Problem 1: [10 points]

Below, define a static method called `largest` that takes three integers as inputs and returns the largest of the three.
Problem 2: [16 points]

Below is some modestly mysterious map code that makes use of the Die class we’ve seen so often. Describe briefly, in plain English, what the mystery method does. What would be a good, descriptive name for this method?

```java
public static int mystery(Die x, Die y) {
    int a;
    int b;
    int c = 0;
    do {
        a = x.roll();
        b = y.roll();
        c = c + 1;
    } while(a != b && c < 1000);
    if (c >= 1000) {
        return -1;
    } else {
        return c;
    }
}
```
Problem 3: [20 points]

a) What characteristic must an array of numbers have in order to use the binary search algorithm on it? Why?

b) What does the final keyword mean when used in defining a variable? Why might you use it in a program?

c) What does the static keyword mean when used in defining a variable? Why might you use it in a program?

d) What’s the difference between a for and a for-each loop? Give some examples of when you might use each.
Problem 4: [16 points]

We tested our selection sort code by passing it a few short arrays — short enough that we could manually inspect the results and verify that the method was sorting arrays correctly. It would be nice to automate that testing, and to be able to verify that much longer arrays were sorted correctly as well. Below, define a static method called \texttt{ordered}, that takes an array of integers and returns \texttt{true} if they’re in order from smallest to largest, \texttt{false} otherwise. (Hint: You can check that the values are in order by making a single pass through the array, comparing neighboring items as you go.) For full credit, your solution should handle duplicate items properly.
Problem 5: [18 points]

On the Keypad assignment, the constructor was passed a string containing the characters to be mapped onto a 2D array of “keys”. The resulting Keypad could behave in unpredictable ways if the same character appeared on more than one key, however, so it would have been nice to verify that there were no duplicate characters in the input string. Below, define a static method called `containsDuplicates` that takes a `String` as its input and returns `true` if it contains duplicate characters anywhere within the string (`false` otherwise). For full credit, don’t use any methods from the `String` class except `charAt`. 
Problem 6: [20 points]

The selection sort method we wrote in class (and that you worked with in lab) is shown below, with some portions omitted. Fill in the six blanks with the appropriate missing code.

```java
public static void selectionSort(int[] nums) {
    for(int start=0; start<nums.length-1; start=start+1) {
        int indexOfMin = _____________;

        for(int i=start+1; i<____________; i=i+1) {
            if (nums[i] < nums[indexOfMin]) {
                indexOfMin = i;
            }
        }
        int temp = nums[_____________];
        nums[_____________] = nums[_____________];
        nums[_____________] = temp;
    }
}
```