

# Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date: 08/2/21

Name: Adithya M S	SRN: PES1UG19CS027	Section: A
-------------------	--------------------	------------

Week# \_\_\_\_3\_\_\_\_

Program Number: \_\_\_\_1\_\_\_\_

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

**Code:**

.data

A:.word 31,63 ;6331

B:.word 63,19 ;1963

C:.word 0,0

.text

ldr r1,=A

ldr r2,=B

ldr r3,=C

ldr r4,[r1],#4

ldr r5,[r2],#4

adds r6,r4,r5

str r6,[r3],#4

ldr r7,[r1]

ldr r8,[r2]

adc r9,r8,r7

str r9,[r3]

swi 0x11

The screenshot displays the ARMSim ARM simulator interface. The main window is titled "Program1.s" and contains the following assembly code:

```
.data
0000103C:      A:.word 31,63 ;6331
00001044:      B:.word 63,19 ;1963
0000104C:      C:.word 0,0
.text
00001000:E59F1028  ldr r1,=A
00001004:E59F2028  ldr r2,=B
00001008:E59F3028  ldr r3,=C
0000100C:E4914004  ldr r4,[r1],#4
00001010:E4925004  ldr r5,[r2],#4
00001014:E0946005  adds r6,r4,r5
00001018:E4836004  str r6,[r3],#4
0000101C:E5917000  ldr r7,[r1]
00001020:E5928000  ldr r8,[r2]
00001024:E0A89007  adc r9,r8,r7
00001028:E5839000  str r9,[r3]
swi 0x11
```

The left sidebar shows the "RegistersView" with the following values:

Register	Value
R0	: 0
R1	: 4160
R2	: 4168
R3	: 4176
R4	: 31
R5	: 63
R6	: 94
R7	: 63
R8	: 19
R9	: 82
R10 (sl)	: 0
R11 (fp)	: 0
R12 (ip)	: 0
R13 (sp)	: 21504
R14 (lr)	: 0
R15 (pc)	: 4140

The "CPSR Register" section shows:

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

The "MemoryView3" section shows a memory dump starting at address 0000103C, with a word size of 32Bit. The memory contains the following values:

Address	Value
0000103C	0000001F 0000003F 0000003F 00000013 0000005E 00000052 81818181 81818181 81818181 81818181
00001064	81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000108C	81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010B4	81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

The "OutputView" section shows the following output:

```
Execution starting ...
Execution ending, Instruction Count:0 Elapsed Time:00:00:00
```

Week# \_\_\_\_3\_\_\_\_

Program Number: \_\_\_\_2\_\_\_\_

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

**Code:**

.data

A:.word 19,16,15,12,32

B:.word

.text

ldr r0,=A

ldr r1,=B

mov r2,#5

loop:

ldr r3,[r0],#4

str r3,[r1],#4

subs r2,r2,#1

cmp r2,#0

bne loop

swi 0x11

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 : 4160

R1 : 4180

R2 : 0

R3 : 32

R4 : 0

R5 : 0

R6 : 0

R7 : 0

R8 : 0

R9 : 0

R10 (s1) : 0

R11 (fp) : 0

R12 (ip) : 0

R13 (sp) : 21504

R14 (lr) : 0

R15 (pc) : 4128

-----

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

Program2.s

```
.data
0000102C:      A:.word 19,16,15,12,32
00001040:      B:.word
               .text
00001000:E59F001C      ldr r0,=A
00001004:E3A01D41      ldr r1,=B
00001008:E3A02005      mov r2,#5
0000100C:      loop:
0000100C:E4903004      ldr r3,[r0],#4
00001010:E4813004      str r3,[r1],#4
00001014:E2522001      subs r2,r2,#1
00001018:E3522000      cmp r2,#0
0000101C:1AFFFFFFFA      bne loop
               swi 0x11
```

MemoryView3

00001040

Word Size

8Bit 16Bit 32Bit

00001040	00000013	00000010	0000000F	0000000C	00000020	81818181	81818181	81818181	81818181	81818181
00001068	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181
00001090	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181
000010B8	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181

OutputView

Console Stdin/Stdout/Stderr

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.0059829

Instructions per second:0

OutputView WatchView

Week# \_\_\_\_3\_\_\_\_

Program Number: \_\_\_\_3\_\_\_\_

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

**Code:**

.data:

A:.word 12,3,43,21,34,61,1,90,14,32

.text:

ldr r0,=A

ldr r1,[r0],#4

mov r2,#9

loop:

ldr r3,[r0],#4

cmp r1,r3

bgt loop1

else:

sub r2,r2,#1

cmp r2,#0

bne loop

swi 0x11

loop1:

mov r1,r3

b else

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 : 4136  
R1 : 1  
R2 : 0  
R3 : 32  
R4 : 0  
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) : 4172

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

Program3.s

```
.data:  
00001000: A: .word 12,3,43,21,34,61,1,90,14,32  
00001028: .text:  
00001028:E3A00A01 ldr r0,-A  
0000102C:E4901004 ldr r1,[r0],#4  
00001030:E3A02009 mov r2,#9  
00001034: loop:  
00001034:E4903004 ldr r3,[r0],#4  
00001038:E1510003 cmp r1,r3  
0000103C:CA000003 bgt loop1  
00001040: else:  
00001040:E2422001 sub r2,r2,#1  
00001044:E3520000 cmp r2,#0  
00001048:1AFFFFFF9 bne loop  
0000104C:EF000011 swi 0x11  
00001050: loop1:
```

MemoryView3

Word Size  
8Bit 16Bit 32Bit

00001040

00001040	E2422001	E3520000	1AFFFFFF9	EF000011	E1A01003	EAFFFFFF9	00000000	81818181	81818181	81818181
00001068	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181
00001090	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181
000010B8	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181	81818181

OutputView

Console Stdin/Stdout/Stderr

Execution starting ...

Execution ending, Instruction Count:0 Elapsed Time:00:00:00

OutputView WatchView

Week#\_\_\_\_3\_\_\_\_\_

Program Number: \_\_\_\_4a\_\_\_\_

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

**Code:**

```
.data
```

```
A:.word 9
```

```
.text
```

```
ldr r0,=A
```

```
ldr r1,[r0]
```

```
mov r2,#0
```

```
loop:
```

```
    movs r1,r1,lsr #1
```

```
    addcs r2,r2,#1
```

```
    cmp r1,#0
```

```
    bne loop
```

```
rsb r3,r2,#32
```

```
swi 0x11
```

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 : 4136

R1 : 0

R2 : 2

R3 : 30

R4 : 0

R5 : 0

R6 : 0

R7 : 0

R8 : 0

R9 : 0

R10 (s1) : 0

R11 (fp) : 0

R12 (ip) : 0

R13 (sp) : 21504

R14 (lr) : 0

R15 (pc) : 4128

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

Program4a.s

```
.data
A: .word 9
.text
00001000:E59F001C ldr r0,=A
00001004:E5901000 ldr r1,[r0]
00001008:E3A02000 mov r2,#0
0000100C:
loop:
0000100C:E1B010A1 movs r1,r1,lsr #1
00001010:22822001 addcs r2,r2,#1
00001014:E3510000 cmp r1,#0
00001018:1AFFFFFFB hne loop
0000101C:E2623020 rab r3,r2,#32
00001020:EF000011 swi 0x11
00001024:00001028
```

MemoryView3

00001040

Word Size 8Bit 16Bit 32Bit

00001040 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

00001068 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

00001090 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

000010B8 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

OutputView

Console Stdin/Stdout/Stderr

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.0075264

Instructions per second:0

OutputView WatchView



Week# \_\_\_\_3\_\_\_\_

Program Number: \_\_\_\_4b\_\_\_\_

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

**Code:**

```
.data
```

```
A:.word 12,-3,21,0,-129
```

```
.text
```

```
ldr r0,=A
```

```
mov r1,#5
```

```
loop:
```

```
    ldr r2,[r0],#4
```

```
    cmp r2,#0
```

```
    beq zero
```

```
    bpl positive
```

```
    bmi negative
```

```
    else:
```

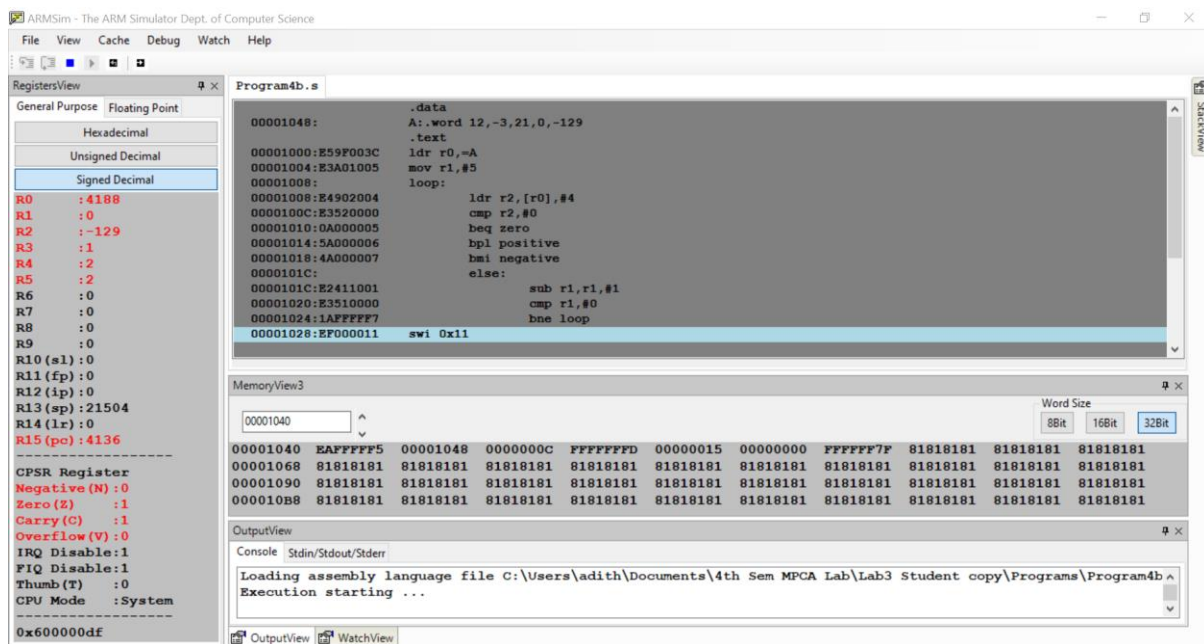
```
        sub r1,r1,#1
```

```
        cmp r1,#0
```

```
        bne loop
```

```
swi 0x11
```

b else



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)**

**Code:**

```
.data
```

```
A:.word 12,32,45,28,21
```

```
.text
```

```
ldr r0,=A
```

```
mov r1,#45 ;key element to be searched
```

```
mov r2,#5
```

```
loop:
```

```
    ldr r3,[r0],#4
```

```
    cmp r3,r1
```

```
    beq found
```

```
    sub r2,r2,#1
```

```
    cmp r2,#0
```

```
    bne loop
```

```
mov r6,#-1
```

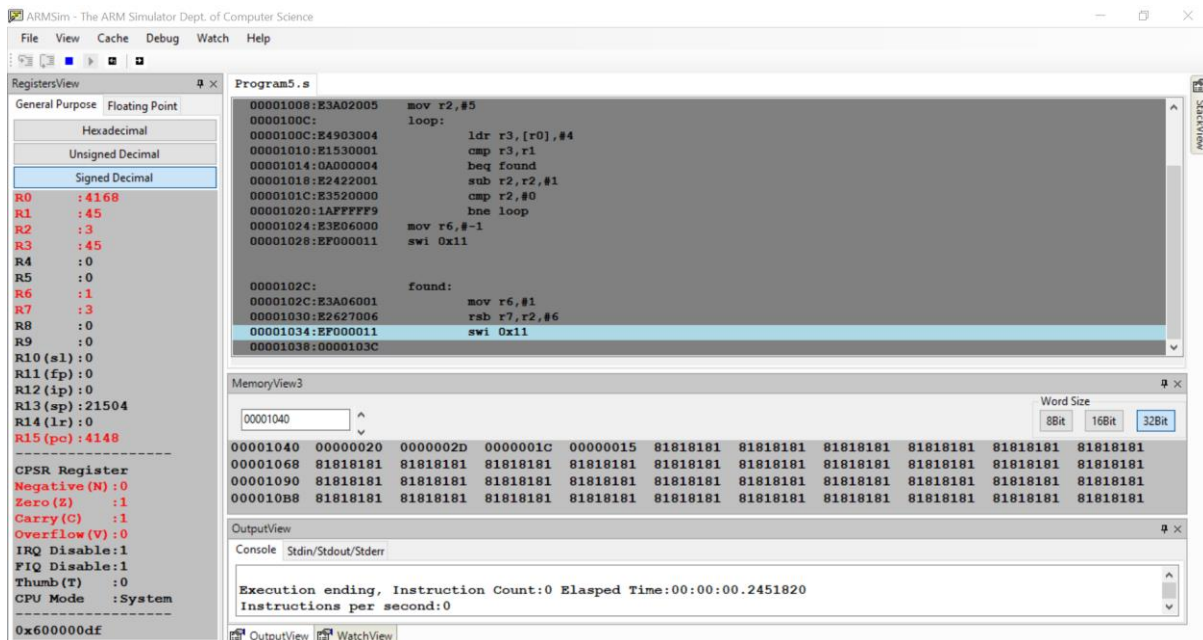
```
swi 0x11
```

found:

mov r6,#1

rsb r7,r2,#6

swi 0x11



Week#\_\_\_\_3\_\_\_\_\_

Program Number: \_\_\_\_6\_\_\_\_

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

**Code:**

.data

A:.word

.text

mov r0,#0

mov r1,#1

mov r4,#10

ldr r2,=A

str r0,[r2],#4

str r1,[r2],#4

loop:

add r3,r0,r1

str r3,[r2],#4

mov r0,r1

mov r1,r3

sub r4,r4,#1

cmp r4,#0

bne loop

# swi 0x11

The screenshot displays the ARMSim - The ARM Simulator interface. The main window shows the assembly code for Program6.s, which includes instructions like `mov r0,#0`, `ldr r2,-A`, and `swi 0x11`. The left sidebar shows the RegistersView with R0 through R15 and CPSR Register. The bottom section shows the MemoryView3 and OutputView.

**RegistersView**

Register	Value
R0	00000037
R1	00000059
R2	0000106c
R3	00000059
R4	00000000
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)	00001034

**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

**MemoryView3**

Word Size: 8Bit, 16Bit, 32Bit

Address	Value
0000103C	00000000 00000001 00000002 00000003 00000005 00000008 0000000D 00000015 00000022
00001064	00000037 00000059 81818181 81818181 81818181 81818181 81818181 81818181 81818181
0000108C	81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181
000010B4	81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181 81818181

**OutputView**

Console Stdin/Stdout/Stderr

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.2683080  
Instructions per second:0