

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year
2020-21

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

Week# 3 Program Number: 1

Write an ALP to add two 64 bit numbers loaded from memory and store the result in memory.

1. ARM Assembly Code for the program.

File Edit Format View Help

```
.text
LDR R0, =a
LDR R1, [R0]
LDR R2, [R0, #4]
LDR R0, =b
LDR R3, [R0]
LDR R4, [R0, #4]
ADDS R6, R2, R4
ADC R5, R1, R3
LDR R0, =c
STR R6, [R0]
.data
a: .word 12293543, 56659987
b: .word 98769432, 45323389
c: .word 0
```

II. Output Screen Shot (One Example of your choice)

The screenshot displays the ARMSim - The ARM Simulator interface. The main window is titled "WEEK3_1_PES20G19CS412.s". The left sidebar shows the "RegistersView" panel with a list of registers (R0-R15) and their values in hexadecimal. The "CPSR Register" section shows various flags (Negative, Zero, Carry, Overflow) and CPU mode (System). The main area displays the assembly code for the file, including instructions like LDR, ADDS, ADC, and STR. The bottom panel shows the "OutputView" with a console window displaying the execution log, including the loading of the assembly file, execution starting, and execution ending with instruction count and elapsed time.

```
.text
00001000:E59F0020 LDR R0, =a
00001004:E5901000 LDR R1, [R0]
00001008:E5902004 LDR R2, [R0, #4]
0000100C:E59F0018 LDR R0, =b
00001010:E5903000 LDR R3, [R0]
00001014:E5904004 LDR R4, [R0, #4]
00001018:E0926004 ADDS R6, R2, R4
0000101C:E0A15003 ADC R5, R1, R3
00001020:E59F0008 LDR R0, =c
00001024:E5806000 STR R6, [R0]

.data
00001034: a: .word 12293543, 56659987
0000103C: b: .word 98769432, 45323389
00001044: c: .word 0
```

Execution log:

```
Loading assembly language file E:\MPCA_LAB\WEEK-3\WEEK3_1_PES20G19CS412.s
Execution starting ...
PC out of valid memory range, address:00011400
PC out of valid memory range, address:00011400

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.1549146
Instructions per second:0
```

III. Output Table for the program

<div> <div>a:</div> <div>.word</div> <div>12293543,</div> <div>56659987</div> </div> <div> <div>b:</div> <div>.word</div> <div>98769432,</div> <div>45323389</div> </div>		
	Upper 32 bits	Lower 32 bits
a: .word	56659987 (03609013)	12293543 (00bb95A7)
b: .word	4 5 3 2 3 3 8 9 (02B3947D)	98769432 (05E31A18)
c: .word	101984376 (06142878)	110977975 (069D63B7)

Microprocessor and Computer Architecture Laboratory

UE19CS256

**4th Semester, Academic Year
2020-21**

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

Week# 3 Program Number: 2

**Write an ALP to copy n numbers from Memory
Location A to Memory Location B**

I. ARM Assembly Code for the program.

```
File Edit Format View Help
.text
    LDR R0,=a;
LDR R1,=b;
MOV R3,#5;
LOOP:
LDR R5,[R1]
LDR R4,[R0];
ADD R0, R0, #4
STR R4,[R1]
ADD R1, R1, #4
SUBS R3, R3, #1
BNE LOOP
.data
a: .word 03, 05, 01, 04,02
b: .word 0, 0, 0, 0,0
```

II. Output Screen Shot (One Example of your choice)

The screenshot displays the ARMSim - The ARM Simulator interface. The main window shows the assembly code from the previous block, with memory addresses and hex values on the left. The left sidebar contains the 'RegistersView' panel, showing the state of registers R0 through R15 and the CPSR register. The bottom panel shows the 'OutputView' with a console window displaying the execution log.

```
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 :00000002
R5 :00000000
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):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

WEEK3_2_PES2UG19CS412.s
.text
00001000:E59F0020 LDR R0,=a;
00001004:E59F1020 LDR R1,=b;
00001008:E3A03005 MOV R3,#5;
0000100C: LOOP:
0000100C:E5915000 LDR R5,[R1]
00001010:E5904000 LDR R4,[R0];
00001014:E2800004 ADD R0, R0, #4
00001018:E5814000 STR R4,[R1]
0000101C:E2811004 ADD R1, R1, #4
00001020:E2533001 SUBS R3, R3, #1
00001024:1AFFFFF0 BNE LOOP
.data
00001030: a: .word 03, 05, 01, 04,02
00001044: b: .word 0, 0, 0, 0,0

OutputView
Console Stdin/Stdout/Stderr
Loading assembly language file E:\MPCA_LAB\WEEK-3\WEEK3_2_PES2UG19CS412.s
Execution starting ...
PC out of valid memory range, address:00011400
PC out of valid memory range, address:00011400

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.1569080
Instructions per second:0
```

III. Output Table for the program

.data

a: .word 3, 5, 1, 4,2

b: .word 0, 0, 0, 0,0

1 st Iteration	a: .word 03, 05, 01, 04,02 b: .word 03, 00, 00, 00,00
2 nd Iteration	a: .word 03, 05, 01, 04,02 b: .word 03, 05, 0, 0,0
3 rd Iteration	a: .word 03, 05, 01, 04,02 b: .word 03, 05, 01, 0,0
4 th Iteration	a: .word 03, 05, 01, 04,02 b: .word 03, 05, 01, 04,0
5 th Iteration	a: .word 03, 05, 01, 04,02 a: .word 03, 05, 01, 04,02

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year
2020-21

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

Week# 3 Program Number: 3

Write an ALP to find smallest number in an array of n 32 bit numbers

1. ARM Assembly Code for the program.

```
File Edit Format View Help
.text
LDR R0,=a;
MOV R1,#6
LDR R2,[R0]
LOOP:
SUBS R1,R1,#1
CMP R1,#0
BEQ END
ADD R0,R0,#4
LDR R3,[R0]
CMP R2,R3
BCC LOOP
MOV R2,R3
CMP R1,#0
BNE LOOP
a:.word 1,2,3,4,5, 6
END: .end
```

II. Output Screen Shot (One Example of your choice)

The screenshot displays the ARMSim - The ARM Simulator interface. The main window is titled "WEEK3_3_PES2UG19CS412.s". The left pane shows the "RegistersView" with a list of registers (R0-R15) and their values in hexadecimal. The right pane shows the assembly code for the file, including instructions like LDR, MOV, LDR, LOOP, SUBS, CMP, BEQ, ADD, and BCC. The bottom pane shows the "OutputView" with a console output that includes the loading of the assembly language file, execution starting, and execution ending with instruction count and elapsed time.

RegistersView

Register	Value
R0	:00001048
R1	:00000000
R2	:00000001
R3	:00000006
R4	:00000000
R5	:00000000
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)	: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

Assembly Code:

```
.text
00001000:E59F0044 LDR R0,=a;
00001004:E3AB1006 MOV R1,#6
00001008:E5902000 LDR R2,[R0]
0000100C:
0000100C:E2511001 SUBS R1,R1,#1
00001010:E3510000 CMP R1,#0
00001014:0A00000C BEQ END
00001018:E2800004 ADD R0,R0,#4
0000101C:E5903000 LDR R3,[R0]
00001020:E1520003 CMP R2,R3
00001024:3AFFFFF8 BCC LOOP
00001028:E1A02003 MOV R2,R3
0000102C:E3510000 CMP R1,#0
00001030:1AFFFFF5 BNE LOOP
00001034:
a: word 1,2,3,4,5, 6
0000104C:00001034 END: .end
```

OutputView

Console Stdin/Stdout/Stderr

```
Loading assembly language file E:\MPCA_LAB\WEEK-3\WEEK3_3_PES2UG19CS412.s
Execution starting ...
PC out of valid memory range, address:00011400
PC out of valid memory range, address:00011400

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.1249270
Instructions per second:0
```


III. Output Table for the program

a:word 1,2,3,4,5, 6	
1 st Iteration	R2=1,R3=2 (R3>R2)
2 nd Iteration	R2=1, R3=3 (R3>R2)
3 rd Iteration	R2=1, R3=4 (R3>R2)
4 th Iteration	R2=1, R3=5 (R3>R2)
5 th Iteration	R2=1, R3=6 (R3>R2)
Smallest number is present in R2	

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year
2020-21

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

Week# 3 Program Number: 4a

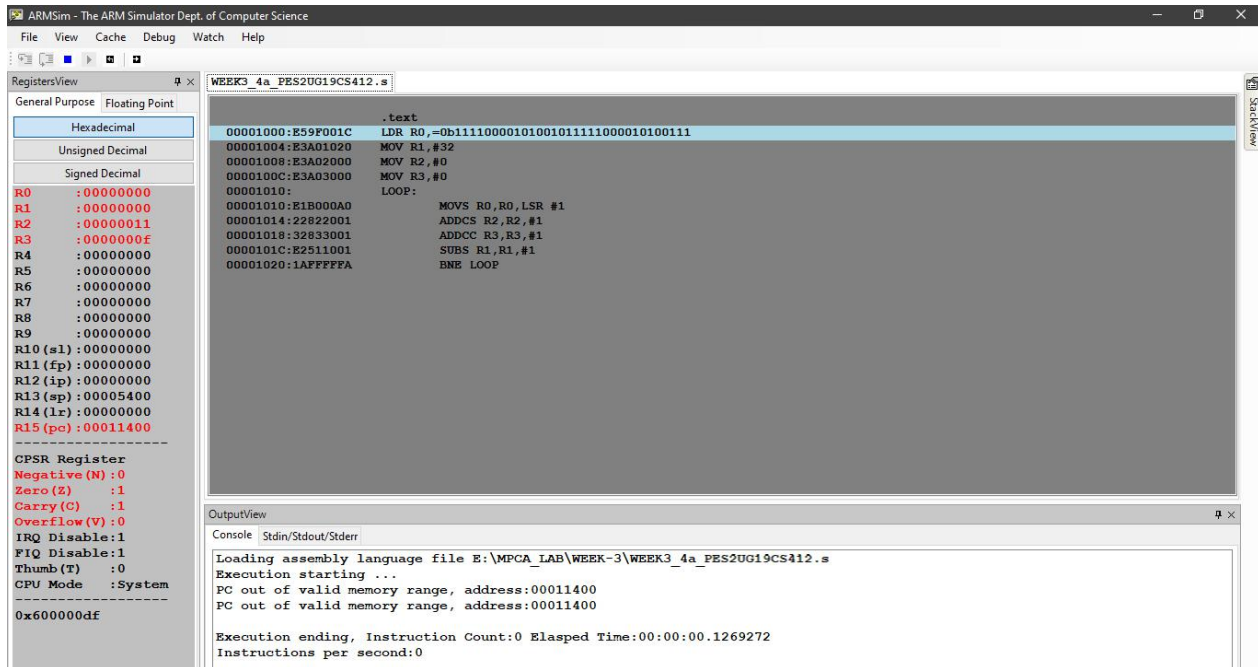
**Write an ALP to count the number of 1's
and 0's in a given 32 bit number.**

I. ARM Assembly Code for the program.

File Edit Format View Help

```
.text
LDR R0,=0b11110000101001011111000010100111
MOV R1,#32
MOV R2,#0
MOV R3,#0
LOOP:
    MOVS R0,R0,LSR #1
    ADDCS R2,R2,#1
    ADDCC R3,R3,#1
    SUBS R1,R1,#1
    BNE LOOP
```

II. Output Screen Shot (One Example of your choice)



III. Output Table for the program

r0, =0b11110000101001011111000010100101		
r1	32	
r2	After execution	17 (=0F in hex)
r3	After execution	16 (=11 in hex)

Microprocessor and Computer Architecture Laboratory

UE19CS256

**4th Semester, Academic Year
2020-21**

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

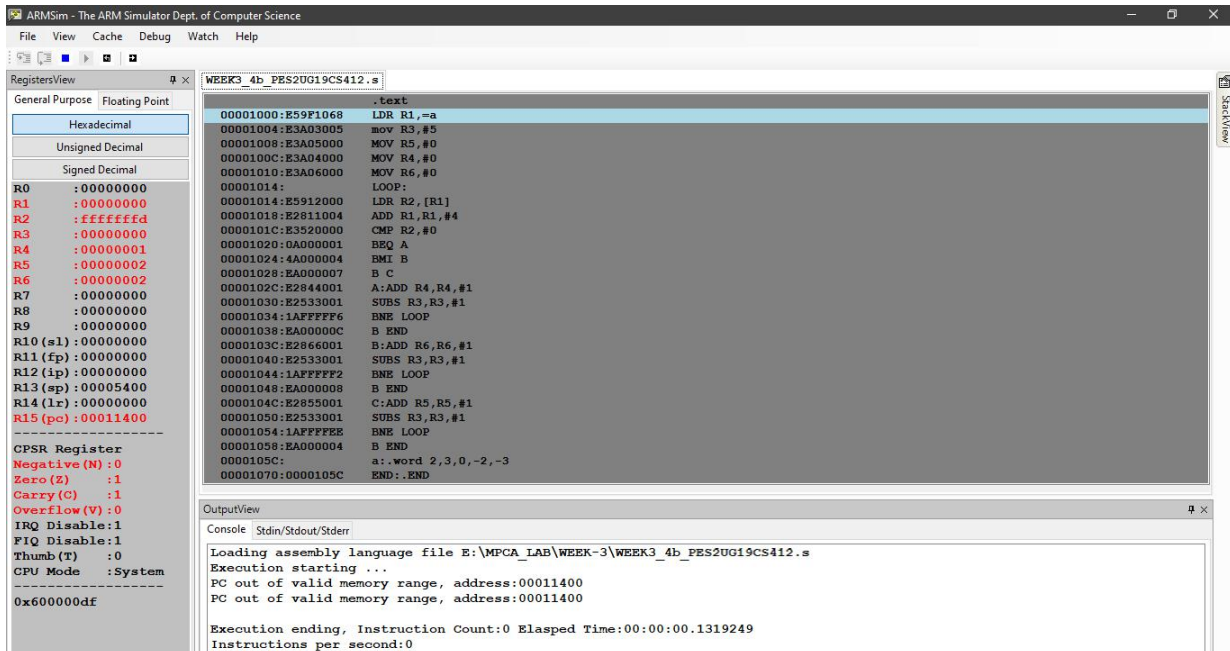
Week# 3 Program Number: 4b

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

I. ARM Assembly Code for the program.

```
File Edit Format View Help
|.text
LDR R1,=a
mov R3,#5
MOV R5,#0
MOV R4,#0
MOV R6,#0
LOOP:
LDR R2,[R1]
ADD R1,R1,#4
CMP R2,#0
BEQ A
BMI B
B C
A:ADD R4,R4,#1
SUBS R3,R3,#1
BNE LOOP
B END
B:ADD R6,R6,#1
SUBS R3,R3,#1
BNE LOOP
B END
C:ADD R5,R5,#1
SUBS R3,R3,#1
BNE LOOP
B END
a:.word 2,3,0,-2,-3
END:.END
```

II. Output Screen Shot (One Example of your choice)



The screenshot displays the ARMSim ARM Simulator interface. The main window shows the assembly code being executed, with the following instructions visible:

```
.text
LDR R1,=a
mov R3,#5
MOV R5,#0
MOV R4,#0
MOV R6,#0
LOOP:
LDR R2,[R1]
ADD R1,R1,#4
CMP R2,#0
BEQ A
BMI B
B C
A:ADD R4,R4,#1
SUBS R3,R3,#1
BNE LOOP
B END
B:ADD R6,R6,#1
SUBS R3,R3,#1
BNE LOOP
B END
C:ADD R5,R5,#1
SUBS R3,R3,#1
BNE LOOP
B END
a:.word 2,3,0,-2,-3
END:.END
```

The RegistersView window on the left shows the state of the registers:

Register	Value
R0	00000000
R1	00000000
R2	fffffffd
R3	00000000
R4	00000001
R5	00000002
R6	00000002
R7	00000000
R8	00000000
R9	00000000
R10 (s1)	00000000
R11 (fp)	00000000
R12 (ip)	00000000
R13 (sp)	00005400
R14 (lr)	00000000
R15 (pc)	00011400

The CPSR Register shows the following status:

Flag	Value
Negative (N)	0
Zero (Z)	1
Carry (C)	1
Overflow (V)	0
IRQ Disable	1
FIQ Disable	1
Thumb (T)	0
CPU Mode	System

The OutputView window shows the execution log:

```
Loading assembly language file E:\MPCA LAB\WEEK-3\WEEK3_4b_PES2UG19CS412.s
Execution starting ...
PC out of valid memory range, address:00011400
PC out of valid memory range, address:00011400

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.1319249
Instructions per second:0
```

III. Output Table for the program

a:.word 2,3,0,-2,-3		
R4	1	
R5	2	
R6	2	

Microprocessor and Computer Architecture Laboratory

UE19CS256

**4th Semester, Academic Year
2020-21**

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

Week# 3 Program Number: 5

Write an ALP to check whether a given number is present in array using Linear Search (Without SWI 0x02), if found move +1 to R6 and key position to R7

else move -1 to R6 (if number not found)

1. ARM Assembly Code for the program.

A)

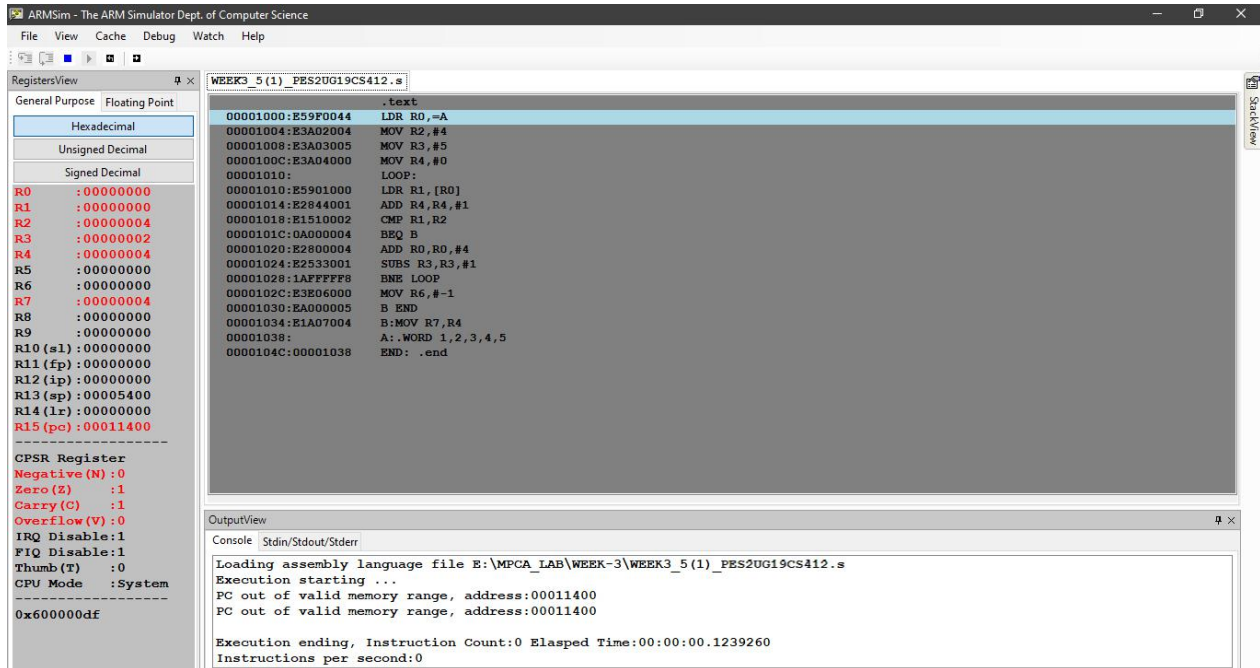
```
File Edit Format View Help
.text
LDR R0,=A
MOV R2,#4
MOV R3,#5
MOV R4,#0
LOOP:
LDR R1,[R0]
ADD R4,R4,#1
CMP R1,R2
BEQ B
ADD R0,R0,#4
SUBS R3,R3,#1
BNE LOOP
MOV R6,#-1
B END
B:MOV R7,R4
A:.WORD 1,2,3,4,5
END: .end
```


B)

```
File Edit Format View Help
.text
LDR R0,=A
MOV R2,#6
MOV R3,#5
MOV R4,#0
LOOP:
LDR R1,[R0]
ADD R4,R4,#1
CMP R1,R2
BEQ B
ADD R0,R0,#4
SUBS R3,R3,#1
BNE LOOP
MOV R6,#-1
B END
B:MOV R7,R4
A:.WORD 1,2,3,4,5
END: .end
```

II. Output Screen Shot (One Example of your choice)

A)



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 : 00000004
R3 : 00000002
R4 : 00000004
R5 : 00000000
R6 : 00000000
R7 : 00000004
R8 : 00000000
R9 : 00000000
R10 (s1) : 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

WEEK3_5(1)_PES2UG19CS412.s

```
.text
00001000:E59F0044 LDR R0,=A
00001004:E3A02004 MOV R2,#4
00001008:E3A03005 MOV R3,#5
0000100C:E3A04000 MOV R4,#0
00001010: LOOP:
00001010:E5901000 LDR R1,[R0]
00001014:E2844001 ADD R4,R4,#1
00001018:E1510002 CMP R1,R2
0000101C:0A000004 BEQ B
00001020:E2800004 ADD R0,R0,#4
00001024:E2533001 SUBS R3,R3,#1
00001028:1AFFFFF8 BNE LOOP
0000102C:E3E06000 MOV R6,#-1
00001030:EA000005 B END
00001034:E1A07004 B:MOV R7,R4
00001038: A: .WORD 1,2,3,4,5
0000104C:00001038 END: .end
```

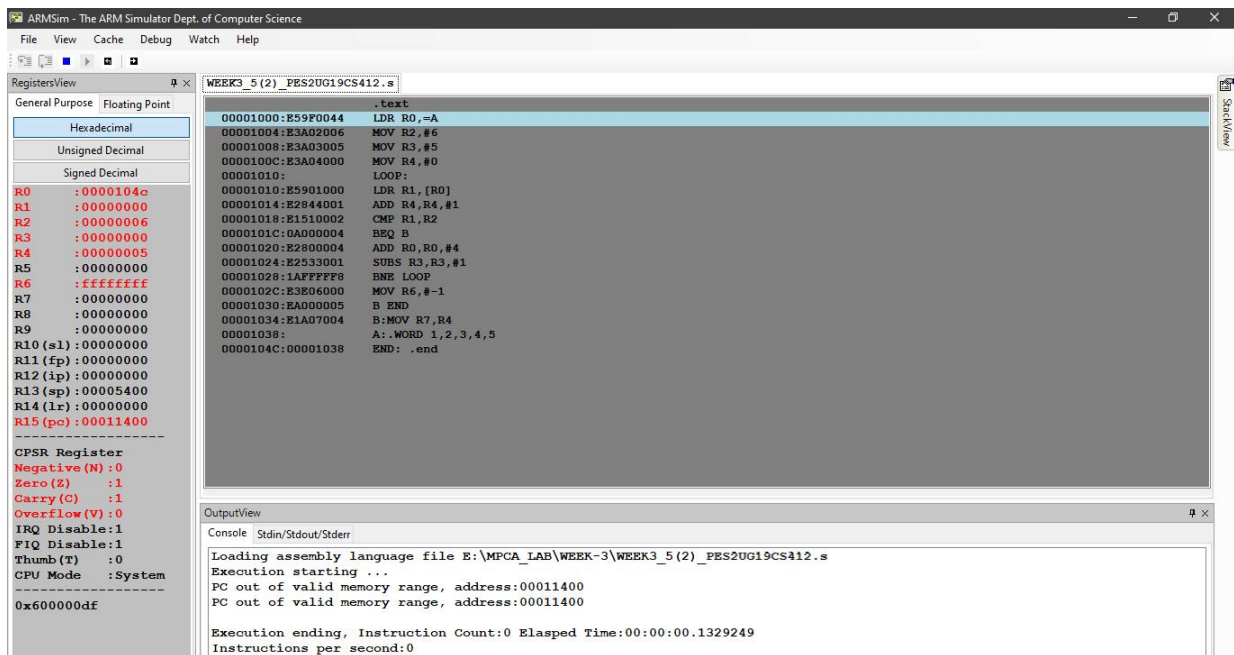
OutputView

Console Stdin/Stdout/Stderr

Loading assembly language file E:\MPCA_LAB\WEEK-3\WEEK3_5(1)_PES2UG19CS412.s
Execution starting ...
PC out of valid memory range, address:00011400
PC out of valid memory range, address:00011400

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.1239260
Instructions per second:0

B)



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 : 00001040
R1 : 00000000
R2 : 00000006
R3 : 00000000
R4 : 00000000
R5 : 00000000
R6 : ffffffff
R7 : 00000000
R8 : 00000000
R9 : 00000000
R10 (s1) : 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

WEEK3_5(2)_PES2UG19CS412.s

```
.text
00001000:E59F0044 LDR R0,=A
00001004:E3A02006 MOV R2,#6
00001008:E3A03005 MOV R3,#5
0000100C:E3A04000 MOV R4,#0
00001010: LOOP:
00001010:E5901000 LDR R1,[R0]
00001014:E2844001 ADD R4,R4,#1
00001018:E1510002 CMP R1,R2
0000101C:0A000004 BEQ B
00001020:E2800004 ADD R0,R0,#4
00001024:E2533001 SUBS R3,R3,#1
00001028:1AFFFFF8 BNE LOOP
0000102C:E3E06000 MOV R6,#-1
00001030:EA000005 B END
00001034:E1A07004 B:MOV R7,R4
00001038: A: .WORD 1,2,3,4,5
0000104C:00001038 END: .end
```

OutputView

Console Stdin/Stdout/Stderr

Loading assembly language file E:\MPCA_LAB\WEEK-3\WEEK3_5(2)_PES2UG19CS412.s
Execution starting ...
PC out of valid memory range, address:00011400
PC out of valid memory range, address:00011400

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.1329249
Instructions per second:0

III. Output Table for the program

		HEX value
A:.WORD 1,2,3,4,5		
R2	KEY =3	03
R3	COUNT =5	
R0	Address of A	00001038
R7	After Execution =3	Position of key element =3

		HEX value
A:.WORD 1,2,3,4,5		
R2	KEY =0	00
R3	COUNT =5	
R0	Address of A	00001038
R6	After Execution =-1 (Decimal Form)	Position of key element =Not found

Microprocessor and Computer Architecture Laboratory

UE19CS256

**4th Semester, Academic Year
2020-21**

Date: 10/02/2021

Name: SUHAN B REVANKAR	SRN:PES2UG19CS412	Section:G
------------------------	-------------------	-----------

Week# 3 Program Number: 6

**Write an ALP to generate Fibonacci Series
and store them in an array**

I. ARM Assembly Code for the program.

```
File Edit Format View Help
.text
MOV R0,#4;
LDR R1,=A;
MOV R2,#00;
MOV R3,#01;
LOOP:
    STR R4, [R1], #4
    SUBS R0, R0, #1
    ADD R4, R2, R3
    MOV R2, R3
    MOV R3, R4
    BNE LOOP
    SWI 0X011

.DATA
A: .WORD 0
```

II. Output Screen Shot (One Example of your choice)

The screenshot displays the ARMSim interface. The main window shows the assembly code for 'WEEK3_6_PES2UG19CS412.s'. The code includes initialization of registers R0, R1, R2, and R3, followed by a loop that increments R4 and decrements R0 until R0 reaches zero, then executes a software interrupt (SWI 0X011). The output window shows the execution progress, including the loading of the assembly file, the start of execution, and the final state of the program.

```
WEEK3_6_PES2UG19CS412.s
.text
00001000:E3A00004 MOV R0,#4;
00001004:E59F1020 LDR R1,=A;
00001008:E3A02000 MOV R2,#00;
0000100C:E3A03001 MOV R3,#01;
00001010:      LOOP:
00001010:E4814004      STR R4, [R1], #4
00001014:E2500001      SUBS R0, R0, #1
00001018:E0824003      ADD R4, R2, R3
0000101C:E1A02003      MOV R2, R3
00001020:E1A03004      MOV R3, R4
00001024:1AFFFFF9      BNE LOOP
00001028:EF000011      SWI 0X011

.DATA
00001030:      A: .WORD 0

RegistersView
General Purpose Floating Point
Hexadecimal
Unsigned Decimal
Signed Decimal
R0 : 00000000
R1 : 00001040
R2 : 00000003
R3 : 00000005
R4 : 00000005
R5 : 00000000
R6 : 00000000
R7 : 00000000
R8 : 00000000
R9 : 00000000
R10 (s1): 00000000
R11 (fp): 00000000
R12 (ip): 00000000
R13 (sp): 00000000
R14 (lr): 00001020
R15 (pc): 00000008
-----
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 : Supervis
-----
0x60000d3

OutputView
Console Stdin/Stdout/Stderr
Loading assembly language file E:\MPCA LAB\WEEK-3\WEEK3_6_PES2UG19CS412.s
Execution starting ...
PC out of valid memory range, address:00000008

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.0634607
Instructions per second:0
```

III. Output Table for the program

R0	Fibonacci Count	4
R1	Address of A	
R2	Initially 0	
R3	Initially 1	
R4	1 st Iteration	0+1=1
R4	2 nd Iteration	1+1=2
R4	3 rd Iteration	2+1=3
R4	4 th Iteration	3+2=5

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: suhanbrevankar

Name: SUHAN B

REVANKAR

SRN: PES2UG19CS412

Section: G

Date: 10/02/2021