Problem Set 2, Part I

Problem 1: Variable scope

```
1) e, i
2) e, i, a, j, b
3) e, i, a
4) e, i, y
5) c
6) c, d
```

Problem 2: String objects and their methods

2-1

```
a) s1.substring(6) + " " + s2.substring(0, 2)
b) s1.charAt(6) + s1.substring(9) + " " + s2.substring(3,8)
c) s1.toUpperCase().charAt(0) + s1.toUpperCase().substring(9) +
s2.charAt(s2.length()-1)
d) s1.charAt(0) + "" +s1.charAt(9) + s2.substring(0,2)
e) s1.charAt(8)
f) s1.substring(8,9)
g) s1.charAt(0) + "" + s2.charAt(0)
h) s1.indexOf(s1.charAt(9))
i) s1.replace(s1.charAt(4), 'u')
```

Problem 3: Understanding code that uses an array

3-1)

i	values
-	{0, 1, 2, 3, 4, 5, 6, 7}
1	{0, 0, 2, 3, 4, 5, 6, 7}
3	{0, 0, 2, 2, 4, 5, 6, 7}
5	{0, 0, 2, 2, 4, 4, 6, 7}
7	{0, 0, 2, 2, 4, 4, 6, 6}

3-2) contents of array just before the method returns:

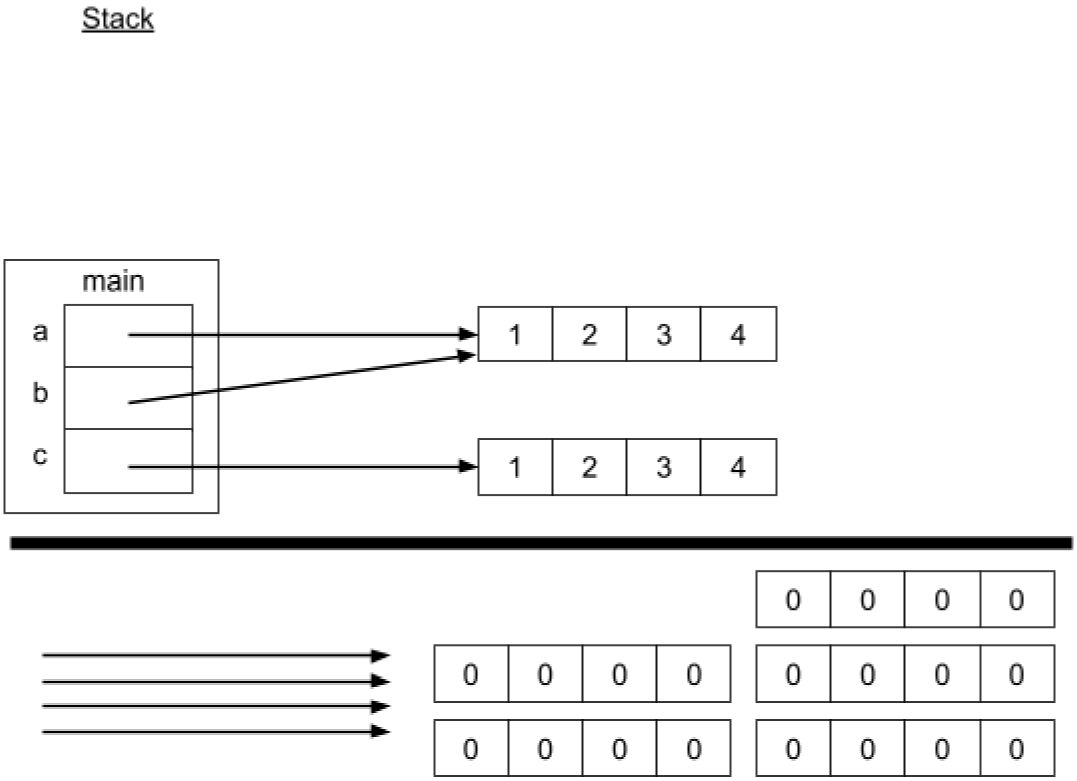
```
{6, 0, 2, 2, 4, 4, 6, 6}
```

3-3)

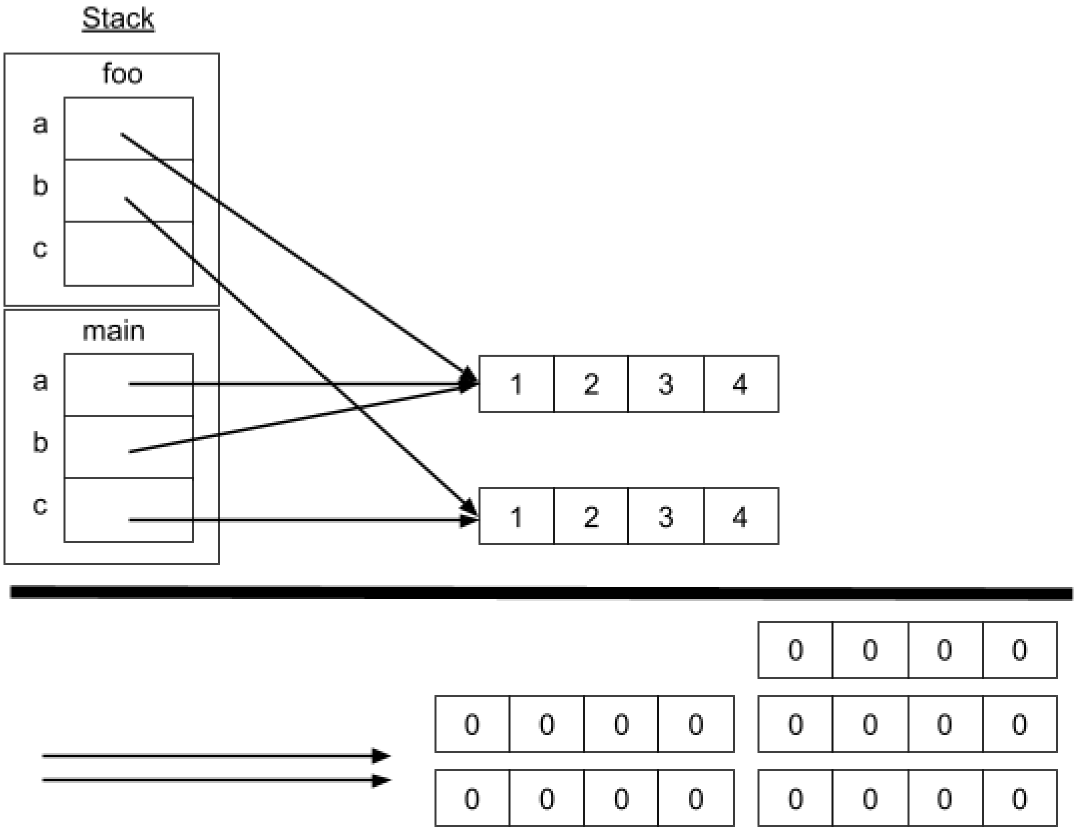
we will see changes made by the call to the mystery() method. When we call mystery(arr), the reference of int[ ] values is the same as int[ ] arr, and the changes made by the mystery() method will also apply to the arrays arr.

Problem 4: Arrays and memory diagrams

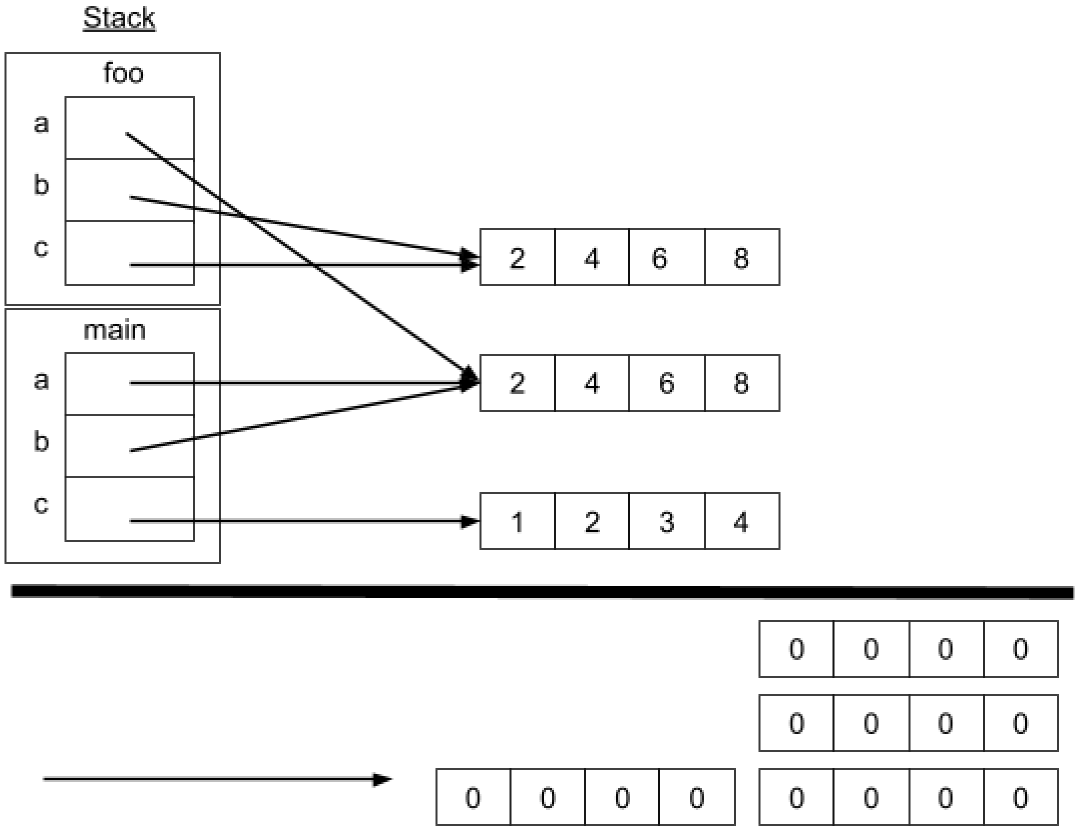
4-1)



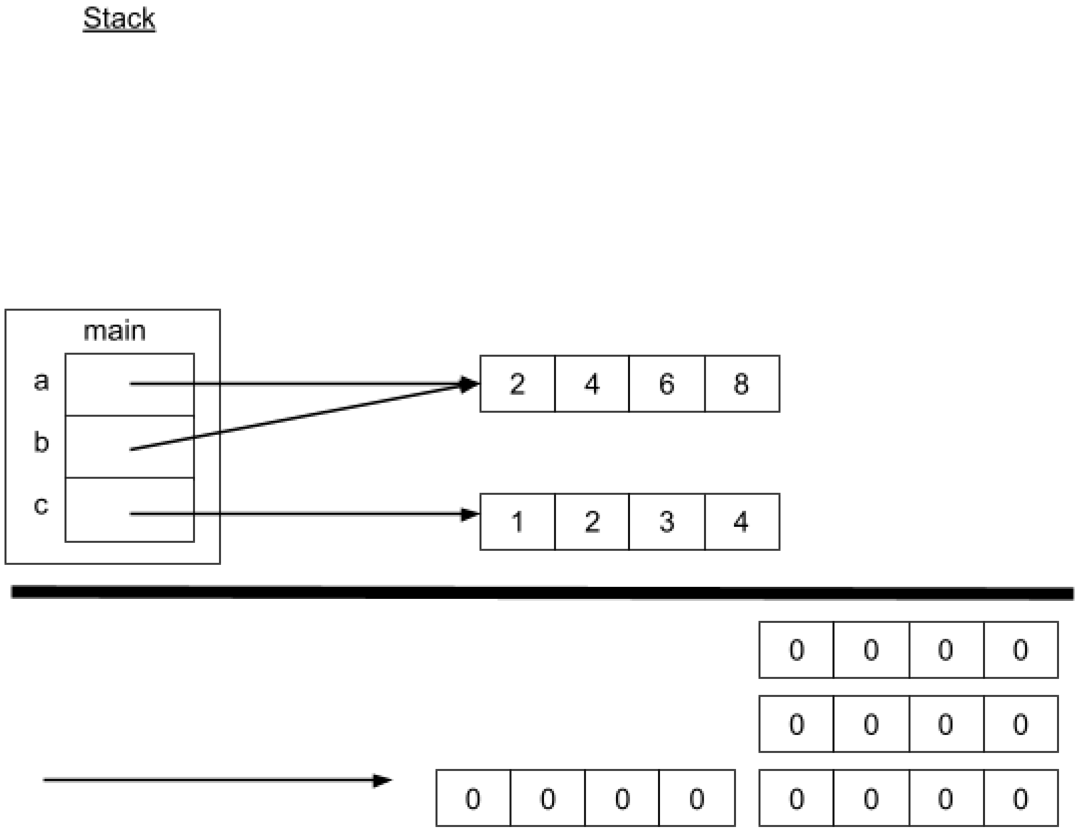
4-2)



4-3)


Stack
foo
a
b
c
2 4 6 8
main
a
b
c
2 4 6 8
1 2 3 4
0 0 0 0
0 0 0 0
0 0 0 0 0 0 0 0

4-4)


Stack
main
a
b
c
2 4 6 8
1 2 3 4
0 0 0 0
0 0 0 0
0 0 0 0 0 0 0 0

Problem 5: Two-dimensional arrays

5-1)

```
twoD[2][1] = 30;
```

5-2)

```
for (int i = 0; i < twoD.length; i++) {
    System.out.println(twoD[i][twoD[i].length - 1]);
}
```

5-3)

```
for (int i = 0; i < twoD.length; i++) {
    System.out.println(twoD[i][i]);
}
```