Please write only on the given booklets, not on these instructions.
Write your name on both the booklets and the instructions.
Return both the booklets and the instructions.

1. Provide short answers (at most two sentences each) to the following questions.
   1.a What is the purpose of casting?
   1.b How do you generate an integer random number between 1 and 12 (inclusive) and assign it to a int variable num?
   1.c What is wrong with the following code? Please give a way to correct it.
      public static void giveBack(int r) {
          ... // Some instructions doing something with r
          return false;
      }
   1.d The variable a, of type int[] refers to an array of length 15. You now would like it to refer to an array of length 12. How can you achieve that (write the code)?

2. What is the output when the following program ArrayCall is run?

```java
public class ArrayCall {
    public static void main(String[] args) {
        int[] q = new int[16];
        for (int i = 0; i < q.length; i++) {
            q[i] = (i * 2) / 3;
        }
        int z = 13;
        int y = duck(z, q);
        int[] a = q;
        z = duck(z, a);
        System.out.println(y);
        System.out.println(a[z-1]);
        System.out.println(q[z]);
        System.out.println(q[z-1]);
        System.out.println(q[z+1]);
        System.out.println(a[z+1]);
        System.out.println(q[q.length -1]);
    }
    public static int duck(int x, int[] a) {
        x = x - 1;
        int t = a[x];
        a[x] = a[x] + 3;
        return t;
    }
}
```
3. Write a method `findAll` with the following specifications. Its parameters are two arrays of integers called `d` and `a`. The array `d` contains distinct integers (i.e., all values in this array are different), while `a` may contain duplicate values. The method should return true if all values stored in `d` are present in `a`, and false otherwise. The array `a` may contain values that are not in `d`. Do not assume that `d` and `a` are sorted.

4. Write a method `findAllSorted` that behaves like the method `findAll` from Exercise 3, but assumes the arrays `d` and `a` are sorted in non-decreasing order.

5. Write a variant `zdInsertionSort` of `insertionSort` that takes an array `a` of positive integers, and sort the array, while replacing all but one copy of every duplicate value with a 0 (zero). All the zeroes must appear, when the array is sorted, in positions with lower index than any positive value. For example, the input array `|4|6|3|4|2|4|3|` should become `|0|0|0|2|3|4|6|`. 