 2020-21

Date:08/02/2021

Name: Priya Mohata	SRN:PES2UG19CS301	Section:E

Week#4 **Program Number: 1**

Write an ALP to implement C[k] =a[i]+b[j]

I. ARM Assembly Code

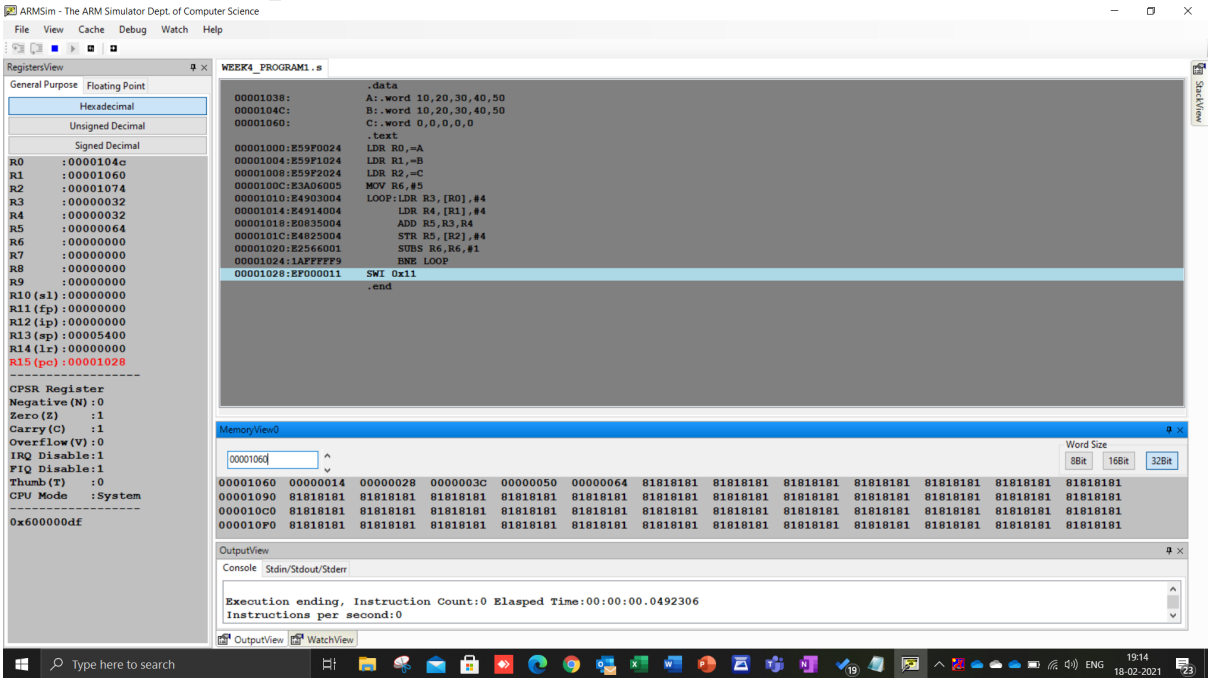
WEEK4_PROGRAM1.s

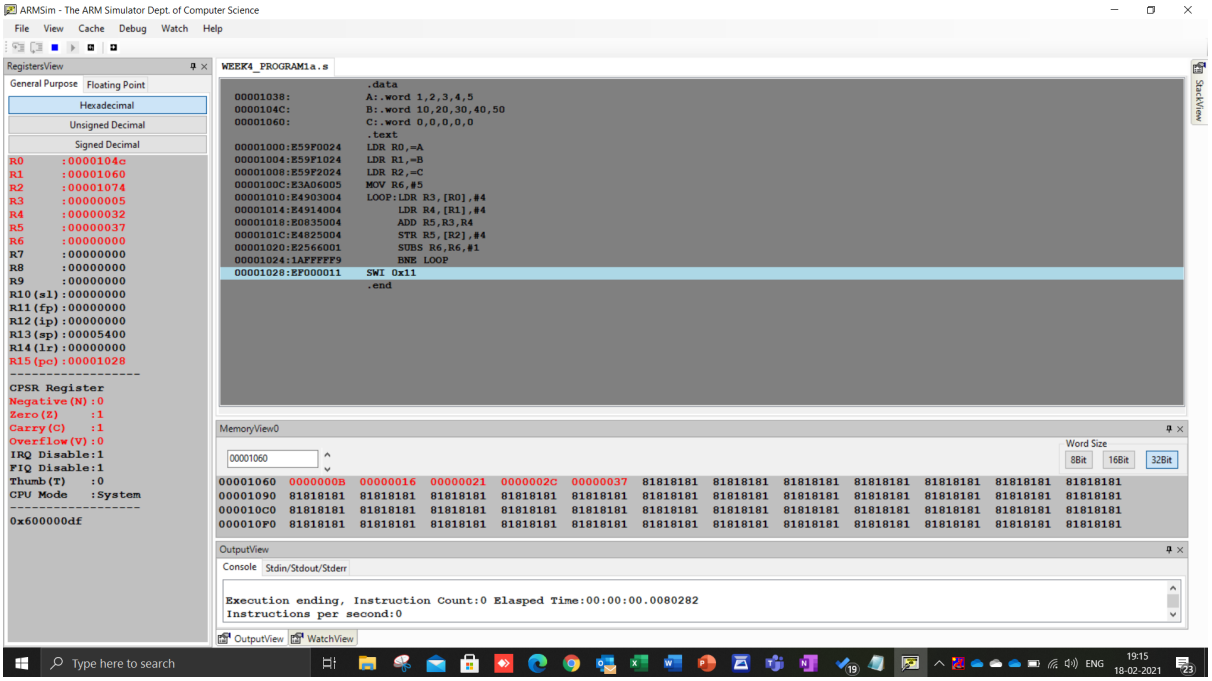
```
.data
A:.word 10,20,30,40,50
B:.word 10,20,30,40,50
C:.word 0,0,0,0,0
.text
LDR R0,=A
LDR R1,=B
LDR R2,=C
MOV R6,#5
LOOP:LDR R3,[R0],#4
     LDR R4,[R1],#4
     ADD R5,R3,R4
     STR R5,[R2],#4
     SUBS R6,R6,#1
     BNE LOOP
SWI 0x11
.end
```

WEEK4_PROGRAM1a.s

```
.data
A:.word 1,2,3,4,5
B:.word 10,20,30,40,50
C:.word 0,0,0,0,0
.text
LDR R0,=A
LDR R1,=B
LDR R2,=C
MOV R6,#5
LOOP:LDR R3,[R0],#4
     LDR R4,[R1],#4
     ADD R5,R3,R4
     STR R5,[R2],#4
     SUBS R6,R6,#1
     BNE LOOP
SWI 0x11
.end
```

II. Output Screen Shot





III. Output Table for the program

a: .word 10, 20, 30, 40, 50 **b: .word 10, 20, 30, 40, 50** **c: .word 0,0,0,0,0**	
After execution the content of array C is	
20	**00000014**
40	00000028
60	**0000003C**
80	00000050
100	00000064

a: .word 1, 2, 3, 4, 5 **b: .word 10, 20, 30, 40, 50** **c: .word 0,0,0,0,0**	
After execution the content of array C is	
20	**0000000B**
40	00000016
60	**00000021**
80	0000002C
100	00000037

Week#4 Program Number: 2

Title : Write an ALP to implement c[k] = a[i] * b[j]

I. ARM Assembly Code.

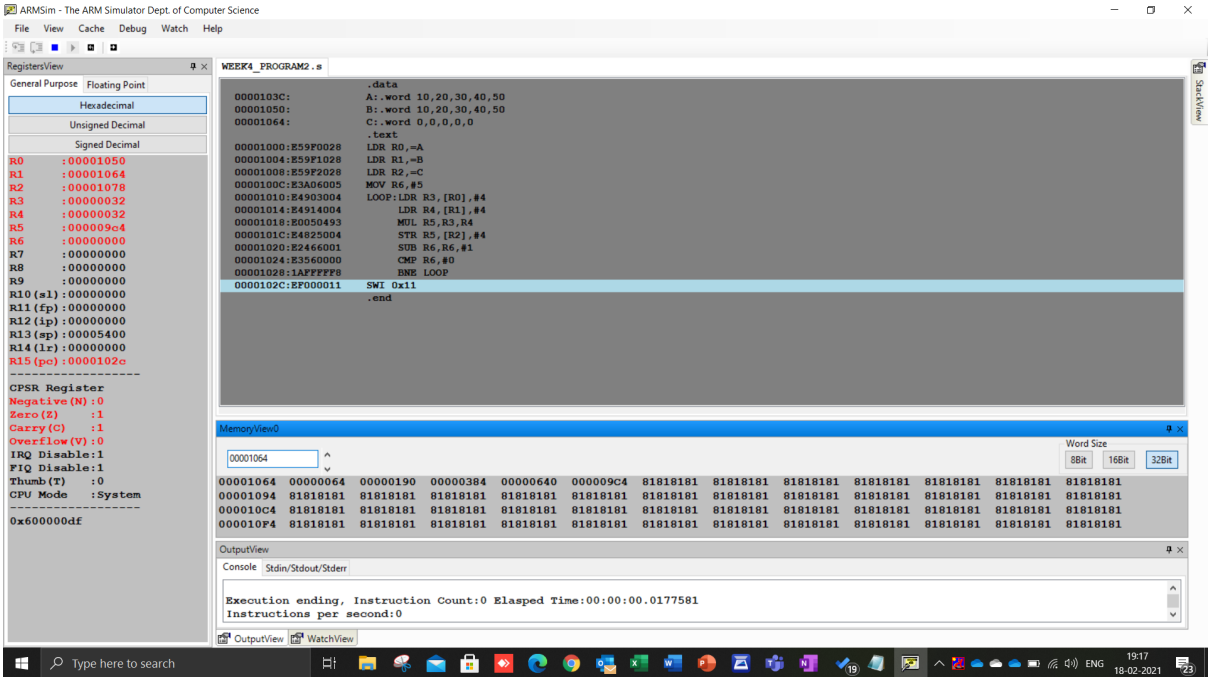
WEEK4_PROGRAM2.s

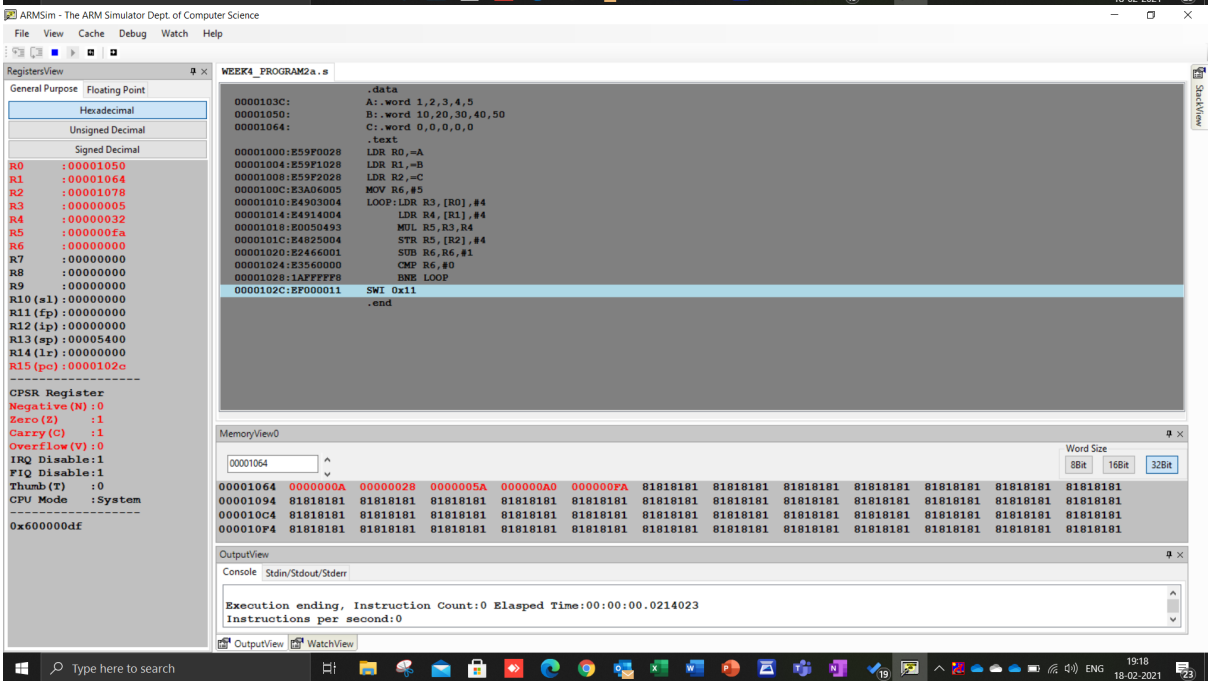
```
.data
A:.word 10,20,30,40,50
B:.word 10,20,30,40,50
C:.word 0,0,0,0,0
.text
LDR R0,=A
LDR R1,=B
LDR R2,=C
MOV R6,#5
LOOP:LDR R3,[R0],#4
     LDR R4,[R1],#4
     MUL R5,R3,R4
     STR R5,[R2],#4
     SUB R6,R6,#1
     CMP R6,#0
     BNE LOOP
SWI 0x11
.end
```

WEEK4_PROGRAM2a.s

```
.data
A:.word 1,2,3,4,5
B:.word 10,20,30,40,50
C:.word 0,0,0,0,0
.text
LDR R0,=A
LDR R1,=B
LDR R2,=C
MOV R6,#5
LOOP:LDR R3,[R0],#4
     LDR R4,[R1],#4
     MUL R5,R3,R4
     STR R5,[R2],#4
     SUB R6,R6,#1
     CMP R6,#0
     BNE LOOP
SWI 0x11
.end
```

II. Output Screenshot


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 :00001050
R1 :00001064
R2 :00001078
R3 :00000032
R4 :00000032
R5 :000009c4
R6 :00000000
R7 :00000000
R8 :00000000
R9 :00000000
R10(sl):00000000
R11(fp):00000000
R12(ip):00000000
R13(sp):00005400
R14(lr):00000000
R15(pc):0000102c
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
WEEK4_PROGRAM2.s
.data
0000103C: A:.word 10,20,30,40,50
00001050: B:.word 10,20,30,40,50
00001064: C:.word 0,0,0,0,0
.text
00001000:E59F0028 LDR R0,=A
00001004:E59F1028 LDR R1,=B
00001008:E59F2028 LDR R2,=C
0000100C:E3A06005 MOV R6,#5
00001010:E4903004 LOOP:LDR R3,[R0],#4
00001014:E4914004 LDR R4,[R1],#4
00001018:E0050493 MUL R5,R3,R4
0000101C:E4825004 STR R5,[R2],#4
00001020:E2466001 SUB R6,R6,#1
00001024:E3560000 CMP R6,#0
00001028:1AFFFFF8 BNE LOOP
0000102C:EF000011 SWI 0x11
.end
MemoryView0
00001064
Word Size 8Bit 16Bit 32Bit
00001064 00000064 00000190 00000384 00000640 000009C4 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001094 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
000010F4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0177581
Instructions per second:0
OutputView WatchView
Type here to search
ENG 19:17 18-02-2021


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 :00001050
R1 :00001064
R2 :00001078
R3 :00000005
R4 :00000032
R5 :000000fa
R6 :00000000
R7 :00000000
R8 :00000000
R9 :00000000
R10(sl):00000000
R11(fp):00000000
R12(ip):00000000
R13(sp):00005400
R14(lr):00000000
R15(pc):0000102c
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
WEEK4_PROGRAM2a.s
.data
0000103C: A:.word 1,2,3,4,5
00001050: B:.word 10,20,30,40,50
00001064: C:.word 0,0,0,0,0
.text
00001000:E59F0028 LDR R0,=A
00001004:E59F1028 LDR R1,=B
00001008:E59F2028 LDR R2,=C
0000100C:E3A06005 MOV R6,#5
00001010:E4903004 LOOP:LDR R3,[R0],#4
00001014:E4914004 LDR R4,[R1],#4
00001018:E0050493 MUL R5,R3,R4
0000101C:E4825004 STR R5,[R2],#4
00001020:E2466001 SUB R6,R6,#1
00001024:E3560000 CMP R6,#0
00001028:1AFFFFF8 BNE LOOP
0000102C:EF000011 SWI 0x11
.end
MemoryView0
00001064
Word Size 8Bit 16Bit 32Bit
00001064 0000000A 00000028 0000005A 000000A0 000000FA 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001094 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
000010F4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0214023
Instructions per second:0
OutputView WatchView
Type here to search
ENG 19:18 18-02-2021

III. Output Table for the program

	a: .word 10, 20, 30, 40, 50 **b: .word 10, 20, 30, 40, 50** **c: .word 0,0,0,0,0**

After Execution The content of array C is	
100	**00000064**
400	00000190
900	**00000384**
1600	00000640
2500	000009C4

	a: .word 1, 2, 3, 4, 5 **b: .word 10, 20, 30, 40, 50** **c: .word 0,0,0,0,0**

After Execution The content of array C is	
100	**0000000A**
400	00000028
900	**0000005A**
1600	000000A0
2500	000000FA

Week#4 Program Number: 3a

a. Write an ALP to perform Convolution using MUL instruction (Addition of multiplication of respective numbers of loc A and loc B)

I. ARM Assembly Code

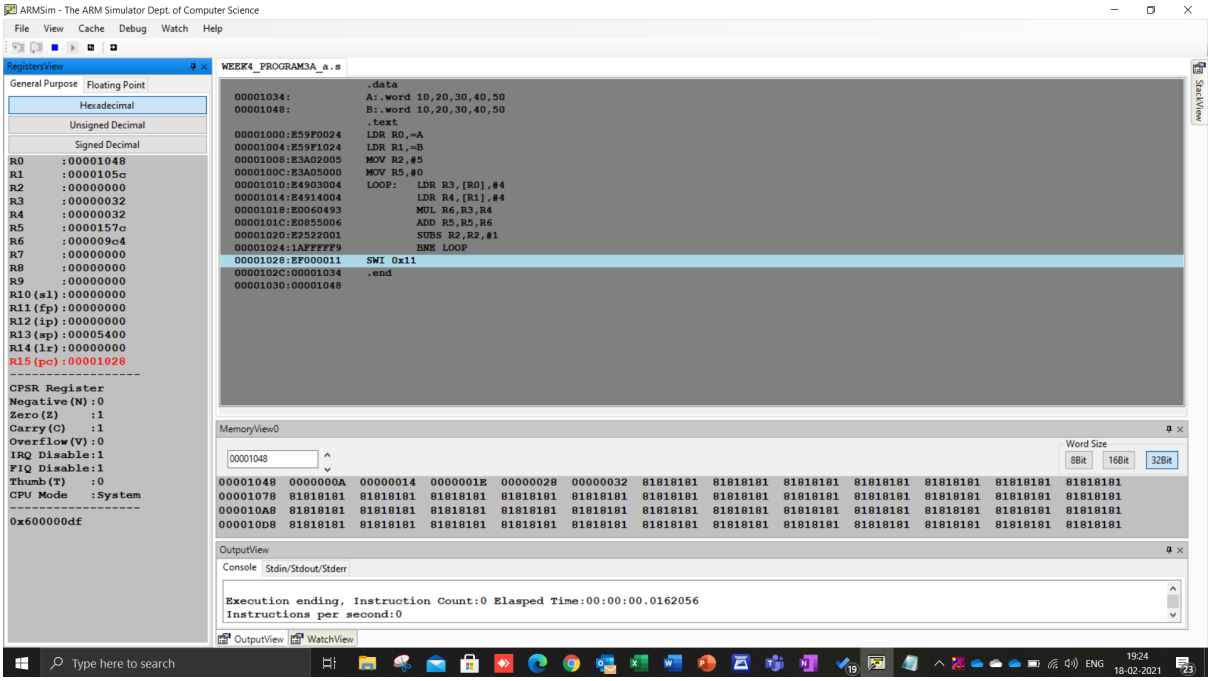
WEEK4_PROGRAM3A_a.s

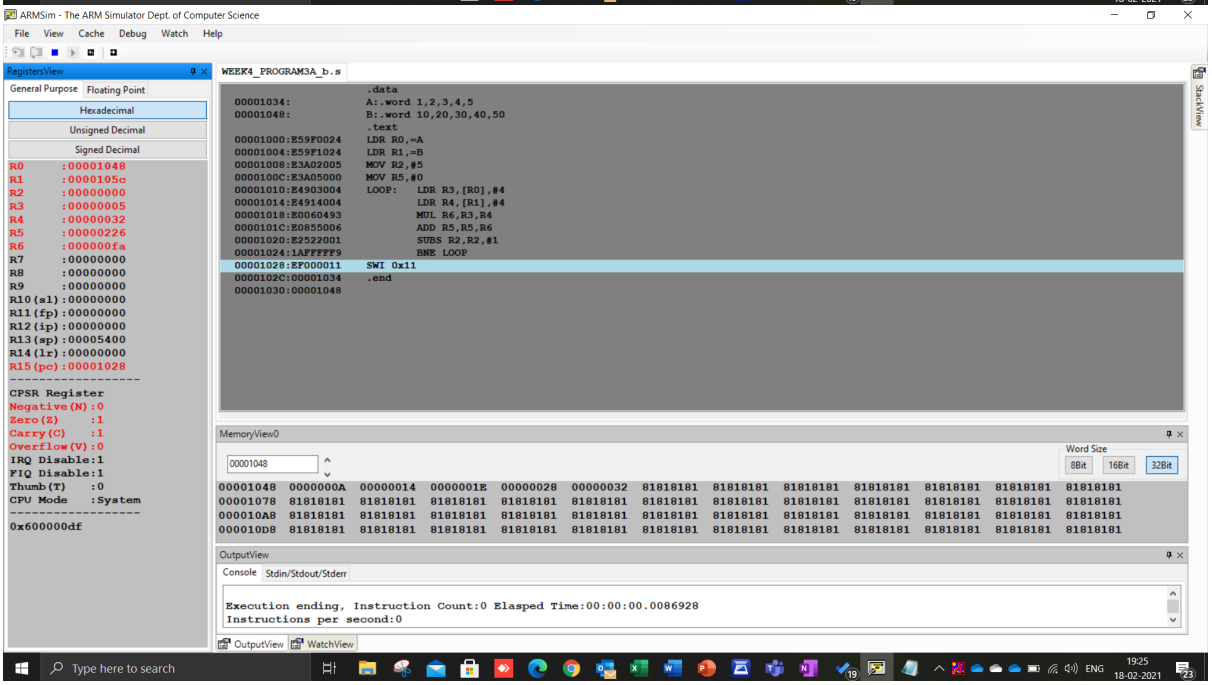
```
.data
A:.word 10,20,30,40,50
B:.word 10,20,30,40,50
.text
LDR R0,=A
LDR R1,=B
MOV R2,#5
MOV R5,#0
LOOP:   LDR R3,[R0],#4
        LDR R4,[R1],#4
        MUL R6,R3,R4
        ADD R5,R5,R6
        SUBS R2,R2,#1
        BNE LOOP
SWI 0x11
.end
```

WEEK4_PROGRAM3A_b.s

```
.data
A:.word 1,2,3,4,5
B:.word 10,20,30,40,50
.text
LDR R0,=A
LDR R1,=B
MOV R2,#5
MOV R5,#0
LOOP:   LDR R3,[R0],#4
        LDR R4,[R1],#4
        MUL R6,R3,R4
        ADD R5,R5,R6
        SUBS R2,R2,#1
        BNE LOOP
SWI 0x11
.end
```

II. Output Screen Shot





III. Output Table for the program

	a: .word 10, 20, 30, 40, 50 b: .word 10, 20, 30, 40, 50
R5	**(10*10)+(20*20)+(30*30) +(40*40)+(50*50) =5500=0x157c**

	a: .word 1, 2, 3, 4, 5 **b: .word 10, 20, 30, 40, 50**
R5	**(1*10)+(2*20)+(3*30)** **+(4*40)+(5*50)** **=550=0x226**

Week #4 Program:3b

Write an ALP to perform Convolution using MLA instruction (Addition of multiplication of respective numbers of loc A and loc B).

I.ARM ASSEMBLY CODE

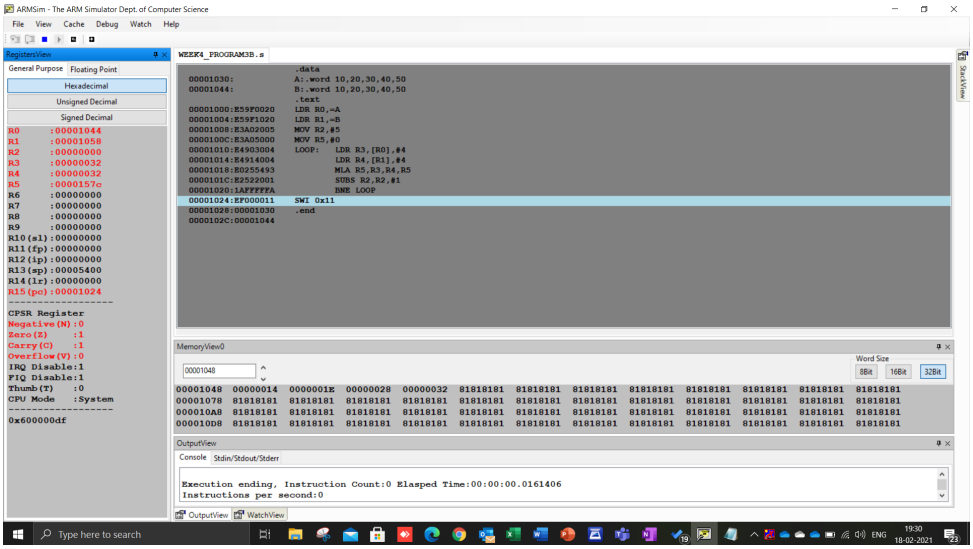
WEEK4_PROGRAM3B.s

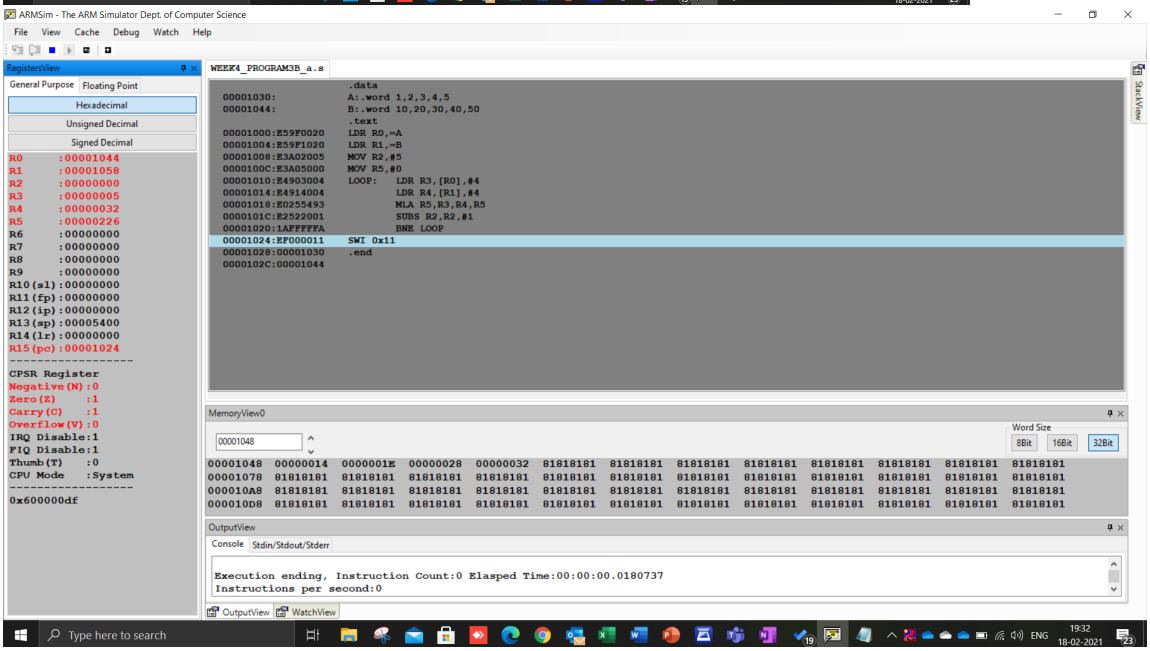
```
.data
A:.word 10,20,30,40,50
B:.word 10,20,30,40,50
.text
LDR R0,=A
LDR R1,=B
MOV R2,#5
MOV R5,#0
LOOP:   LDR R3,[R0],#4
        LDR R4,[R1],#4
        MLA R5,R3,R4,R5
        SUBS R2,R2,#1
        BNE LOOP
SWI 0x11
.end
```

WEEK4_PROGRAM3B_a.s

```
.data
A:.word 1,2,3,4,5
B:.word 10,20,30,40,50
.text
LDR R0,=A
LDR R1,=B
MOV R2,#5
MOV R5,#0
LOOP:   LDR R3,[R0],#4
        LDR R4,[R1],#4
        MLA R5,R3,R4,R5
        SUBS R2,R2,#1
        BNE LOOP
SWI 0x11
.end
```

II.OUTPUT SCREENSHOTS





III.OUTPUT TABLE

	a: .word 10, 20, 30, 40, 50 **b: .word 10, 20, 30, 40, 50**
R5	**(10*10)+(20*20)+(30*30)** **+(40*40)+(50*50)** **=5500=0x157c**

	a: .word 1, 2, 3, 4, 5 **b: .word 10, 20, 30, 40, 50**
R5	**(1*10)+(2*20)+(3*30)** **+(4*40)+(5*50)** **=550=0x226**

Week#4 Program Number:4

Write an ALP to read from a 2D array such that B=a[i][j]

I. ARM Assembly Code

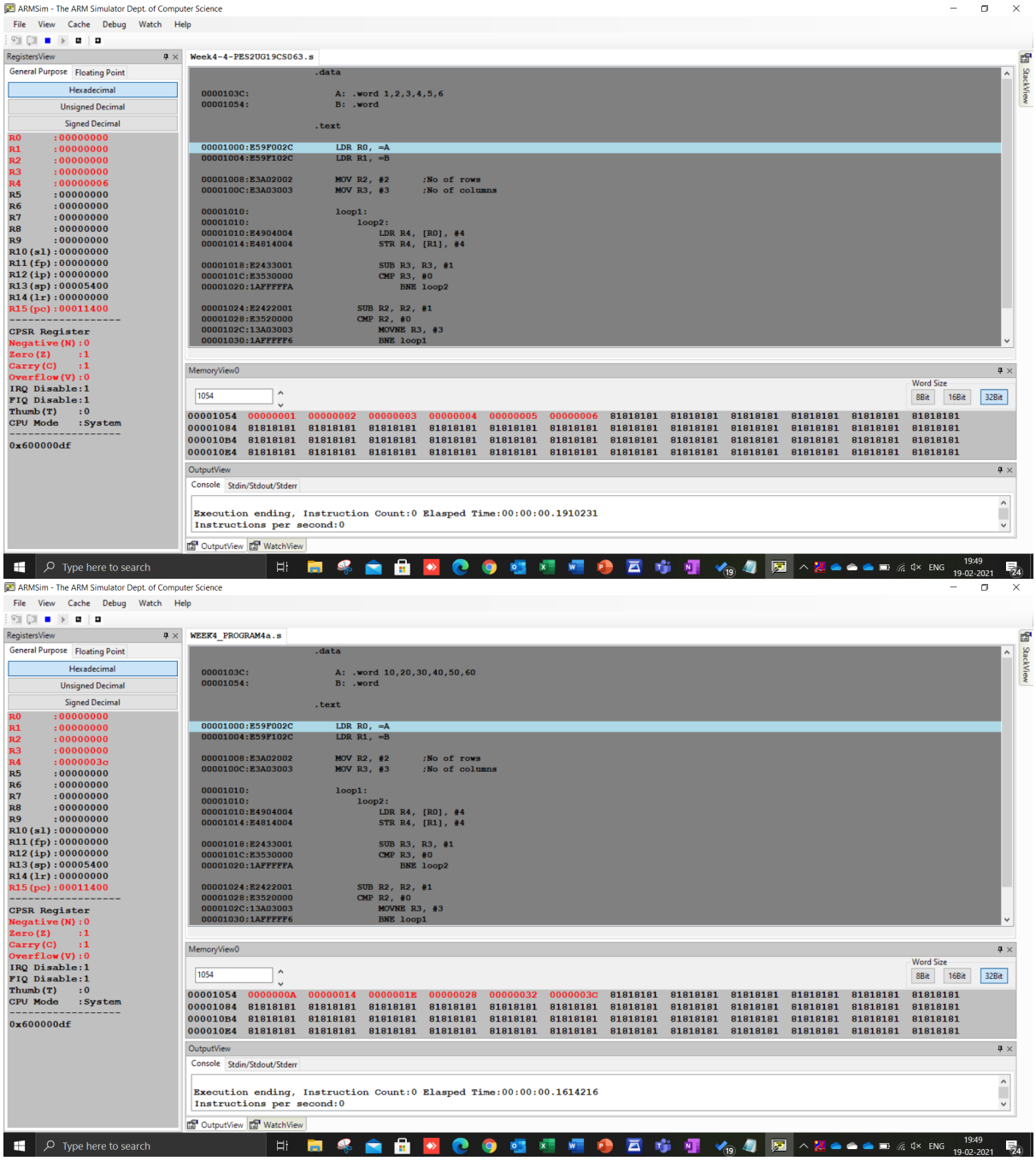
Week4_PROGRAM4_PES2UG19CS301.s

```
.data

    A: .word 1,2,3,4,5,6
    B: .word

.text

    LDR R0, =A
    LDR R1, =B

    MOV R2, #2      ;No of rows
    MOV R3, #3      ;No of columns

    loop1:
        loop2:
            LDR R4, [R0], #4
            STR R4, [R1], #4

            SUB R3, R3, #1
            CMP R3, #0
                BNE loop2

        SUB R2, R2, #1
        CMP R2, #0
            MOVNE R3, #3
            BNE loop1

.end
```

WEEK4_PROGRAM4a.s

```
.data

    A: .word 10,20,30,40,50,60
    B: .word

.text

    LDR R0, =A
    LDR R1, =B

    MOV R2, #2      ;No of rows
    MOV R3, #3      ;No of columns

    loop1:
        loop2:
            LDR R4, [R0], #4
            STR R4, [R1], #4

            SUB R3, R3, #1
            CMP R3, #0
                BNE loop2

        SUB R2, R2, #1
        CMP R2, #0
            MOVNE R3, #3
            BNE loop1

.end
```

II. Output Screen Shot


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 :00000000
R1 :00000000
R2 :00000000
R3 :00000000
R4 :00000006
R5 :00000000
R6 :00000000
R7 :00000000
R8 :00000000
R9 :00000000
R10(sl) :00000000
R11(fp) :00000000
R12(ip) :00000000
R13(sp) :00005400
R14(lr) :00000000
R15(pc) :00011400
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
Week4-4-PES2UG19CS063.s
.data
0000103C: A: .word 1,2,3,4,5,6
00001054: B: .word
.text
00001000:E59F002C LDR R0, =A
00001004:E59F102C LDR R1, =B
00001008:E3A02002 MOV R2, #2 ;No of rows
0000100C:E3A03003 MOV R3, #3 ;No of columns
00001010: loop1:
00001010: loop2:
00001010:E4904004 LDR R4, [R0], #4
00001014:E4814004 STR R4, [R1], #4
00001018:E2433001 SUB R3, R3, #1
0000101C:E3530000 CMP R3, #0
00001020:1AFFFFFA BNE loop2
00001024:E2422001 SUB R2, R2, #1
00001028:E3520000 CMP R2, #0
0000102C:13A03003 MOVNE R3, #3
00001030:1AFFFFF6 BNE loop1
MemoryView0
1054
Word Size
8Bit 16Bit 32Bit
00001054 00000001 00000002 00000003 00000004 00000005 00000006 81818181 81818181 81818181 81818181 81818181 81818181
00001084 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010B4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010E4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.1910231
Instructions per second:0
OutputView WatchView
StackView
Type here to search
ENG 19:49 19-02-2021

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 :00000000
R1 :00000000
R2 :00000000
R3 :00000000
R4 :0000003c
R5 :00000000
R6 :00000000
R7 :00000000
R8 :00000000
R9 :00000000
R10(sl) :00000000
R11(fp) :00000000
R12(ip) :00000000
R13(sp) :00005400
R14(lr) :00000000
R15(pc) :00011400
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
WEEK4_PROGRAM4a.s
.data
0000103C: A: .word 10,20,30,40,50,60
00001054: B: .word
.text
00001000:E59F002C LDR R0, =A
00001004:E59F102C LDR R1, =B
00001008:E3A02002 MOV R2, #2 ;No of rows
0000100C:E3A03003 MOV R3, #3 ;No of columns
00001010: loop1:
00001010: loop2:
00001010:E4904004 LDR R4, [R0], #4
00001014:E4814004 STR R4, [R1], #4
00001018:E2433001 SUB R3, R3, #1
0000101C:E3530000 CMP R3, #0
00001020:1AFFFFFA BNE loop2
00001024:E2422001 SUB R2, R2, #1
00001028:E3520000 CMP R2, #0
0000102C:13A03003 MOVNE R3, #3
00001030:1AFFFFF6 BNE loop1
MemoryView0
1054
Word Size
8Bit 16Bit 32Bit
00001054 0000000A 00000014 0000001E 00000028 00000032 0000003C 81818181 81818181 81818181 81818181 81818181 81818181
00001084 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010B4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010E4 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.1614216
Instructions per second:0
OutputView WatchView
StackView
Type here to search
ENG 19:49 19-02-2021

III. Output Table for the program

Before execution	a:.word 1,2,3,4,5,6	b: .word
After Execution	00000001	00000001
	00000002	00000002
	00000003	00000003
	00000004	00000004
	00000005	00000005
	00000006	00000006

Before execution	a:.word 10,20,30,40,50,60	b: .word
After Execution	0000000A	0000000A
	00000014	00000014
	0000001E	0000001E
	00000028	00000028
	00000032	00000032
	0000003C	0000003C

Week#4 **Program Number: 5**

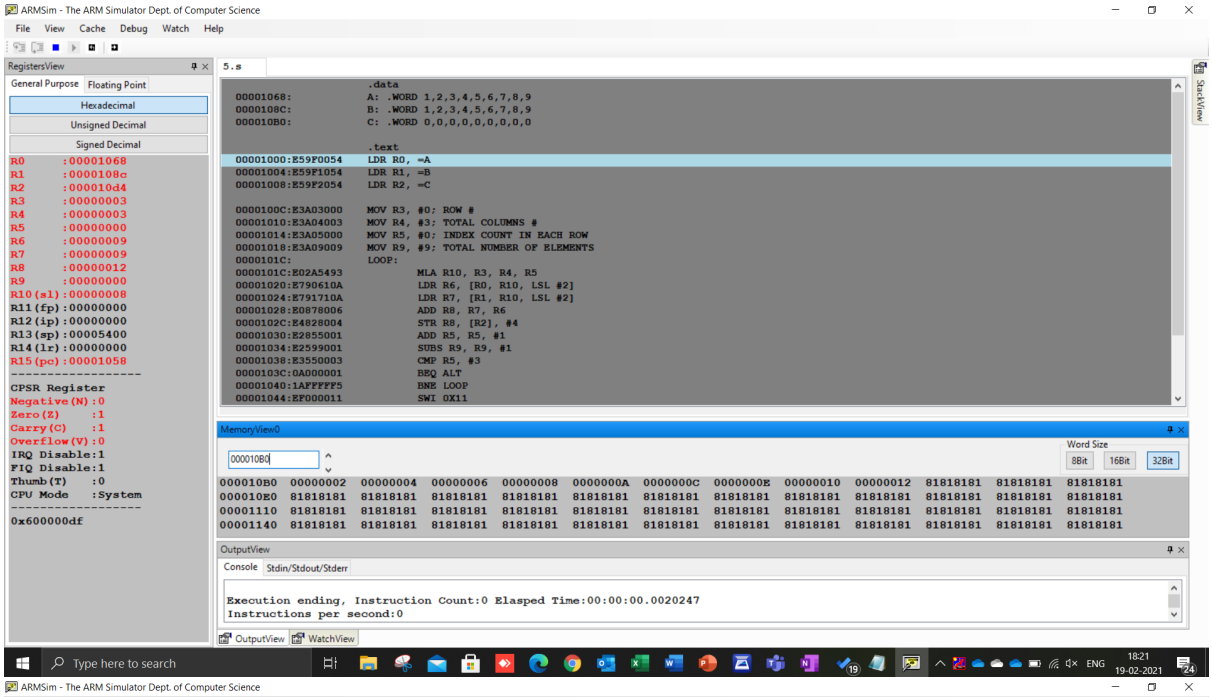
Title -Write an ALP to implement C[i][j]=a[i][j]+b[i][j]

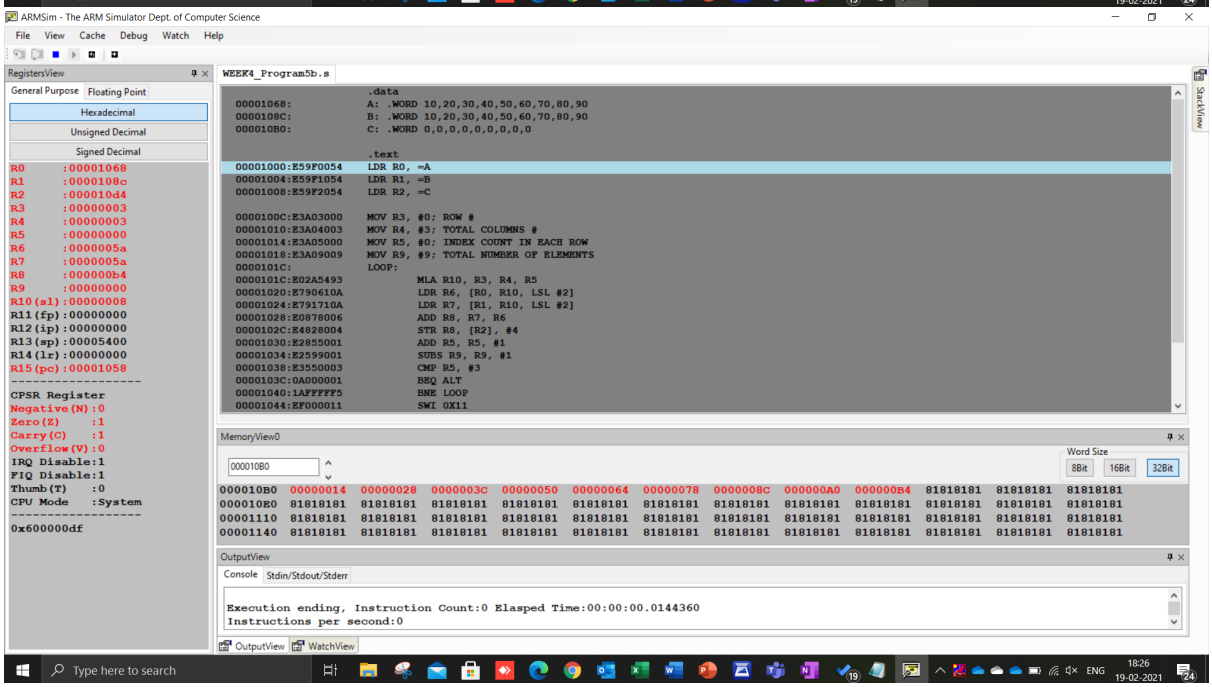
I. ARM Assembly Code

```
.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 R3, #0;
MOV R4, #3;
MOV R5, #0;
MOV R9, #9;
LOOP:
        MLA R10, R3, R4, R5
        LDR R6, [R0, R10, LSL #2]
        LDR R7, [R1, R10, LSL #2]
        ADD R8, R7, R6
        STR R8, [R2], #4
        ADD R5, R5, #1
        SUBS R9, R9, #1
        CMP R5, #3
        BEQ ALT
        BNE LOOP
        SWI 0X11
ALT:
        MOV R5, #0
        ADD R3, R3, #1
        CMP R9, #0
        BNE LOOP
        SWI 0X11
```

```
.data
A: .WORD 10,20,30,40,50,60,70,80,90
B: .WORD 10,20,30,40,50,60,70,80,90
C: .WORD 0,0,0,0,0,0,0,0,0

.text
LDR R0, =A
LDR R1, =B
LDR R2, =C

MOV R3, #0;
MOV R4, #3;
MOV R5, #0;
MOV R9, #9;
LOOP:
        MLA R10, R3, R4, R5
        LDR R6, [R0, R10, LSL #2]
        LDR R7, [R1, R10, LSL #2]
        ADD R8, R7, R6
        STR R8, [R2], #4
        ADD R5, R5, #1
        SUBS R9, R9, #1
        CMP R5, #3
        BEQ ALT
        BNE LOOP
        SWI 0X11
ALT:
        MOV R5, #0
        ADD R3, R3, #1
        CMP R9, #0
        BNE LOOP
        SWI 0X11
```

II. Output Screen Shot


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 :00001068
R1 :0000108c
R2 :000010d4
R3 :00000003
R4 :00000003
R5 :00000000
R6 :00000009
R7 :00000009
R8 :00000012
R9 :00000000
R10(sl) :00000008
R11(fp) :00000000
R12(ip) :00000000
R13(sp) :00005400
R14(lr) :00000000
R15(pc) :00001058
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.s
.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 R3, #0; ROW #
MOV R4, #3; TOTAL COLUMNS #
MOV R5, #0; INDEX COUNT IN EACH ROW
MOV R9, #9; TOTAL NUMBER OF ELEMENTS
LOOP:
MLA R10, R3, R4, R5
LDR R6, [R0, R10, LSL #2]
LDR R7, [R1, R10, LSL #2]
ADD R8, R7, R6
STR R8, [R2], #4
ADD R5, R5, #1
SUBS R9, R9, #1
CMP R5, #3
BEQ ALT
BNE LOOP
SWI 0X11
MemoryView0
000010B0
Word Size
8Bit 16Bit 32Bit
000010B0 00000002 00000004 00000006 00000008 0000000A 0000000C 0000000E 00000010 00000012 81818181 81818181 81818181
000010E0 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001110 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001140 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0020247
Instructions per second:0
OutputView WatchView
Type here to search
ENG 18:21 19-02-2021


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 :00001068
R1 :0000108c
R2 :000010d4
R3 :00000003
R4 :00000003
R5 :00000000
R6 :0000005a
R7 :0000005a
R8 :000000b4
R9 :00000000
R10(sl) :00000008
R11(fp) :00000000
R12(ip) :00000000
R13(sp) :00005400
R14(lr) :00000000
R15(pc) :00001058
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
WEEK4_Program5b.s
.data
A: .WORD 10,20,30,40,50,60,70,80,90
B: .WORD 10,20,30,40,50,60,70,80,90
C: .WORD 0,0,0,0,0,0,0,0,0
.text
LDR R0, =A
LDR R1, =B
LDR R2, =C
MOV R3, #0; ROW #
MOV R4, #3; TOTAL COLUMNS #
MOV R5, #0; INDEX COUNT IN EACH ROW
MOV R9, #9; TOTAL NUMBER OF ELEMENTS
LOOP:
MLA R10, R3, R4, R5
LDR R6, [R0, R10, LSL #2]
LDR R7, [R1, R10, LSL #2]
ADD R8, R7, R6
STR R8, [R2], #4
ADD R5, R5, #1
SUBS R9, R9, #1
CMP R5, #3
BEQ ALT
BNE LOOP
SWI 0X11
MemoryView0
000010B0
Word Size
8Bit 16Bit 32Bit
000010B0 00000014 00000028 0000003C 00000050 00000064 00000078 0000008C 000000A0 000000B4 81818181 81818181 81818181
000010E0 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001110 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
00001140 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0144360
Instructions per second:0
OutputView WatchView
Type here to search
ENG 18:26 19-02-2021

III. Output Table for the program

Before execution	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
After Execution	00000001	00000001	00000002
	00000002	00000002	00000004
	00000003	00000003	00000006
	00000004	00000004	00000008
	00000005	00000005	0000000A
	00000006	00000006	0000000C
	00000007	00000007	0000000E
	00000008	00000008	00000010
	00000009	00000009	00000012

Before execution	a:.word 10,20,30,40,50,60,70,80,90	b:.word 10,20,30,40,50,60,70,80,90	c:.word 0
After Execution	0000000A	0000000A	00000014
	00000014	00000014	00000028
	0000001E	0000001E	0000003C
	00000028	00000028	00000050
	00000032	00000032	00000064

	0000003C	**0000003C**	**00000078**
	00000046	**00000046**	**0000008C**
	00000050	**00000050**	**000000A0**
	0000005A	**0000005A**	**00000084**

Week#4 Program Number:6

Title : **Write an ALP to implement Sum[i] +=a[i][j]**

I. ARM Assembly Code

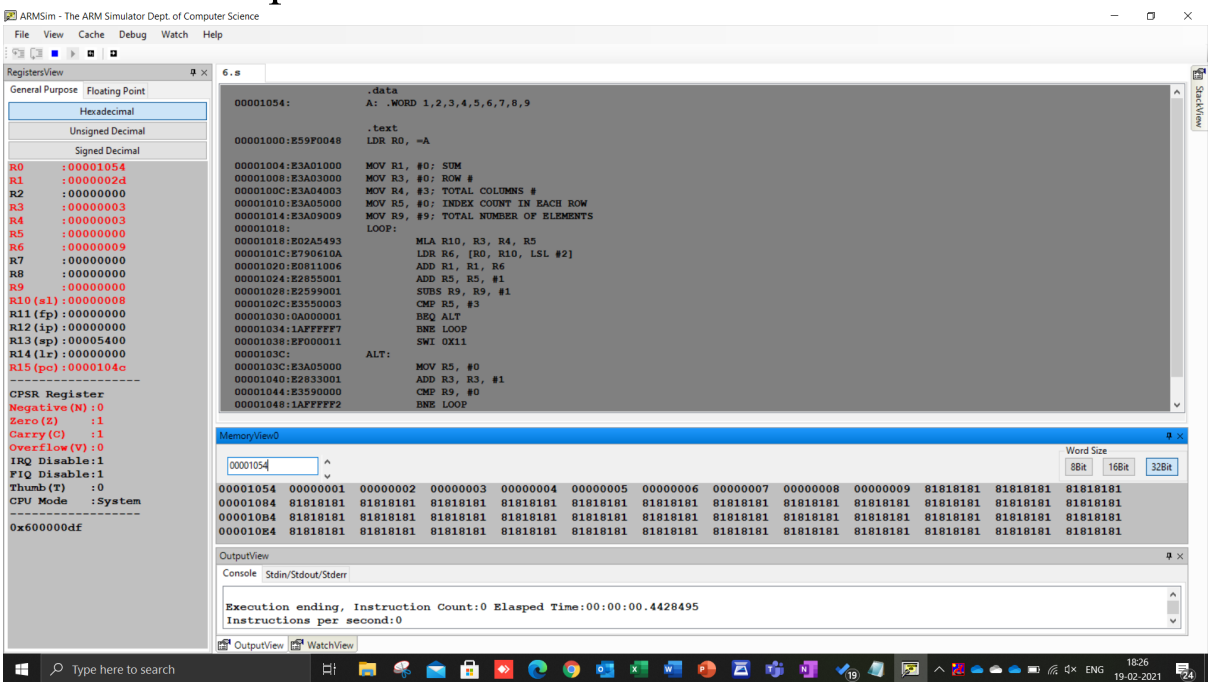
6.s

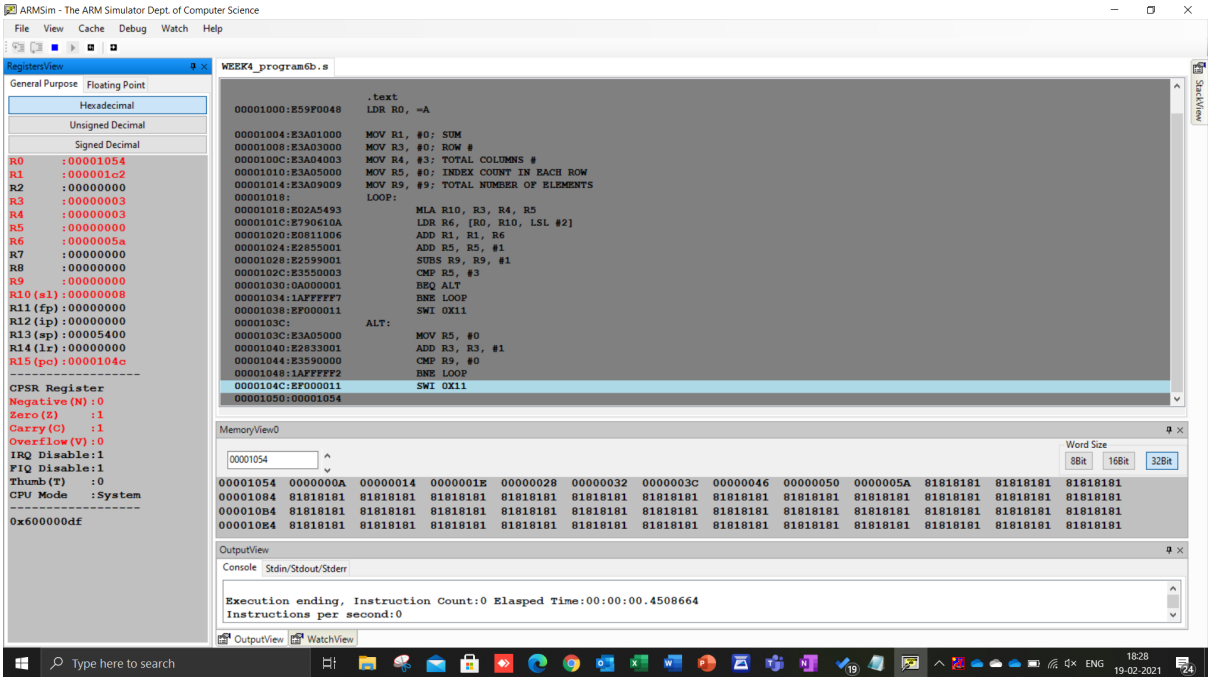
```
.data
A: .WORD 1,2,3,4,5,6,7,8,9

.text
LDR R0, =A

MOV R1, #0; SUM
MOV R3, #0; ROW #
MOV R4, #3; TOTAL COLUMNS #
MOV R5, #0; INDEX COUNT IN EACH ROW
MOV R9, #9; TOTAL NUMBER OF ELEMENTS
LOOP:
        MLA R10, R3, R4, R5
        LDR R6, [R0, R10, LSL #2]
        ADD R1, R1, R6
        ADD R5, R5, #1
        SUBS R9, R9, #1
        CMP R5, #3
        BEQ ALT
        BNE LOOP
        SWI 0X11
ALT:
        MOV R5, #0
        ADD R3, R3, #1
        CMP R9, #0
        BNE LOOP
        SWI 0X11
```

WEEK4_program6b.s

```
.data
A: .WORD 10,20,30,40,50,60,70,80,90

.text
LDR R0, =A

MOV R1, #0; SUM
MOV R3, #0; ROW #
MOV R4, #3; TOTAL COLUMNS #
MOV R5, #0; INDEX COUNT IN EACH ROW
MOV R9, #9; TOTAL NUMBER OF ELEMENTS
LOOP:
        MLA R10, R3, R4, R5
        LDR R6, [R0, R10, LSL #2]
        ADD R1, R1, R6
        ADD R5, R5, #1
        SUBS R9, R9, #1
        CMP R5, #3
        BEQ ALT
        BNE LOOP
        SWI 0X11
ALT:
        MOV R5, #0
        ADD R3, R3, #1
        CMP R9, #0
        BNE LOOP
        SWI 0X11
```

II. Output Screen Shot





III. Output Table for the program

Before execution	**a:.word 1,2,3,4,5,6,7,8,9**		
After Execution	**Addition result**	**45**	**2D**

Before execution	**a:.word 10,20,30,40,50,60,70,80,90**		
After Execution	**Addition result**	450	1c2

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: PRIYA MOHATA
Name: PRIYA MOHATA
SRN: PES2UG19CS301
Section: E
Date: 19.02.2021