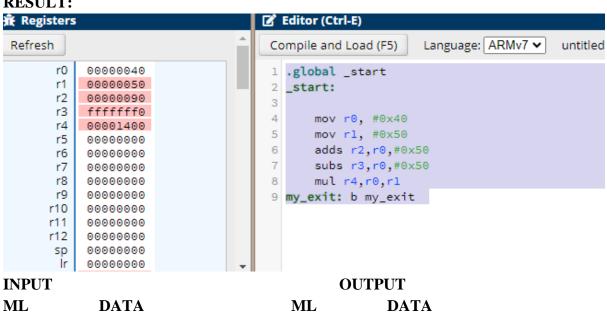
Lab-4

Obj-1: Perform Addition and Subtraction of two 32-bit numbers using data processing addressing mode (with immediate data).

```
Program:
.global _start
_start:
      mov r0, #0x40
      mov r1, #0x50
       adds r2,r0,#0x50
       subs r3,r0,#0x50
       mul r4,r0,r1
my_exit: b my_exit
```

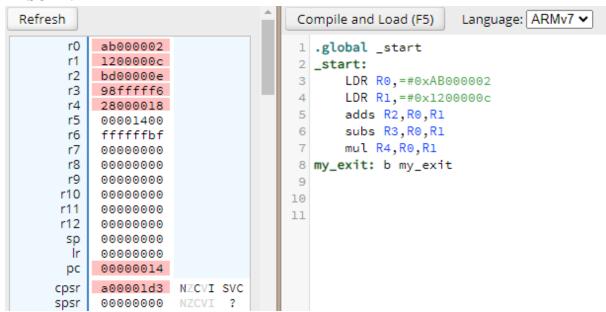
RESULT:



0X40 (R0) 0X90 (R2) 0Xfffffff0 (R3) 0X50 (R1) 0X1400 (R4)

```
Objective-1
.global _start
_start:
    LDR R0,=#0xAB000002
    LDR R1,=#0x1200000c
    adds R2,R0,R1
    subs R3,R0,R1
    mul R4,R0,R1
my_exit: b my_exit
```

RESULT:



INPUT		OUTPUT	
ML	DATA	\mathbf{ML}	DATA
-	0XAB000002	-	0Xbd00000e
-	0X1200000C	-	0X98fffff6
		-	0X28000018

Objective 2: Perform Addition, Subtraction, and Multiplication of two 32-bit numbers using load/store addressing mode.

```
Program:
```

 $. global _start$

_start:

LDR R0,=0X10100000

LDR R1,[R0],#4

LDR R2,[R0],#4

ADDS R3,R1,R2

STR R3,[R0],#4

SUBS R4,R1,R2

STR R4,[R0],#4

MUL R5,R1,R2

STR R5,[R0]

my_exit: b my_exit

RESULT

Refresh				Go to addre	ss, label, or re	gister: 1010	0000	
r0	10100010			Address	Memory co	ntents and A	SCII	
r1	00000040			10011100	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r2	00000050			100fff90	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r3	00000090			100fffa0	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r4	fffffff0			100fffb0	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r5	00001400			100fffc0	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r6	ffffffbf			100fffd0	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r7	00000000			100fffe0	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r8	00000000			100ffff0	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
r9 r10	00000000			10100000	00000040	00000050	00000090	fffffff0
r11	00000000			10100010	00001400	00000000	aaaaaaaa	aaaaaaaa
r12	00000000			10100020	aaaaaaaa	aaaaaaaa	aaaaaaaa	88888888
sp	00000000			10100030	aaaaaaaa	88888888	aaaaaaaa	88888888
lr lr	00000000			10100030				
pc	00000024			10100040	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
cpsr	800001d3	NZCVI SVC			aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
spsr	00000000	NZCVI ?		10100060	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
5051	0000000		_	10100070	aaaaaaaa	aaaaaaaa	aaaaaaaa	aaaaaaaa
INPUT OUTPUT								
ML DATA			\mathbf{ML}		DATA			
0X10100000 0X40			0X10100008		0X90			
0X10100004 0X50			0X1010000C		0Xfffffff0			
					0X1010	00010	0X10	

Objective-3: Perform the logical operations (AND, OR, XOR, and NOT) on two 32-bit numbers using load/store addressing mode

Program

 $. global _start$

_start:

LDR R0,=0X10100000

LDR R1,[R0],#4

LDR R2,[R0],#4

ANDS R3,R2,R1

STR R3,[R0],#4

ORR R4,R2,R1

STR R4,[R0],#4

EOR R5,R2,R1

STR R5,[R0],#4

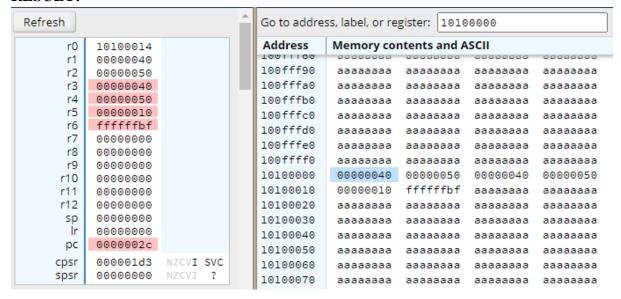
MVN R6, R1

STR R6,[R0]

my_exit: b my_exit

RESULT:

INPLIT



OUTPUT

11 11 0 1		001101	
ML	DATA	\mathbf{ML}	DATA
0X10100000	0X40	0X10100008	0X40
0X10100004	0X50	0X1010000C	0X50
		0X10100010	0X10
		0X10100014	0Xffffffbf