Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date: 31/1/21

Name: Adithya M S	SRN: PES1UG19CS027	Section: A

Week#____2_____ Program Number: ___1___

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.

I.

```
mov r0,#5
cmp r0,#0
beq zero
bpl positive
mov r1,#3
swi 0x11
```

zero:

mov r1,#1

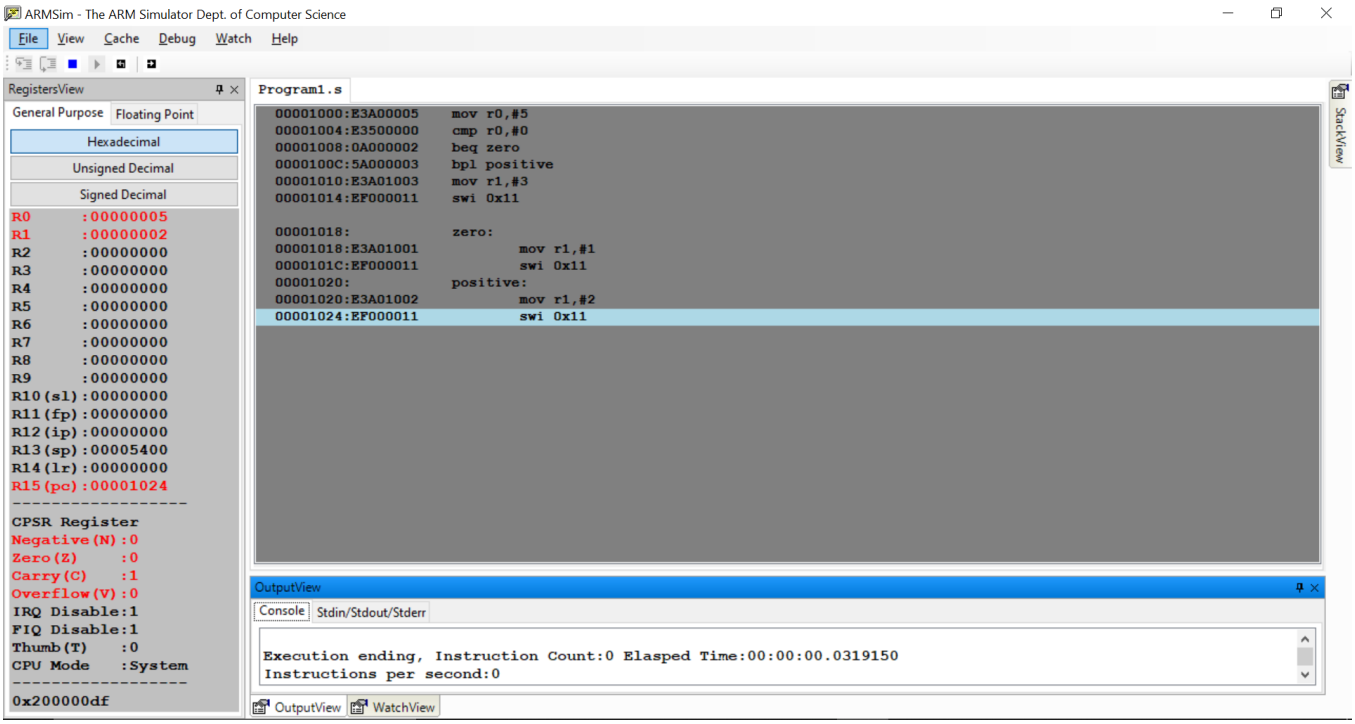
swi 0x11

positive:

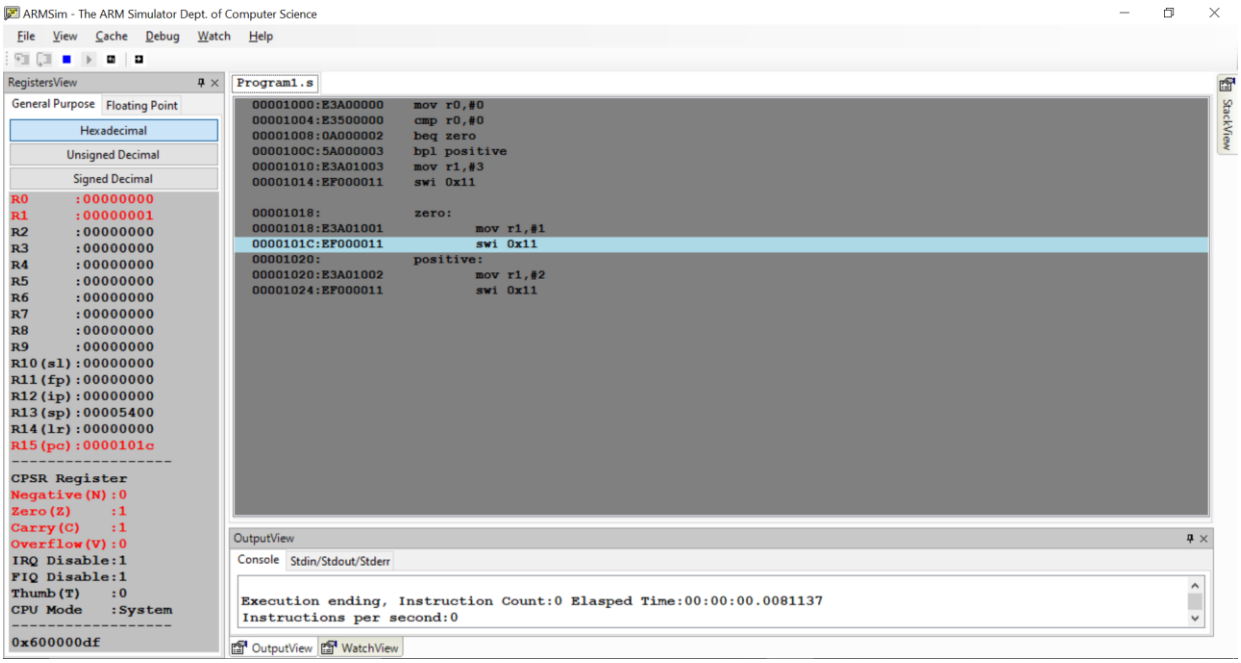
mov r1,#2

swi 0x11

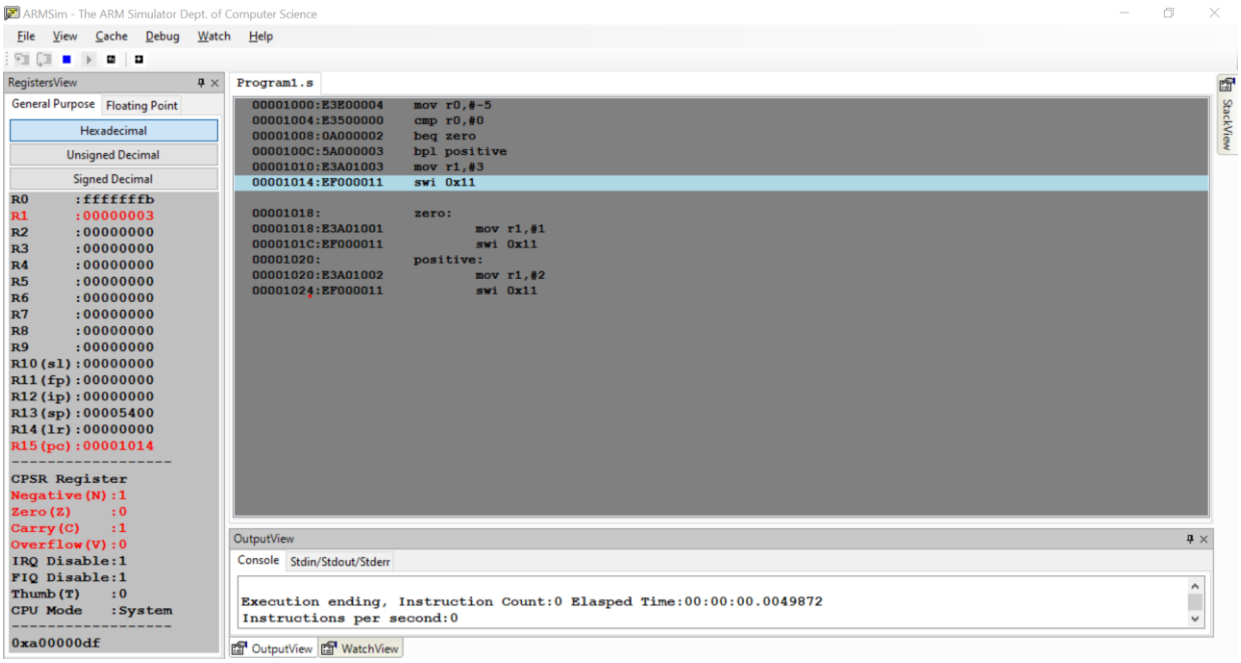
Example 1:



Example 2:



Example 3:



Week#____2_____ Program Number: ___2___

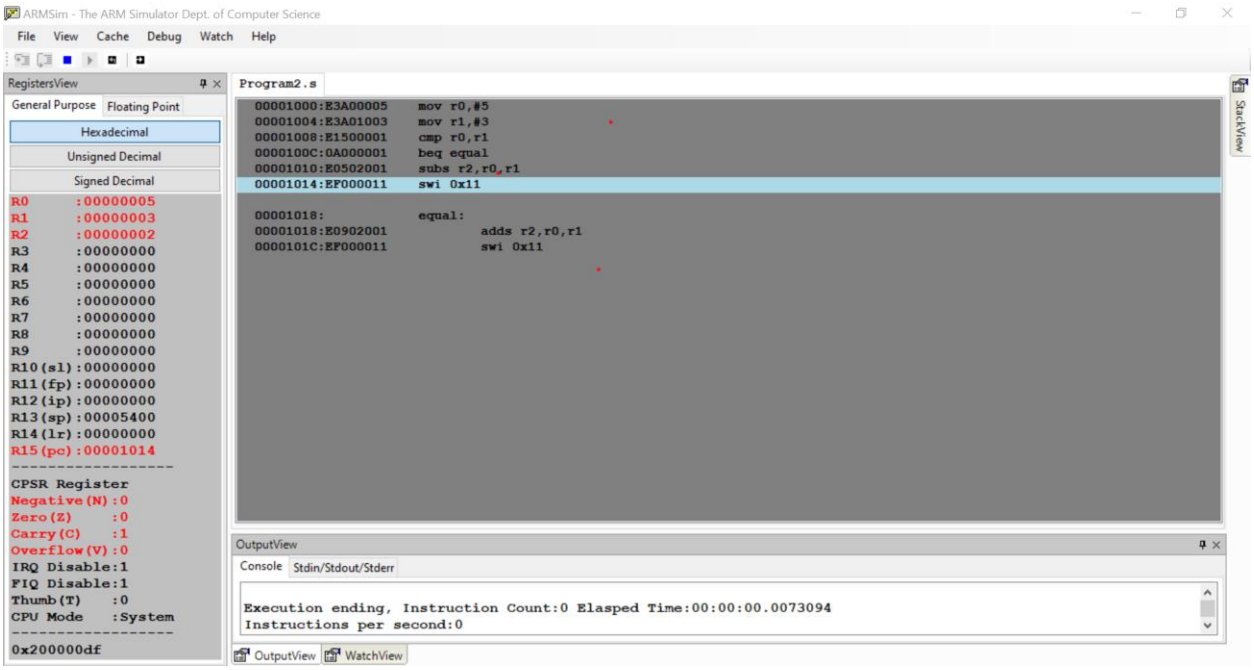
Write an ALP to compare the value of R0 and R1, add if R0 = R1, else subtract

I.

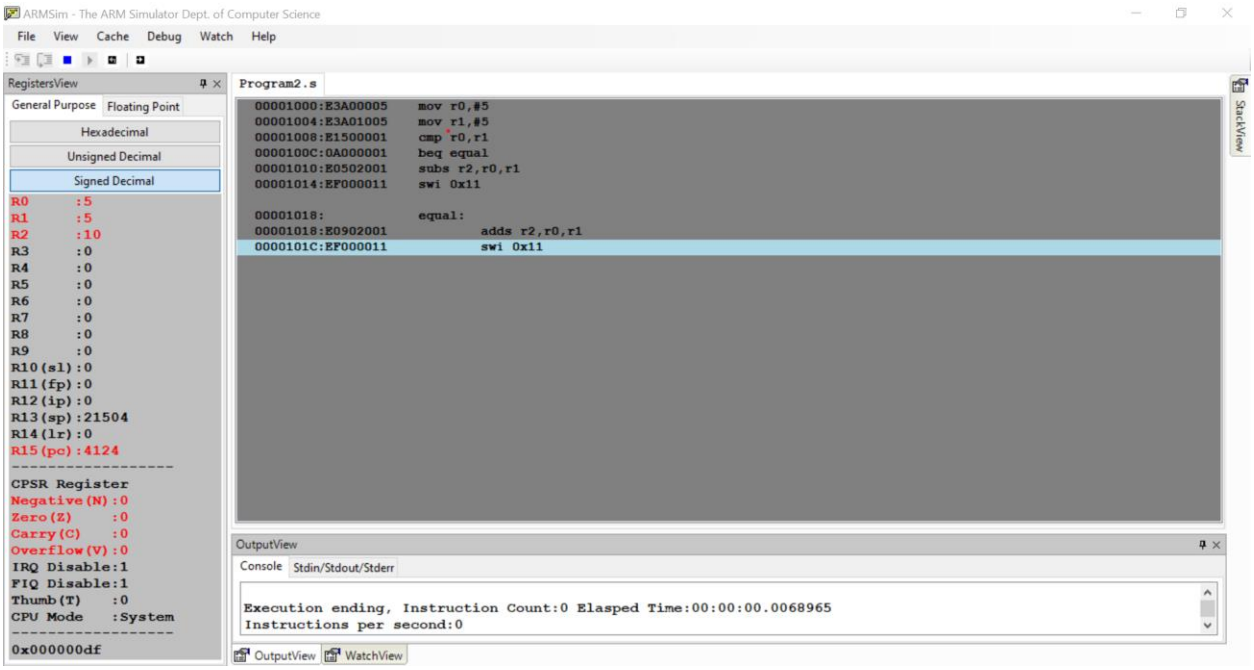
```
mov r0,#5
mov r1,#3
cmp r0,r1
beq equal
subs r2,r0,r1
swi 0x11

equal:
    adds r2,r0,r1
    swi 0x11
```

Example 1:



Example 2:

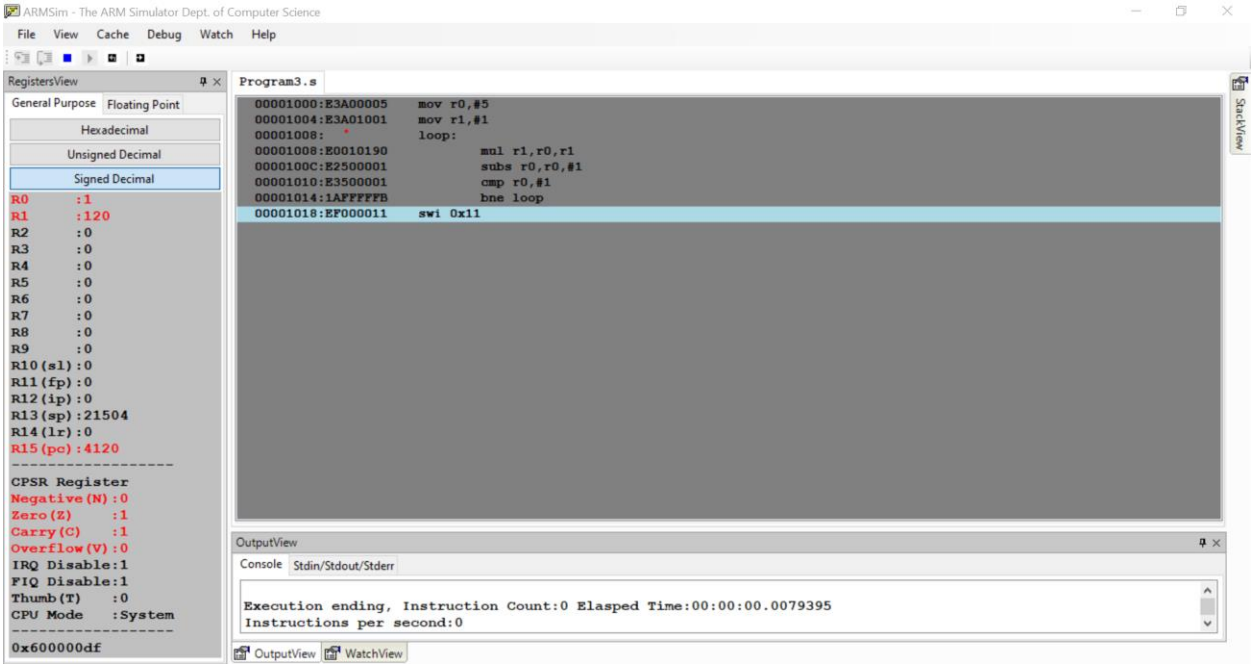


Week#____2_____ Program Number: ___3___

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.

I.

```
mov r0,#5
mov r1,#1
loop:
    mul r1,r0,r1
    subs r0,r0,#1
    cmp r0,#1
    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 :1
R1 :120
R2 :0
R3 :0
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):4120
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
00001000:E3A00005 mov r0,#5
00001004:E3A01001 mov r1,#1
00001008: loop:
00001008:E0010190 mul r1,r0,r1
0000100C:E2500001 subs r0,r0,#1
00001010:E3500001 cmp r0,#1
00001014:1AFFFFFB bne loop
00001018:EF000011 swi 0x11
StackView
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0079395
Instructions per second:0
OutputView WatchView

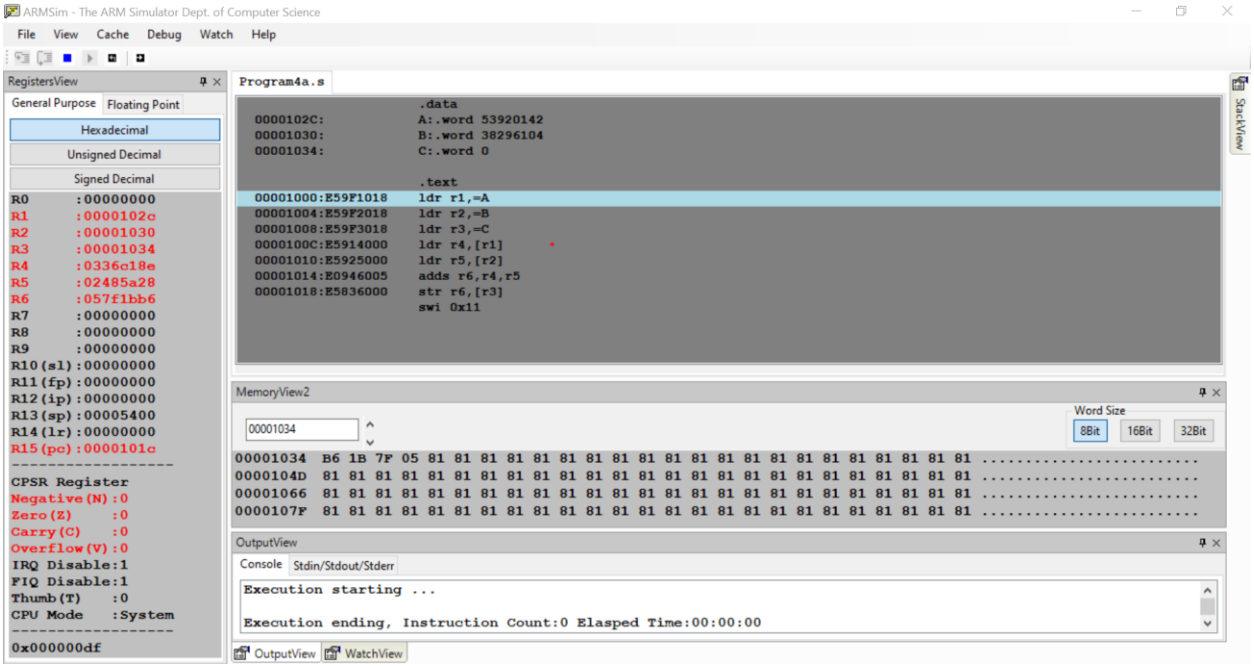
Week#____2_____ Program Number: ___4a___

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

I.

```
.data
A:.word 53920142
B:.word 38296104
C:.word 0

.text
ldr r1,=A
ldr r2,=B
ldr r3,=C
ldr r4,[r1]
ldr r5,[r2]
adds r6,r4,r5
str r6,[r3]
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 :00000000
R1 :0000102c
R2 :00001030
R3 :00001034
R4 :0336a18e
R5 :02485a28
R6 :057f1bb6
R7 :00000000
R8 :00000000
R9 :00000000
R10(sl):00000000
R11(fp):00000000
R12(ip):00000000
R13(sp):00005400
R14(lr):00000000
R15(pc):0000101c
CPSR Register
Negative(N):0
Zero(Z) :0
Carry(C) :0
Overflow(V):0
IRQ Disable:1
FIQ Disable:1
Thumb(T) :0
CPU Mode :System
0x000000df
Program4a.s
.data
0000102C: A:.word 53920142
00001030: B:.word 38296104
00001034: C:.word 0
.text
00001000:E59F1018 ldr r1,=A
00001004:E59F2018 ldr r2,=B
00001008:E59F3018 ldr r3,=C
0000100C:E5914000 ldr r4,[r1]
00001010:E5925000 ldr r5,[r2]
00001014:E0946005 adds r6,r4,r5
00001018:E5836000 str r6,[r3]
swi 0x11
StackView
MemoryView2
00001034
Word Size
8Bit 16Bit 32Bit
00001034 B6 1B 7F 05 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
0000104D 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
00001066 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
0000107F 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
OutputView
Console Stdin/Stdout/Stderr
Execution starting ...
Execution ending, Instruction Count:0 Elasped Time:00:00:00
OutputView WatchView

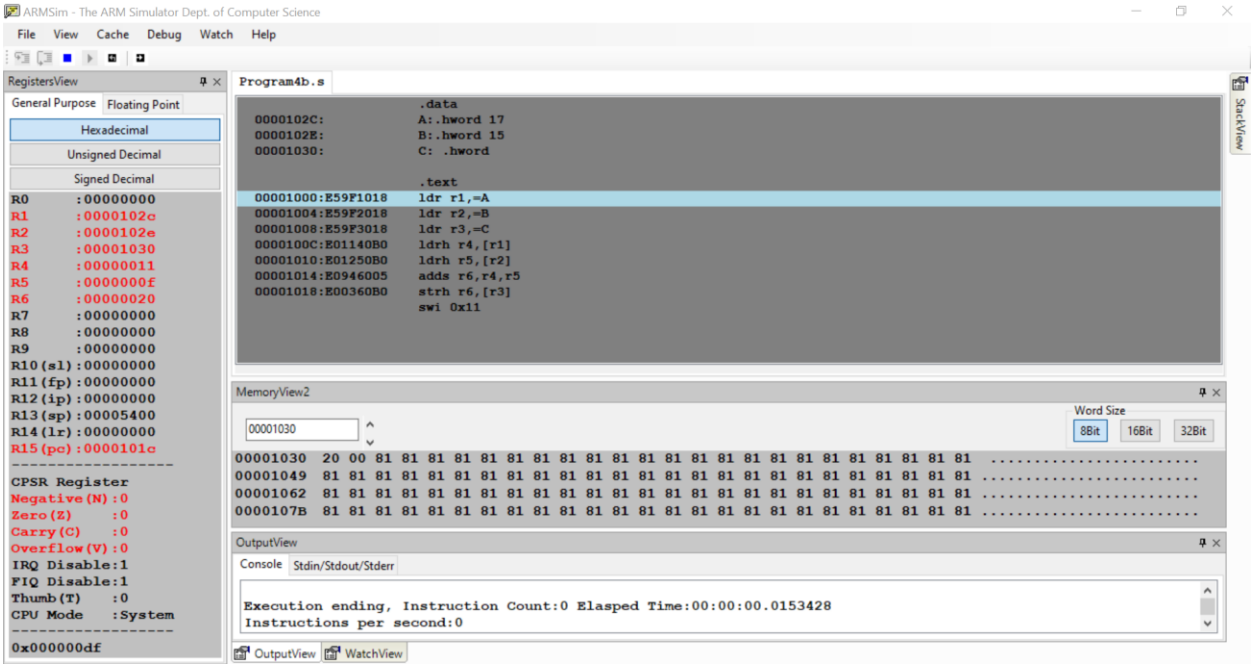
Week#____2_____ Program Number: ___4b___

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

I.

```
.data
A:.hword 17
B:.hword 15
C: .hword

.text
ldr r1,=A
ldr r2,=B
ldr r3,=C
ldrh r4,[r1]
ldrh r5,[r2]
adds r6,r4,r5
strh r6,[r3]
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 :00000000
R1 :0000102c
R2 :0000102e
R3 :00001030
R4 :00000011
R5 :0000000f
R6 :00000020
R7 :00000000
R8 :00000000
R9 :00000000
R10(sl):00000000
R11(fp):00000000
R12(ip):00000000
R13(sp):00005400
R14(lr):00000000
R15(pc):0000101c
CPSR Register
Negative(N):0
Zero(Z) :0
Carry(C) :0
Overflow(V):0
IRQ Disable:1
FIQ Disable:1
Thumb(T) :0
CPU Mode :System
0x000000df
Program4b.s
.data
0000102C: A:.hword 17
0000102E: B:.hword 15
00001030: C: .hword
.text
00001000:E59F1018 ldr r1,=A
00001004:E59F2018 ldr r2,=B
00001008:E59F3018 ldr r3,=C
0000100C:E01140B0 ldrh r4,[r1]
00001010:E01250B0 ldrh r5,[r2]
00001014:E0946005 adds r6,r4,r5
00001018:E00360B0 strh r6,[r3]
swi 0x11
StackView
MemoryView2
00001030
Word Size
8Bit 16Bit 32Bit
00001030 20 00 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
00001049 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
00001062 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
0000107B 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0153428
Instructions per second:0
OutputView WatchView

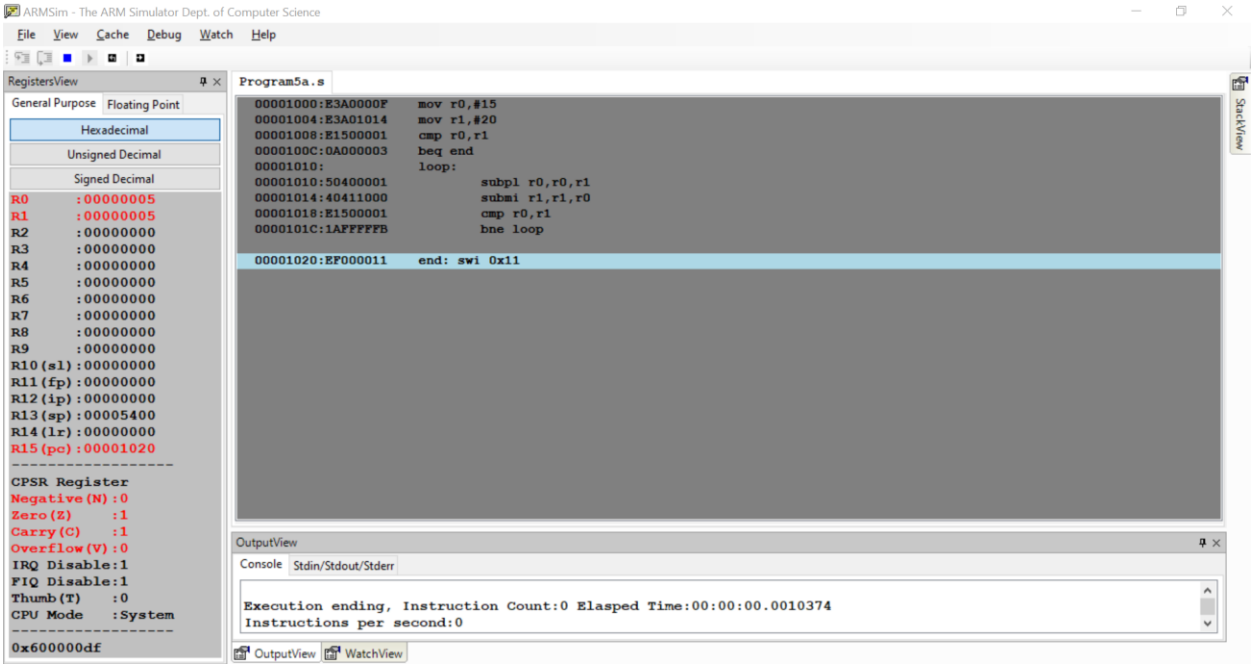
Week#____2_____ Program Number: ___5a___

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

I.

```
mov r0,#15
mov r1,#20
cmp r0,r1
beq end
loop:
    subpl r0,r0,r1
    submi r1,r1,r0
    cmp r0,r1
    bne loop

end: 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 :00000005
R1 :00000005
R2 :00000000
R3 :00000000
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):00001020
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
Program5a.s
00001000:E3A0000F mov r0,#15
00001004:E3A01014 mov r1,#20
00001008:E1500001 cmp r0,r1
0000100C:0A000003 beq end
00001010: loop:
00001010:50400001 subpl r0,r0,r1
00001014:40411000 submi r1,r1,r0
00001018:E1500001 cmp r0,r1
0000101C:1AFFFFFB bne loop
00001020:EF000011 end: swi 0x11
StackView
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0010374
Instructions per second:0
OutputView WatchView

Week#____2_____ Program Number: ___5b___

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

I.

```
.data
A:.word 15
B:.word 20
C:.word

.text
ldr r0,=A
ldr r1,=B
ldr r2,=C
ldr r3,[r0]
ldr r4,[r1]
cmp r3,r4
beq end
loop:
```

subpl r3,r3,r4

submi r4,r4,r3

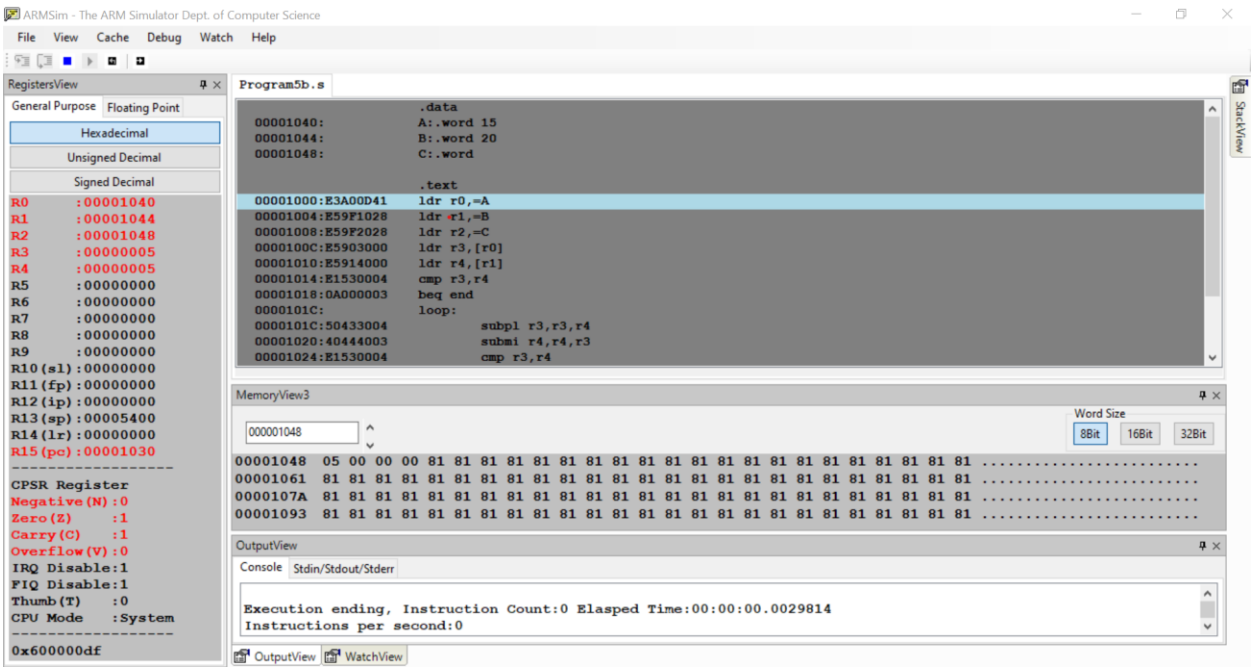
cmp r3,r4

bne loop

end:

str r3,[r2]

swi 0x11



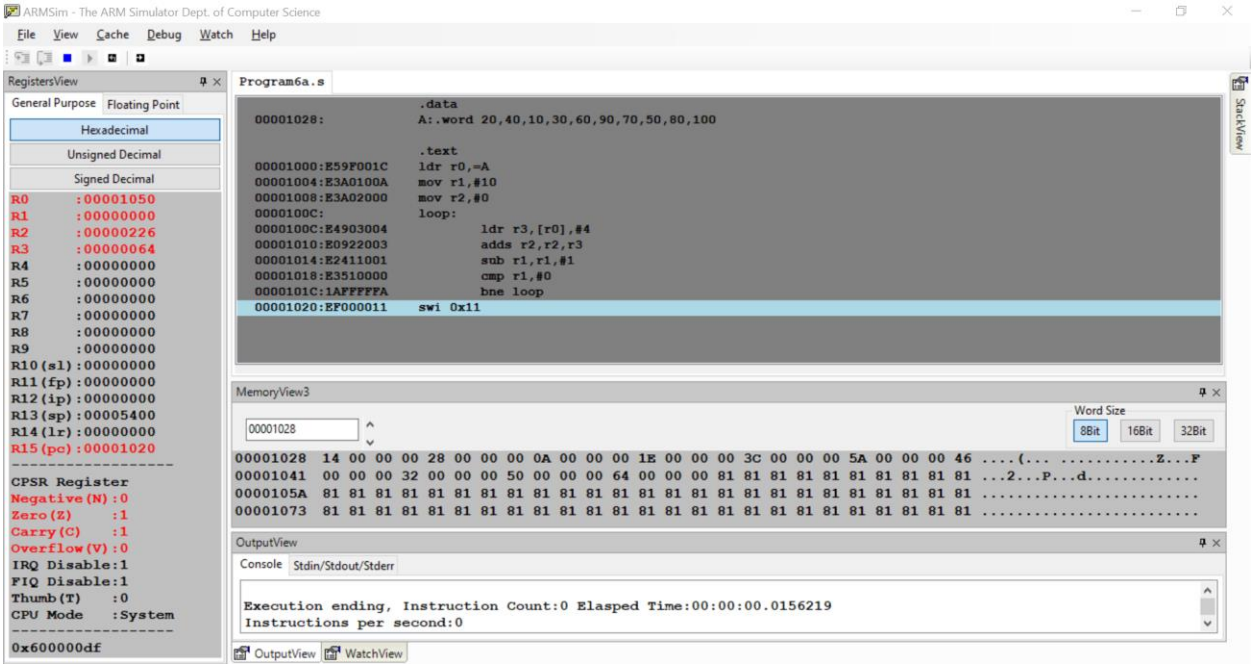
Week#____2_____ Program Number: ___6a___

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

I.

```
.data
A:.word 20,40,10,30,60,90,70,50,80,100

.text
ldr r0,=A
mov r1,#10
mov r2,#0
loop:
    ldr r3,[r0],#4
    adds r2,r2,r3
    sub r1,r1,#1
    cmp r1,#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 :00001050
R1 :00000000
R2 :00000226
R3 :00000064
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):00001020
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
Program6a.s
.data
00001028: A:.word 20,40,10,30,60,90,70,50,80,100
.text
00001000:E59F001C ldr r0,=A
00001004:E3A0100A mov r1,#10
00001008:E3A02000 mov r2,#0
0000100C: loop:
0000100C:E4903004 ldr r3,[r0],#4
00001010:E0922003 adds r2,r2,r3
00001014:E2411001 sub r1,r1,#1
00001018:E3510000 cmp r1,#0
0000101C:1AFFFFFA bne loop
00001020:EF000011 swi 0x11
StackView
MemoryView3
00001028
Word Size
8Bit 16Bit 32Bit
00001028 14 00 00 00 28 00 00 00 0A 00 00 00 1E 00 00 00 3C 00 00 00 5A 00 00 00 46 ....(... ...<...Z...F
00001041 00 00 00 32 00 00 00 50 00 00 00 64 00 00 00 81 81 81 81 81 81 81 81 81 ...2...P...d.....
0000105A 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
00001073 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0156219
Instructions per second:0
OutputView WatchView

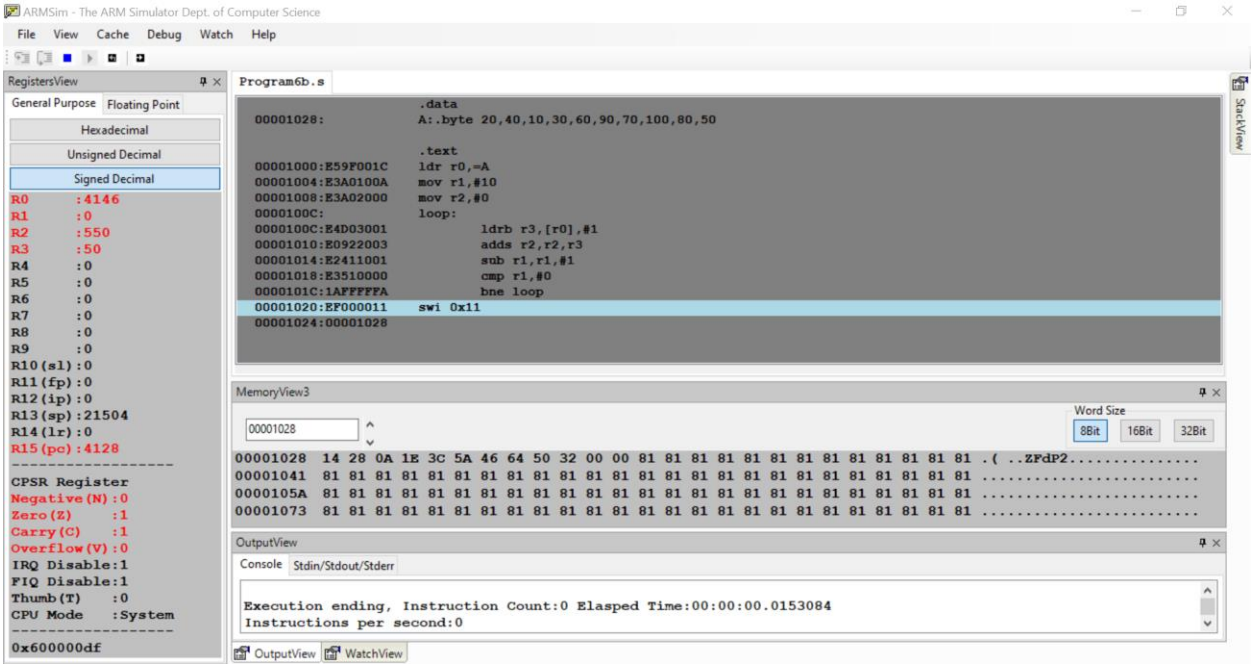
Week#____2_____ Program Number: ___6b___

Write an ALP to add array of ten 8-bit numbers taking data from memory location stored as byte data

I.

```
.data
A:.byte 20,40,10,30,60,90,70,100,80,50

.text
ldr r0,=A
mov r1,#10
mov r2,#0
loop:
    ldrb r3,[r0],#1
    adds r2,r2,r3
    sub r1,r1,#1
    cmp r1,#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 :4146
R1 :0
R2 :550
R3 :50
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):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
Program6b.s
.data
00001028: A:.byte 20,40,10,30,60,90,70,100,80,50
.text
00001000:E59F001C ldr r0,=A
00001004:E3A0100A mov r1,#10
00001008:E3A02000 mov r2,#0
0000100C: loop:
0000100C:E4D03001 ldrb r3,[r0],#1
00001010:E0922003 adds r2,r2,r3
00001014:E2411001 sub r1,r1,#1
00001018:E3510000 cmp r1,#0
0000101C:1AFFFFFA bne loop
00001020:EF000011 swi 0x11
00001024:00001028
StackView
MemoryView3
00001028
Word Size
8Bit 16Bit 32Bit
00001028 14 28 0A 1E 3C 5A 46 64 50 32 00 00 81 81 81 81 81 81 81 81 81 81 81 81 .(..ZFdP2.....
00001041 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
0000105A 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
00001073 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 .....
OutputView
Console Stdin/Stdout/Stderr
Execution ending, Instruction Count:0 Elasped Time:00:00:00.0153084
Instructions per second:0
OutputView WatchView

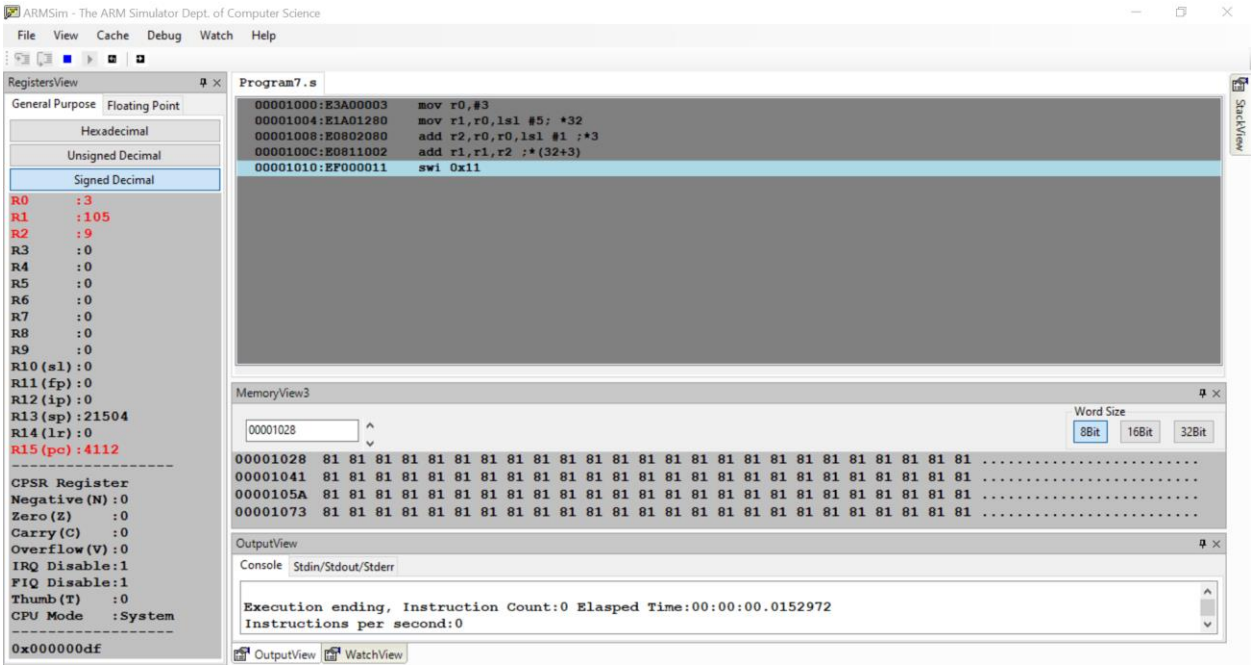
Week#____2_____ Program Number: ___7___

Write an ALP to multiply using barrel shifter.

35*R0

I.

```
mov r0,#3
mov r1,r0,lsl #5; *32
add r2,r0,r0,lsl #1 ;*3
add r1,r1,r2 ;*(32+3)
swi 0x11
```



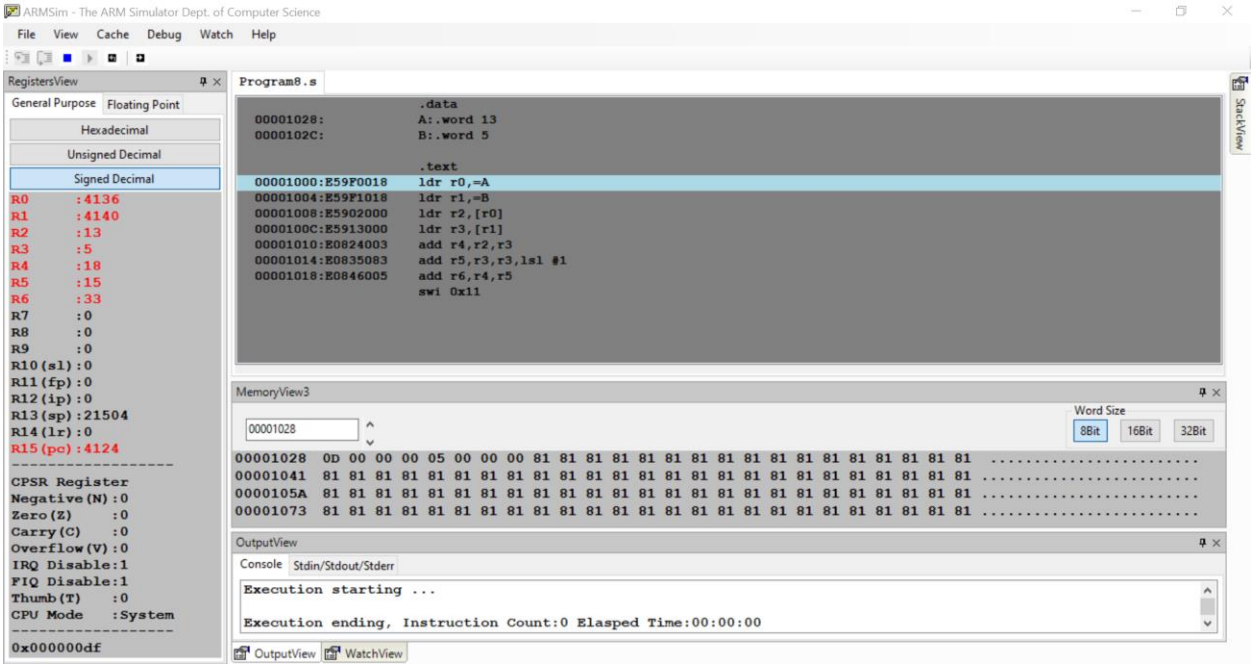
Week#____2_____ Program Number: ___8___

Write an ALP to evaluate the expression (A+B) + (3*B), where A and B are memory location.

I.

```
.data
A:.word 13
B:.word 5

.text
ldr r0,=A
ldr r1,=B
ldr r2,[r0]
ldr r3,[r1]
add r4,r2,r3
add r5,r3,r3,lsl #1
add r6,r4,r5
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 :4140
R2 :13
R3 :5
R4 :18
R5 :15
R6 :33
R7 :0
R8 :0
R9 :0
R10(sl):0
R11(fp):0
R12(ip):0
R13(sp):21504
R14(lr):0
R15(pc):4124
CPSR Register
Negative(N):0
Zero(Z) :0
Carry(C) :0
Overflow(V):0
IRQ Disable:1
FIQ Disable:1
Thumb(T) :0
CPU Mode :System
0x000000df
Program8.s
.data
00001028: A:.word 13
0000102C: B:.word 5
.text
00001000:E59F0018 ldr r0,=A
00001004:E59F1018 ldr r1,=B
00001008:E5902000 ldr r2,[r0]
0000100C:E5913000 ldr r3,[r1]
00001010:E0824003 add r4,r2,r3
00001014:E0835083 add r5,r3,r3,lsl #1
00001018:E0846005 add r6,r4,r5
swi 0x11
StackView
MemoryView3
00001028
Word Size
8Bit 16Bit 32Bit
00001028 0D 00 00 00 05 00 00 00 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81
00001041 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81
0000105A 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81
00001073 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81 81
OutputView
Console Stdin/Stdout/Stderr
Execution starting ...
Execution ending, Instruction Count:0 Elasped Time:00:00:00
OutputView WatchView