**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

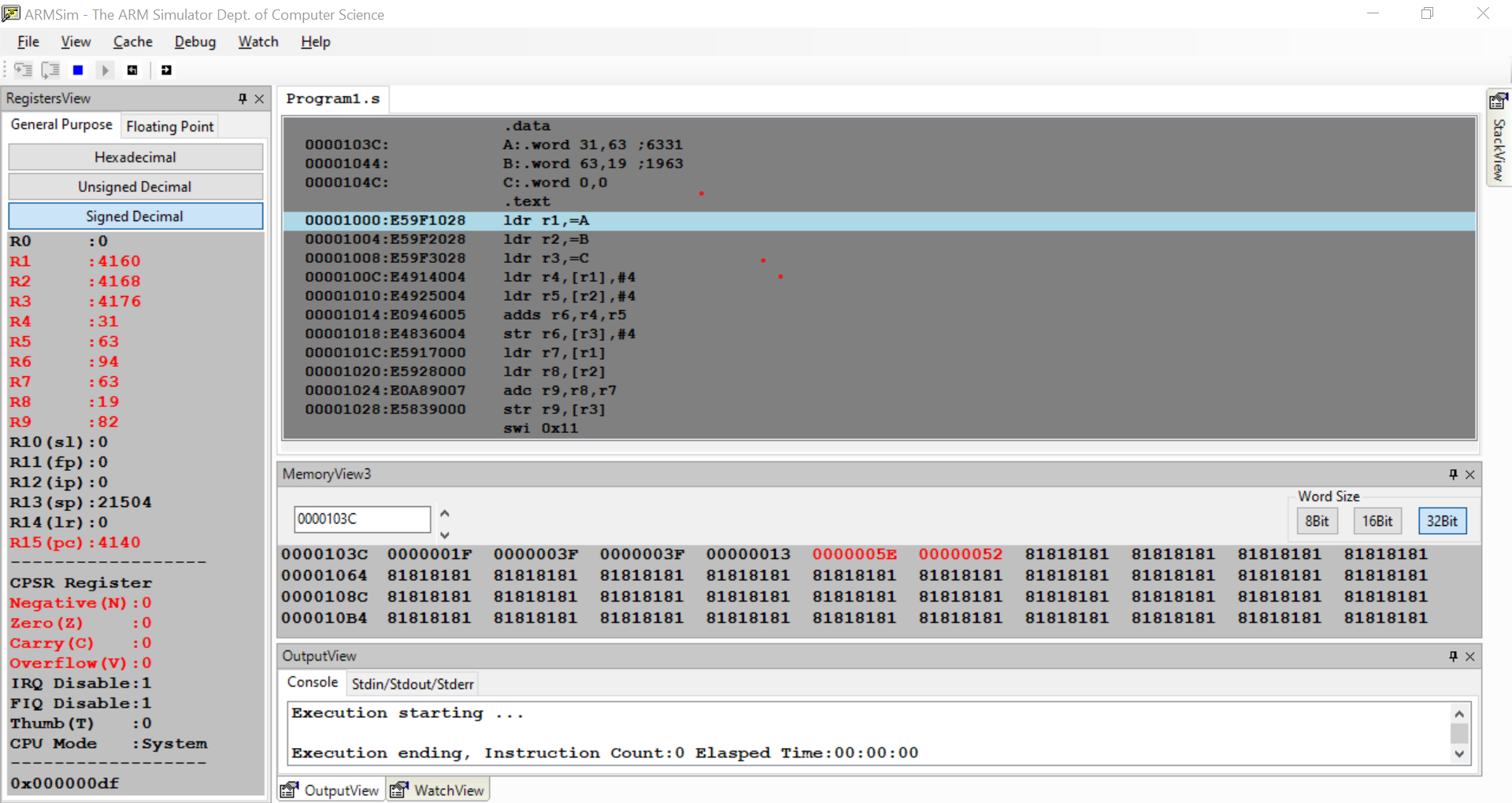
ldr r7,[r1]

ldr r8,[r2]

adc r9,r8,r7

str r9,[r3]

swi 0x11



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

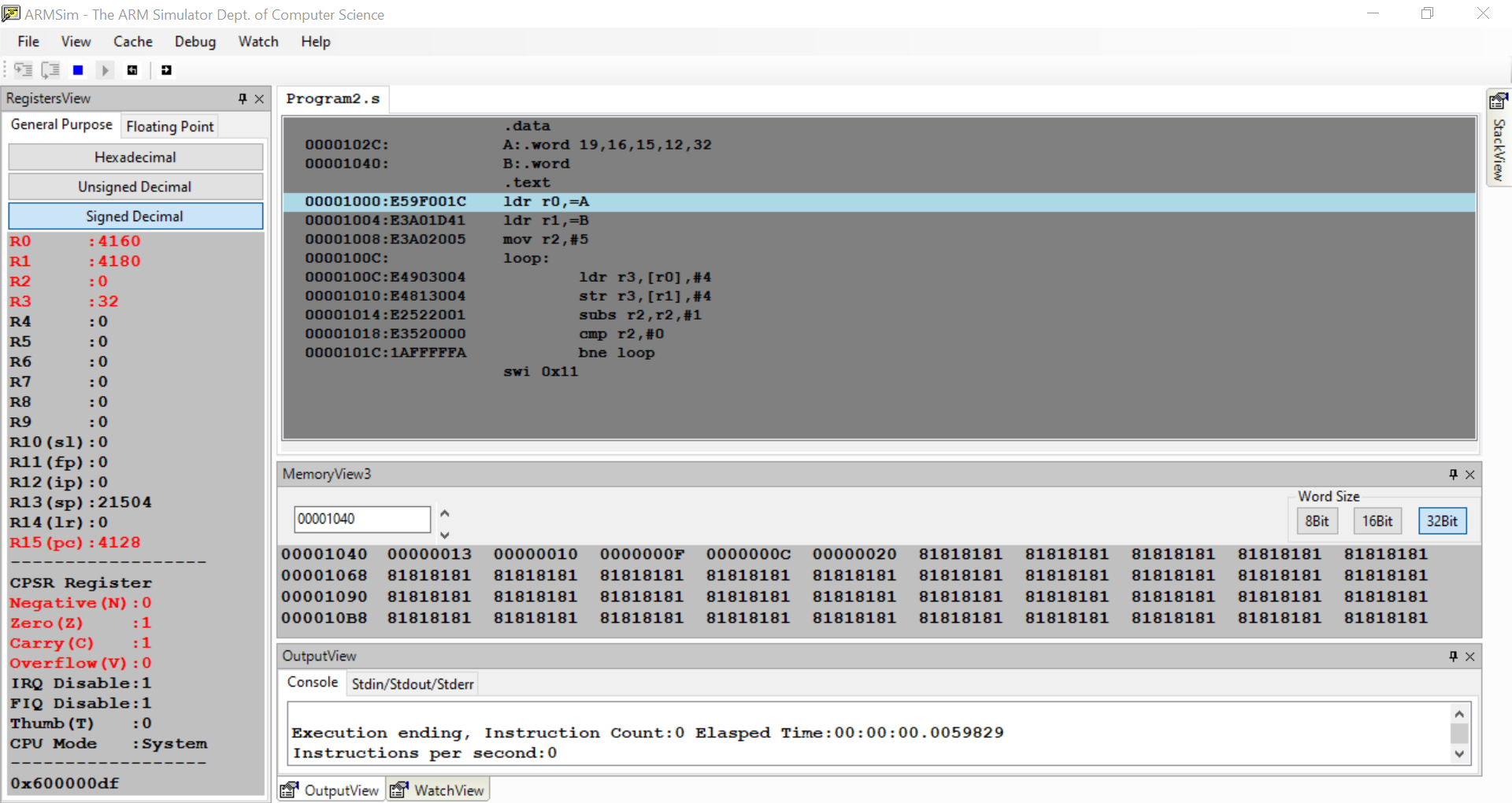
str r3,[r1],#4

subs r2,r2,#1

cmp r2,#0

bne loop

swi 0x11



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

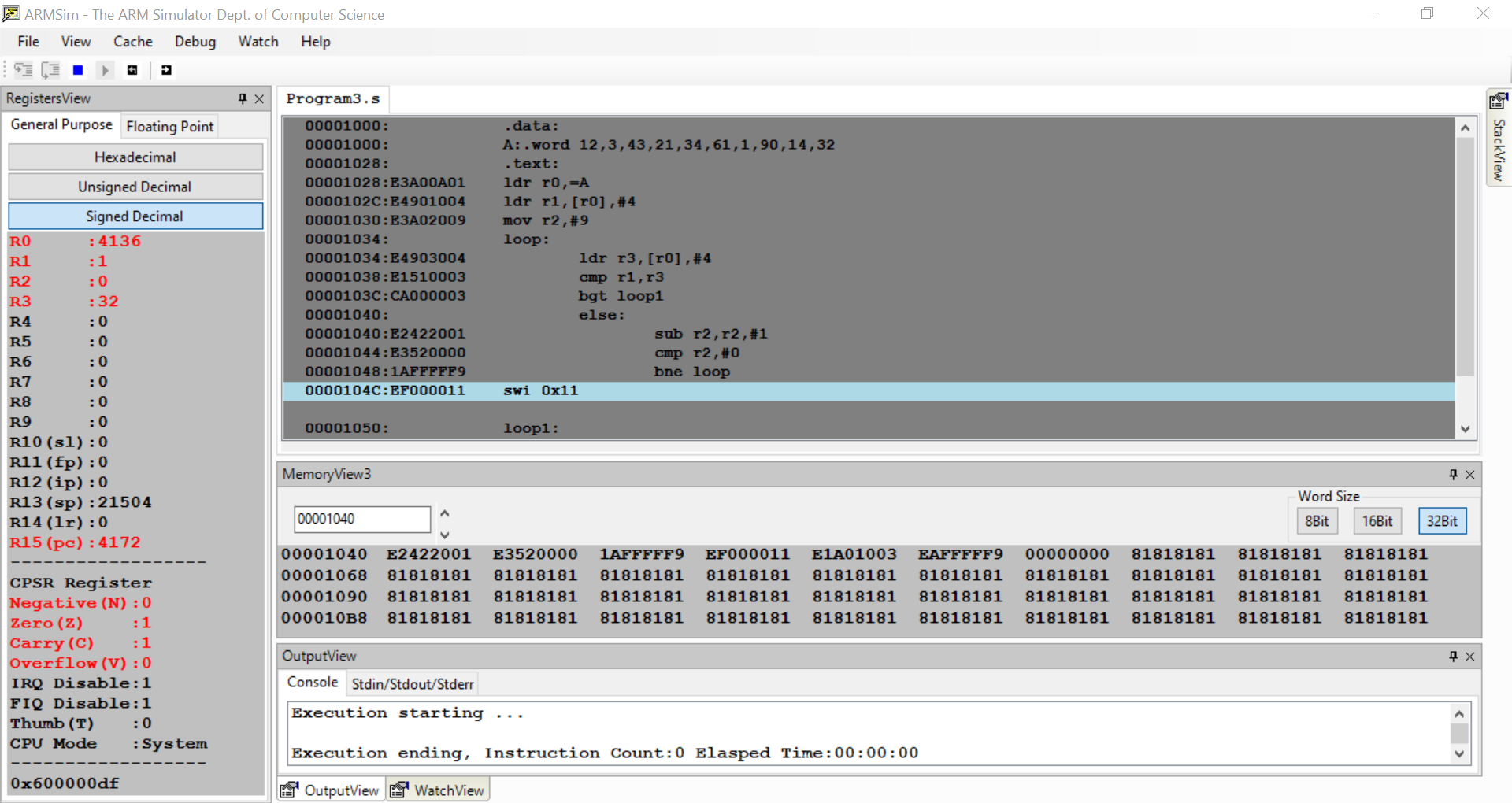
bne loop

swi 0x11

loop1:

mov r1,r3

b else



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

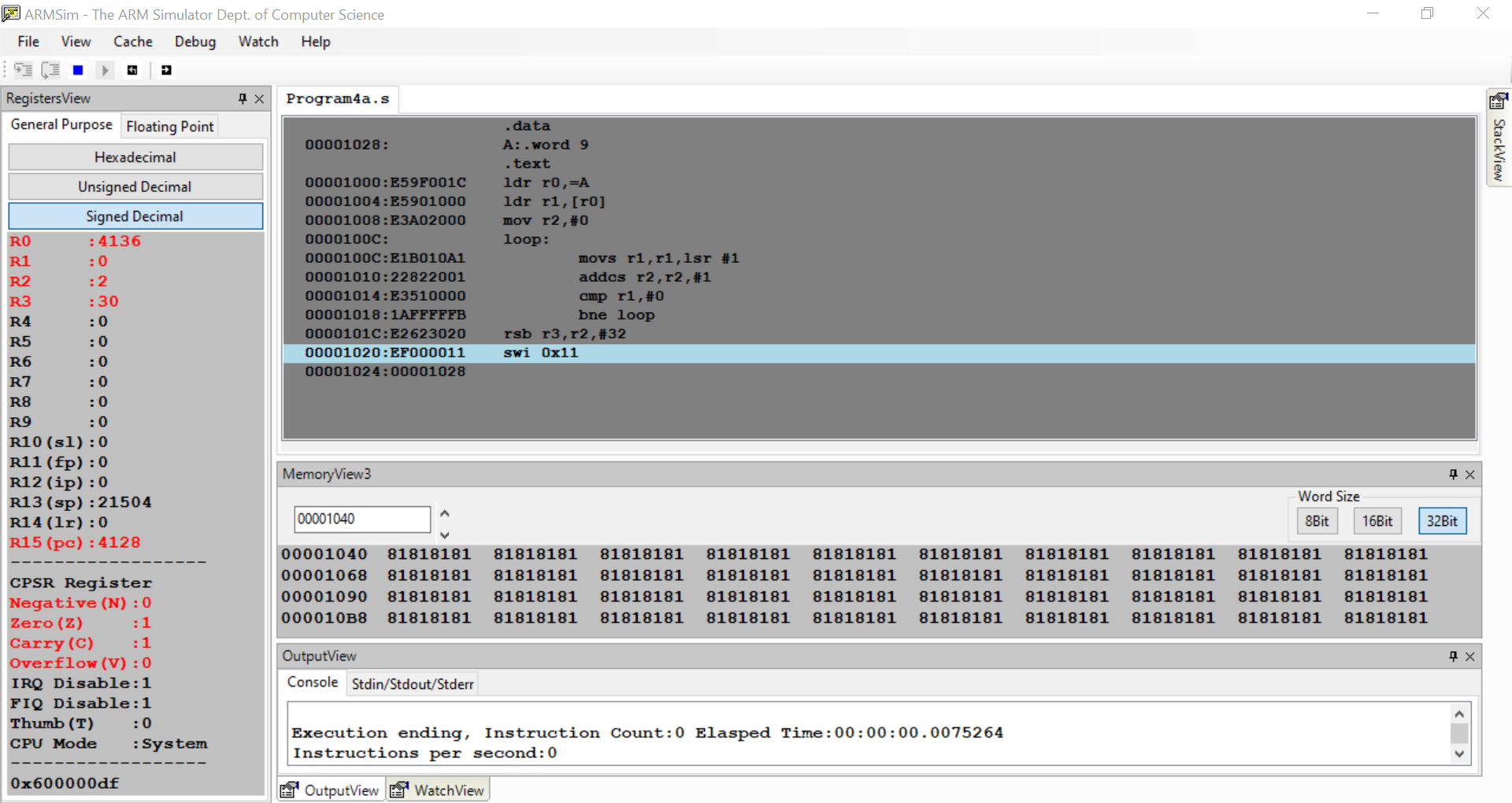
addcs r2,r2,#1

cmp r1,#0

bne loop

rsb r3,r2,#32

swi 0x11



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

zero:

add r3,r3,#1

b else

positive:

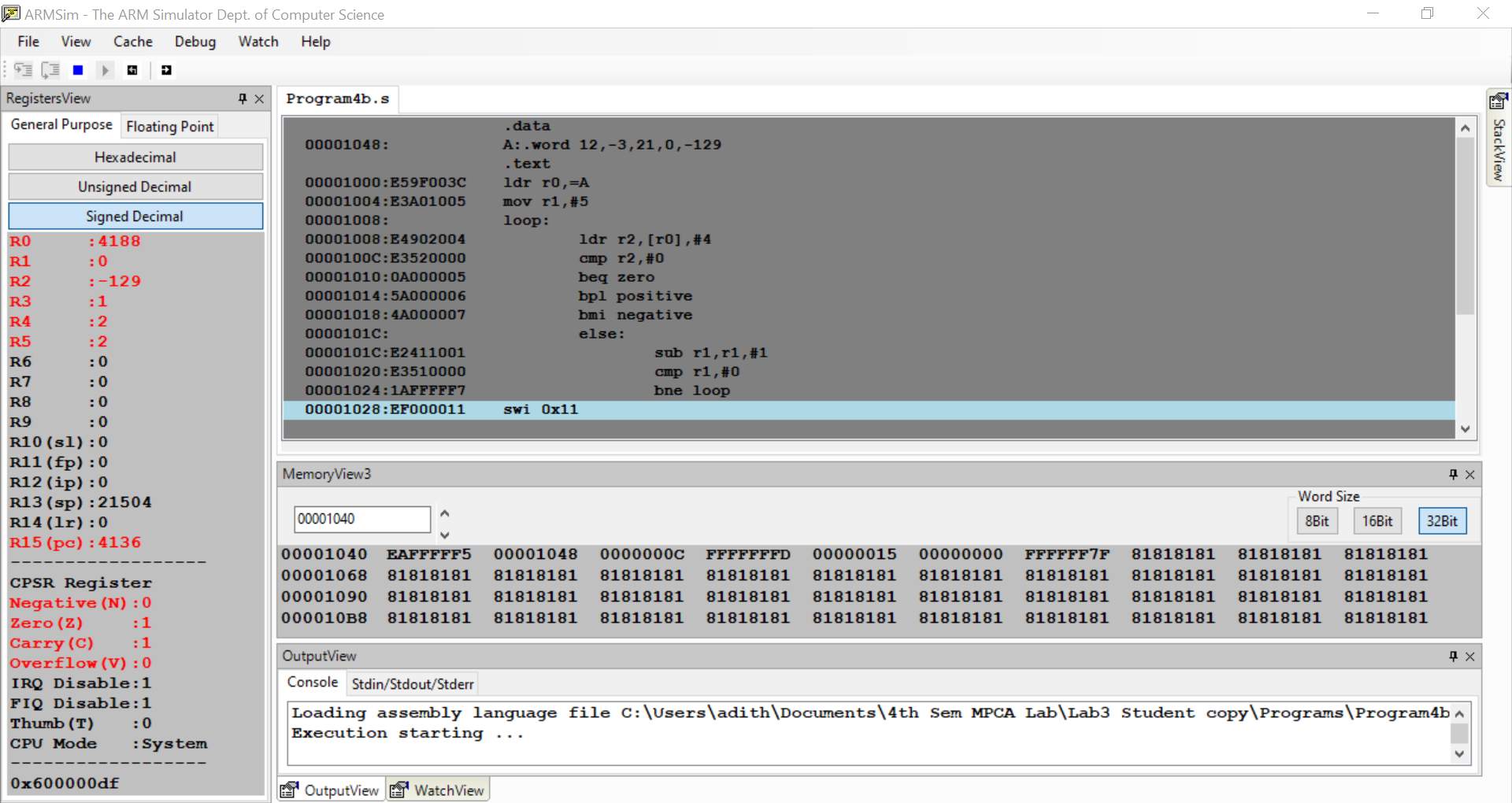
add r4,r4,#1

b else

negative:

add r5,r5,#1

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

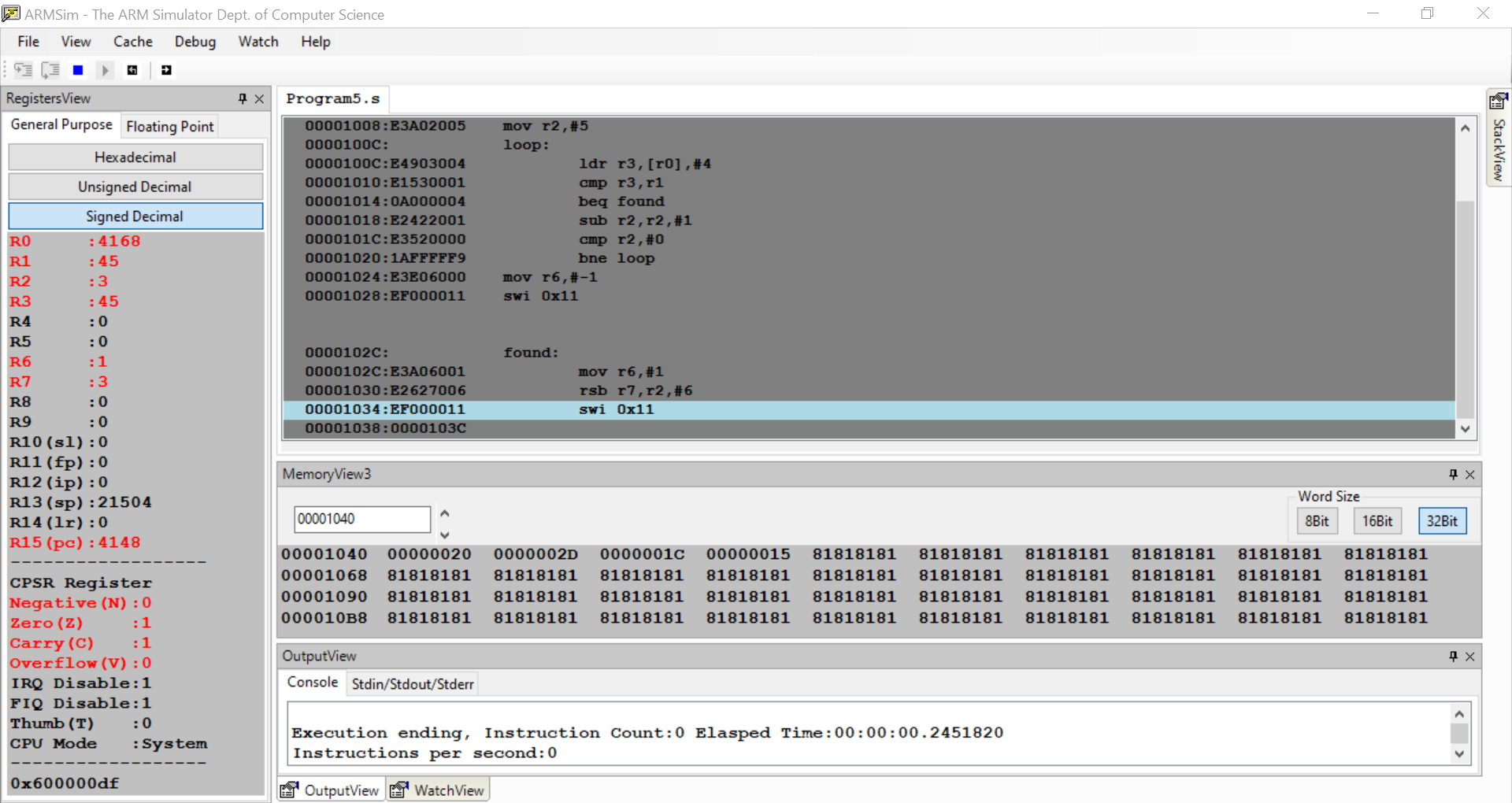
swi 0x11

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

