Introduction

Like on the previous exam, you will be faced with code snippets and an important component is being able to read the code you are seeing within a few seconds and correctly answer the question being asked of you. For this review we will be doing Program analysis and answering some questions about the code snippets shown. I will try my best to group these by difficulty but no promises. Try these on your own prior to viewing the solutions. Good luck!

Easy Difficulty

Consider the following code snippet:

```
private int[] arrOfInts = {1,2,3,4,5};
public void doubleDoubleArray(){
     int[] temp = new int[arrOfInts.length*2];
     for(int i = 0; i < arrOfInts.length; i++){
          temp[i] = arrOfInts[i];
     }
     arrOfInts = temp;
}
```

What is the length of arrOfInts after we compile and run the method above?

a. 5
b. 8
c. 10
d. A Compiler Error Will Occur
e. A Runtime Error Will Occur

Referring to the same code snippet above, how many object reference(s) will be eligible for garbage collection after the method concludes?

a. 2

b. 1

c. 3

d. A Compiler Error Will Occur

e. A Runtime Error Will Occur

Recall the BankAccount class we saw in lecture a snippet is shown below:

```
public class BankAccount{
     private double balance;
     private long accountID;
     private String name;

     public BankAccount(String name, long id){
          //implementation hidden
     }
     public double getBalance(){
          //implementation hidden
     }
}
```

Now say we create a driver class called BankTester.java and we have the following code snippet:

```
public class BankTester {
     public static void main(String[] args){
          BankAccount[] bankAccounts = new BankAccount[args.length];
          for(BankAccount b : bankAccounts){
               System.out.println(b.getBalance());
          }
     }
}
```

Upon successful compilation and execution of the above files, what is the result of the main method in the driver class?

a. 0.0 0.0 0.0

b. 1 2 3 4

c. A Compiler Error will occur

d. A Runtime Error will occur

e. Output Depends

How many instance variables are present in the driver class?

a. 1

b. 2

c. 0

d. 3

e. args.length

How many copies of accountID exist assuming we successfully instantiate 5 BankAccount Objects?

a. 3

b. 1

c. 5

d. 2

e. None of the above

Intermediate Difficulty

Consider the following method:

```
public void mystery(int[] list){
     for(int i = 1; i < list.length-1; i += 2){
          if(list[i] % 2 == 0)
               list[i+1] =  list[i];
          else
               list[i-1] = list[i+1];
     }
}
```

Say we execute the above method on the following list {2,1,5,3,-2}
What will the list look like after the method finishes.

a. 5,1,2,2,2

b. -2,3,-2,1,5

c. 5,1,-2,3,-2

d. A Compiler Error Will Occur

e. A Runtime Error Will Occur

Assume we change the following lines in the method as described below

1. Change the for-loop definition such that we iterate backwards, still ignoring the first and last element
2. Swap the lines in the conditional

If we were to execute this new method on the same list as above what would the list look like after execution?

a. -2,1,-2,3,-2

b. 2,1,1,3,3

c. 5,5,-2,-2,1

d. A Compiler Error Will Occur

e. A Runtime Error Will Occur

Hard Difficulty

Consider the following full class definition that simulates a self-checkout system:

```
import java.util.Arrays;
import java.util.ArrayList;
public class SelfCheckout{
     private ArrayList<Item> basketOfItems;
     private double subtotal;
     private double taxRate;

     public double getSubTotal(){return subtotal;}

     public double getTaxRate(){return taxRate;}

     public double getTotal(){return subtotal * (1 + taxRate);}

     public SelfCheckout(Item[] items, double taxRate){
          this.basketOfItems = new ArrayList<>(Arrays.asList(items));
          this.subtotal = 0.0;
          this.taxRate = taxRate;
     }

     public void calculateSubTotal(){
          //TODO IN QUESTION
     }
}
```

Assuming that we know the Item object has a method called getPrice() that returns the price of the item. Which of the following is not a valid substitute for the comment above:

a.

```
for(Item i : basketOfItems){
     this.subtotal += i.getPrice();
}
```

b.

```
for(int i = basketOfItems.length-1; i >= 0; i--){
     this.subtotal += basketOfItems.get(i).getPrice();
}
```

c.

```
int i = 0;
while(i < basketOfItems.size()){
     this.subtotal += basketOfItems.get(i++).getPrice();
}
```

d. All of the above are valid

e. None of the above are valid

[SCROLL DOWN FOR SOLUTIONS/EXPLANATIONS]

Solutions and Explanations

----

1. C: The original array had length 5 and we assigned it such that it points to the same object as temp meaning the length doubles
2. B: temp was locally defined, making it eligible for garbage collection once the method goes out of scope!
3. E: If args.length is any value greater than 0 we will receive a NullPointerException If args.length is 0 than we print nothing to the terminal
4. C: Generally Driver classes do not contain instance variables
5. C: The accountID variable is an instance variable meaning each object has its own copy.
6. C: You only need to perform one loop iteration to get the answer, the first element we access is odd meaning the element before it gets set to the element after, leaving C as the only possible option
7. B: When we alter the code it looks like the following:

   ```
   public void mystery(int[] list){
       for(int i = list.length-2; i >0; i -= 2){
           if(list[i] % 2 == 0)
               list[i-1] = list[i+1];
           else
               list[i+1] = list[i];
       }
   }
   ```

   Then you can just run through the code on the original array. Thank you to the person on ED who caught this mistake :D
8. B: Remember ArrayLists use size() to give you how many elements are in there not .length