Lab 1 - Getting Started with RISC-V (Assembly Language) in VS Code

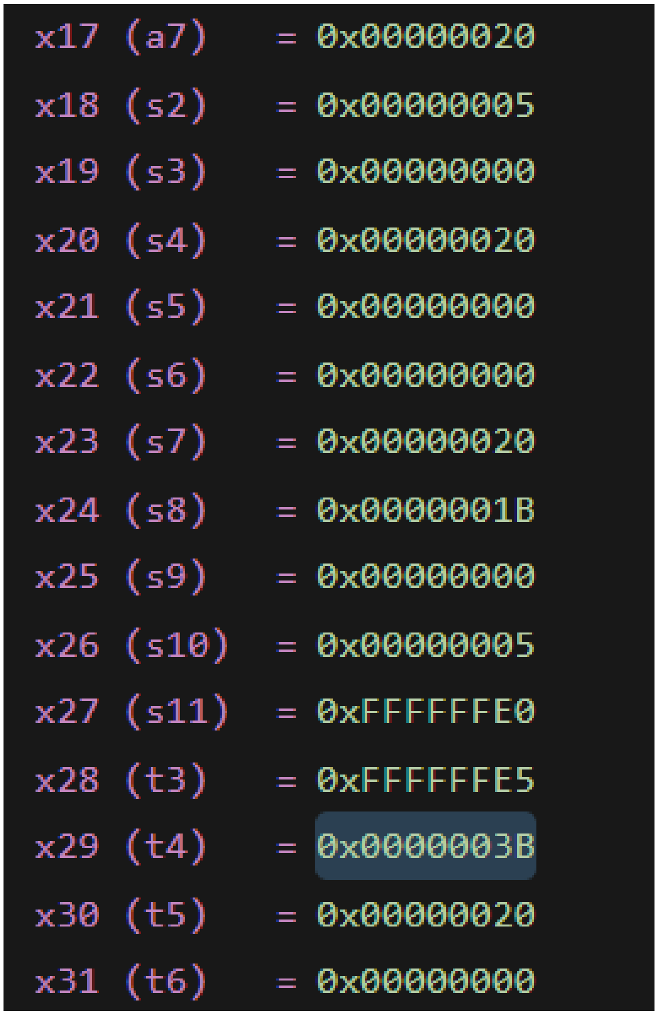
Name: Aqsa Muneer, Maleeha Khan	Student ID:am10527, mk09991

Task 3:

Code:

```
li x20, 5 #a=5

li x18, 5 #just to store 5

li x19, 0

li x17, 32

add x21, x19, x19 #b = 0+0

add x20, x21, x17 #a=b+32

add x23, x20, x21 #newvar1=a+b

sub x24, x23, x18 #newvar2=newvar1-5 (=d)


sub x26, x20, x24 #newvar3=a-d

sub x27, x21, x20 #newvar4=b-a

add x28, x27, x26 #newvar5= newvar3+newvar4

add x29, x28, x24 # added d to newvar5

add x30, x20, x21 #nv=a+b

add x29, x29, x24 #nv2=d+e

add x29, x29, x30
```

Results:

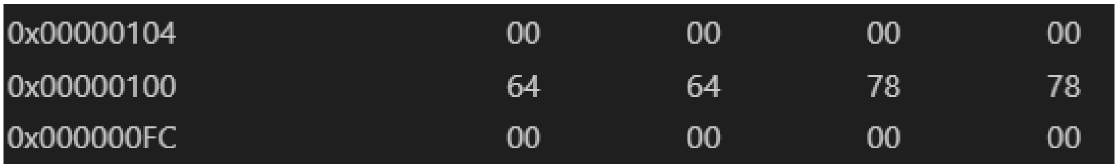


```
x17 (a7)   = 0x00000020
x18 (s2)   = 0x00000005
x19 (s3)   = 0x00000000
x20 (s4)   = 0x00000020
x21 (s5)   = 0x00000000
x22 (s6)   = 0x00000000
x23 (s7)   = 0x00000020
x24 (s8)   = 0x0000001B
x25 (s9)   = 0x00000000
x26 (s10)  = 0x00000005
x27 (s11)  = 0xFFFFFFE0
x28 (t3)   = 0xFFFFFFE5
x29 (t4)   = 0x0000003B
x30 (t5)   = 0x00000020
x31 (t6)   = 0x00000000
```

Task 4a:

1. Store x10 as unsigned integer at address 0x100.

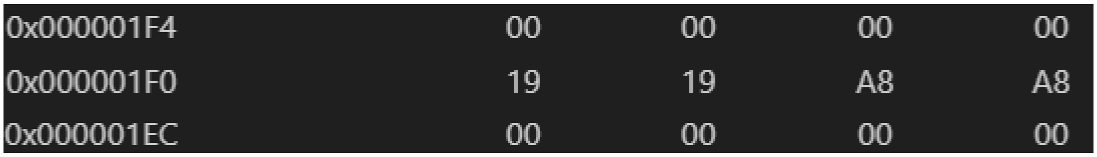
```
li x10, 0x78786464

li x11,  0xA8A81919

sw x10, 0x100(x0)

sw x11, 0x1F0 (x0)
```



```
0x00000104    00    00    00    00
0x00000100    64    64    78    78
0x000000FC    00    00    00    00
```

2. Store x11 as unsigned integer at address 0x1F0.

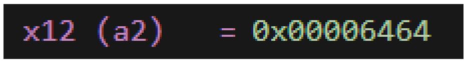
```
li x10, 0x78786464

li x11,  0xA8A81919

sw x10, 0x100(x0)

sw x11, 0x1F0 (x0)
```



```
0x000001F4    00    00    00    00
0x000001F0    19    19    A8    A8
0x000001EC    00    00    00    00
```

3. Load an unsigned short integer (two bytes) from address 0x100 in x12.

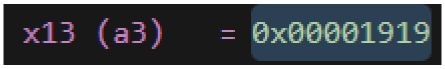
```
li x5, 0x100 #temp reg

lhu x12, 0(x5)
```



```
x12 (a2)   = 0x00006464
```

4. Load a short integer from address 0x1F0 in register x13.

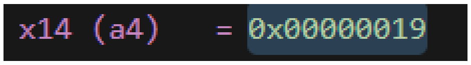
```
li x6, 0x1F0

lh x13, 0(x6)
```



```
x13 (a3)    = 0x00001919
```

5. Load a singed character from address 0x1F0 in register x14.

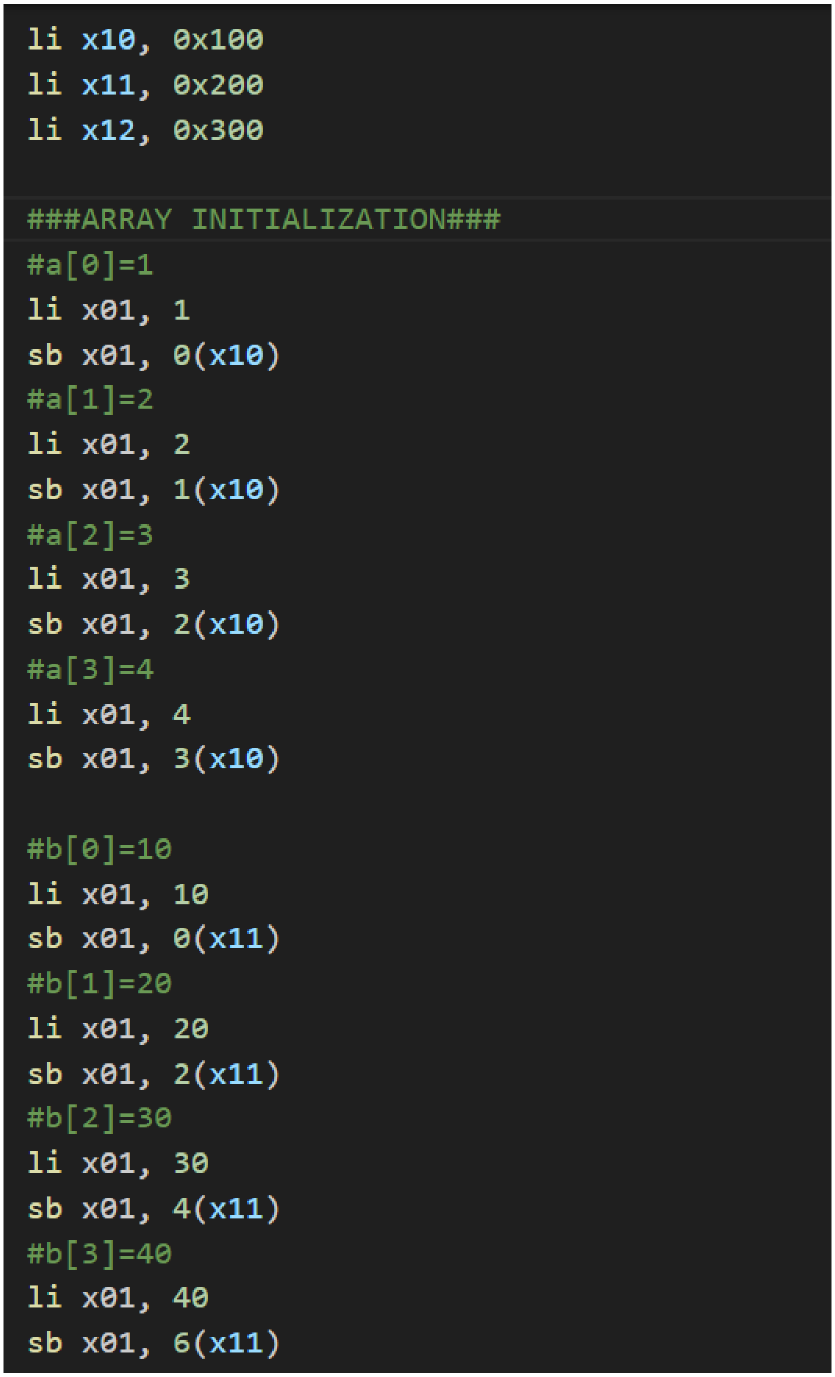
```
li x7, 0x1F0

lb x14, 0(x7)
```



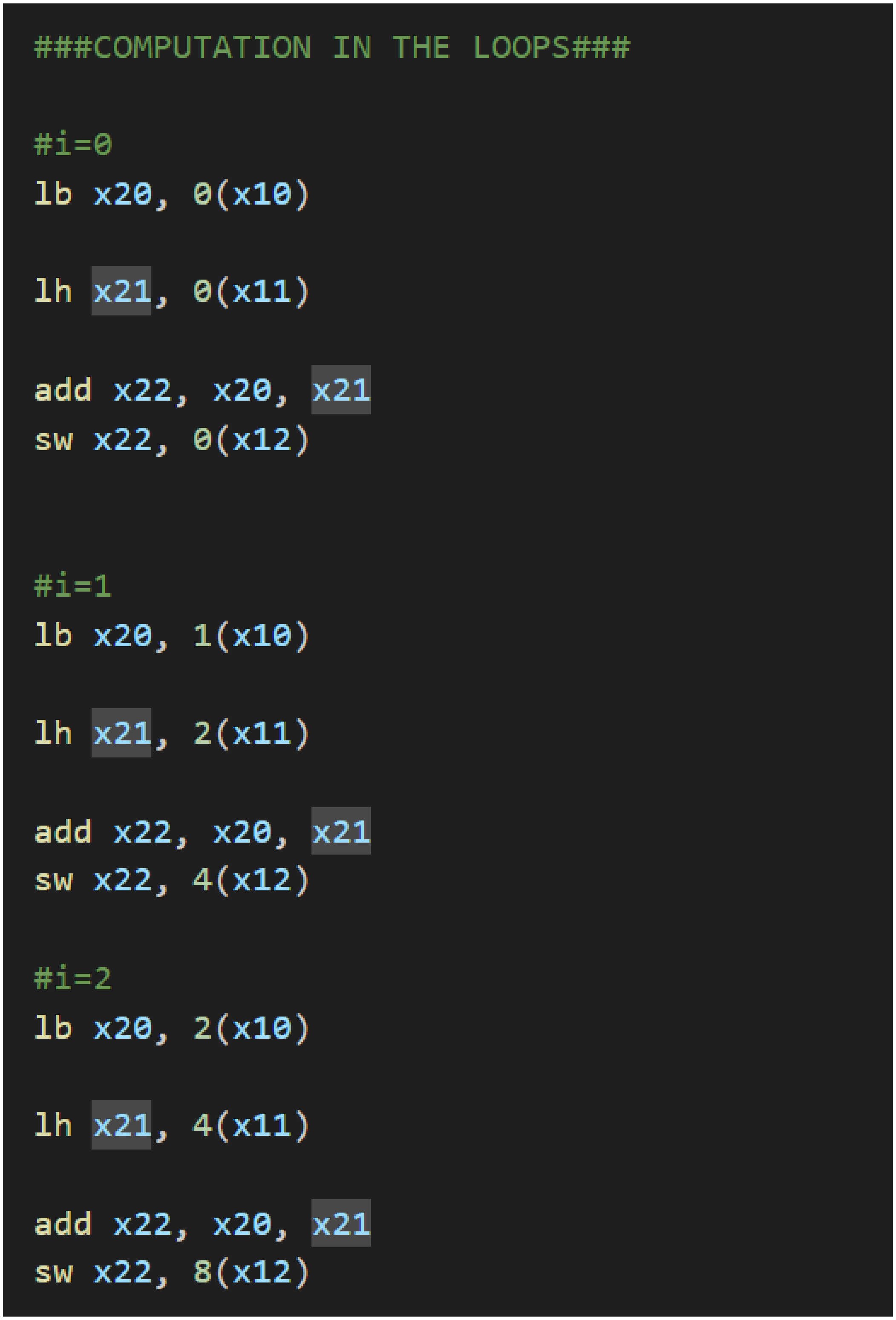
```
x14 (a4)    = 0x00000019
```

Task 4b:

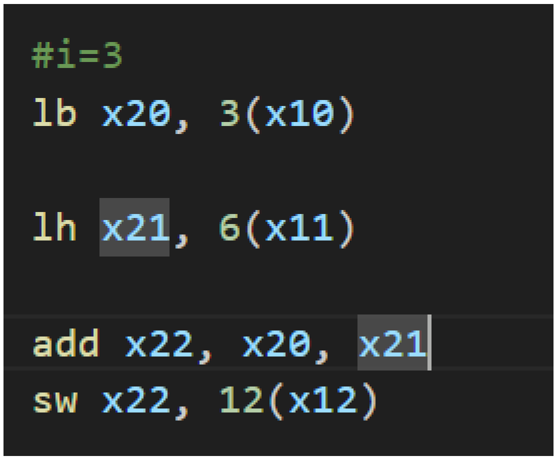
Code:



```
li x10, 0x100
li x11, 0x200
li x12, 0x300

###ARRAY INITIALIZATION###
#a[0]=1
li x01, 1
sb x01, 0(x10)
#a[1]=2
li x01, 2
sb x01, 1(x10)
#a[2]=3
li x01, 3
sb x01, 2(x10)
#a[3]=4
li x01, 4
sb x01, 3(x10)

#b[0]=10
li x01, 10
sb x01, 0(x11)
#b[1]=20
li x01, 20
sb x01, 2(x11)
#b[2]=30
li x01, 30
sb x01, 4(x11)
#b[3]=40
li x01, 40
sb x01, 6(x11)
```

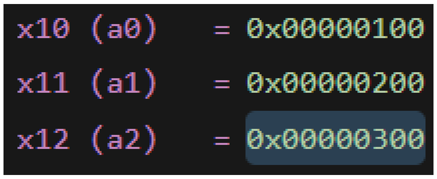


```
###COMPUTATION IN THE LOOPS###

#i=0
lb x20, 0(x10)

lh x21, 0(x11)

add x22, x20, x21
sw x22, 0(x12)



#i=1
lb x20, 1(x10)

lh x21, 2(x11)

add x22, x20, x21
sw x22, 4(x12)

#i=2
lb x20, 2(x10)

lh x21, 4(x11)

add x22, x20, x21
sw x22, 8(x12)
```

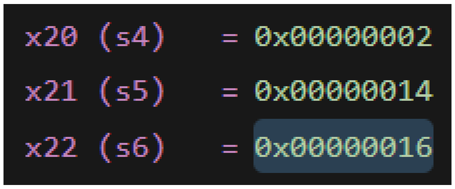


```
#i=3
lb x20, 3(x10)

lh x21, 6(x11)

add x22, x20, x21
sw x22, 12(x12)
```

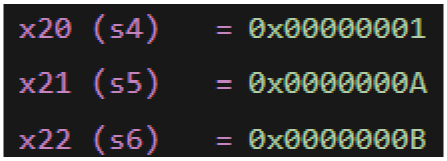
Results:



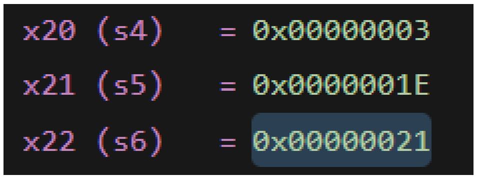
```
x10 (a0)    = 0x00000100
x11 (a1)    = 0x00000200
x12 (a2)    = 0x00000300
```



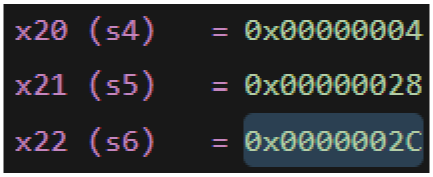
```
x20 (s4)    = 0x00000002
x21 (s5)    = 0x00000014
x22 (s6)    = 0x00000016
```



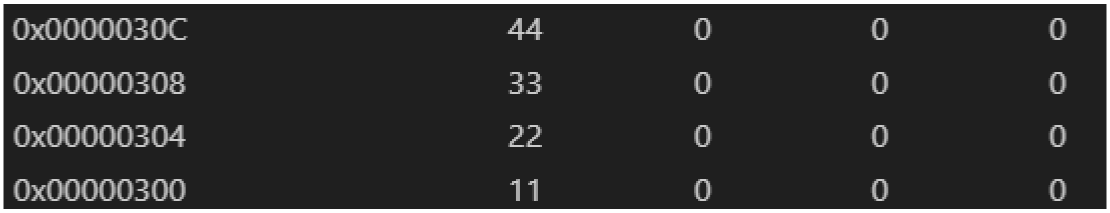
```
x20 (s4)    = 0x00000001
x21 (s5)    = 0x0000000A
x22 (s6)    = 0x0000000B
```



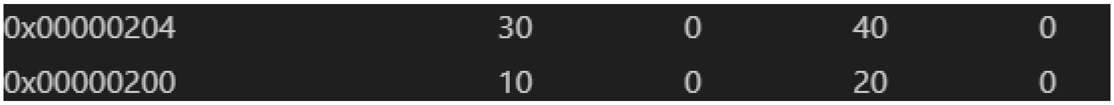
```
x20 (s4)    = 0x00000003
x21 (s5)    = 0x0000001E
x22 (s6)    = 0x00000021
```



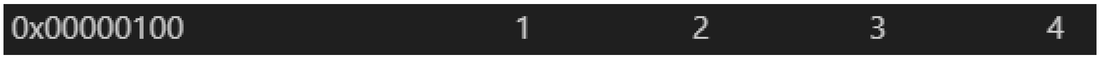
```
x20 (s4)    = 0x00000004
x21 (s5)    = 0x00000028
x22 (s6)    = 0x0000002C
```



```
0x0000030C          44      0      0      0
0x00000308          33      0      0      0
0x00000304          22      0      0      0
0x00000300          11      0      0      0
```



```
0x00000204          30      0     40      0
0x00000200          10      0     20      0
```



```
0x00000100           1      2      3      4
```

Assessment Rubric
Lab 1: Getting Started with RISC-V (Assembly Language) n VS Code

Name	Student ID:	Section:

<u>Points Distribution:</u>

	Task No.	**LR 2 (Code)**	**LR 5 (Results)**
In-Lab	**Task 1**	-	/15
	Task 2	-	/15
	Task 3	/10	/5
	Task 4a	/10	/5
	Task 4b	/10	/10
Total Points: 100		**/30**	**/50**
CLO Mapped		**CLO 2**	

Affective Domain Rubric		**Points**	**CLO Mapped**
AR7	**Report Submission & Git Upload**	/10 & /10	**CLO 2**

CLO	**Total Points**	**Points Obtained**
2	100	
Total	**100**	