

1.

1 A=1 b=2 c=33

2 A=4 b=3

3 c=3

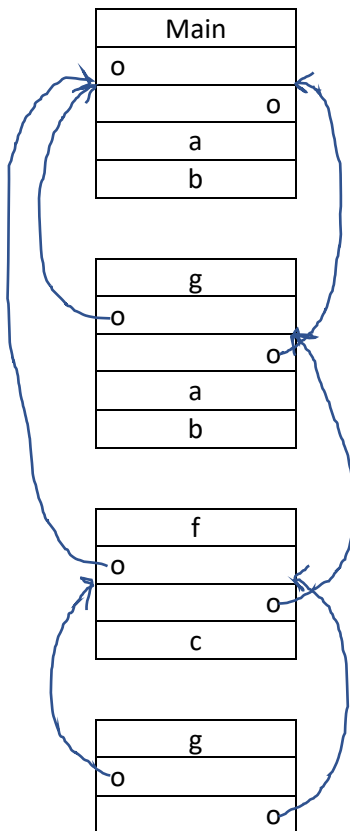
4 A=1 b=9

5 A=5 b=3

6 A=5 b=2

2.

1)



2)

for *A*

loadAI r0, -4 => r1

loadAI r1, -4 => r2

loadAl r2, 8 => r3

loadAl r1, 4 => r4

add r3, r4 => r5

for *B*

loadAl r0, -4 => r6

loadAl r6, 8 => r7

loadAl r0, 4 => r8

add r7, r8 => r9

3.

/*1*/

loadl 3=> r1

storeAl r1=> r0,4

/*2*/

loadAl r0, -12 => r2

load r2 => r3

loadAl r0, -8 => r4

add r3,r4 => r5

loadAl r0, 4=> r6

add r5,r6 =>r7

storeAl r7=>r2,0

/*3*/

loadl 1=> r8

storeAl r8 => r0, -8

program will print 12 for a, and 6 for b.

4.

```
procedure bar (integer a){  
    a = a+1;  
    a = x+2;  
}
```

Call by value	Call by reference	Call by value-result
<p>a = a+1; => a = 2 a = x+2; => a = 3 and it will not affect x in main. Therefore, it prints out 1</p>	<p>a = a+1; => a = 2 a = x+2; => a = 4 since a = &x, first line changes x value in main. and also after 2nd line it also changes x value in main. Therefore, it prints out 4</p>	<p>a = a+1; => a = 2 a = x+2; => a = 3 it will not affect x in main until the procedure bar end. However, after the 2nd line (end of procedure), program is going to change x value in main. Therefore, it prints out 3</p>