**Java 1: Test #2 Outline**

Tuesday, December 17, 2017

Here are some general guidelines on what you should expect for the test. Please know the test will be written using the IDE in class. I expect **everyone to remove** all previous projects in the IDE before the test begins. I will not be providing a practice test with questions for this test, although, below I provided general guidelines for the questions. For best preparation, review the relevant coding examples I have provided for our lectures and your most recent assignment 2. The test will cover **lectures 6 – 10**. In other words everything from arrays up to strings. I will also provide any mathematical equations on the test or methods I expect you to use for a given question. If you have **any questions** please let me know right away.

* To keep things clear **I will not be asking**:
  1. Reading input from a file.
  2. JFileChooser.
  3. JOptionPane dialog boxes.
* Most importantly since I am providing use of Netbeans the test will be graded based on the following:
  1. The **output** of the solution.
  2. Code **readability** such as naming conventions and meaningful variable names.
  3. Best practices I have thought such as isolating code into methods.
  4. **Comments**, I **do not expect** extensive comments, but I expect a couple lines explaining logic or methods where the code may be a little confusing.
* There will be one question per topic on the test:
  1. Question 1: Arrays
  2. Question 2: 2D Arrays
  3. Question 3: Objects
  4. Question 4: Strings
* For question 1 you should know how to:
  1. Declare an array.
  2. Set an array’s length through user input.
  3. Fill an array from user input.
  4. Fill an array from randomly generated numbers.
  5. Perform calculations on the array such as sum, average, determining min and max.
  6. Searching through an array.

* For question 2 you should know how to:
  1. Declare a 2D array.
  2. Set a 2D array’s row length and col length.
  3. Fill a 2D array from user input.
  4. Fill a 2D array from randomly generated numbers.
  5. Perform calculations on the array such as sum, average, determining min and max.
  6. Searching through a 2D array.
  7. Printing out a 2D array from either rows or columns.
* For question 3 you should know how to:
  1. Create a new file to build a class.
  2. Declare data members.
  3. Specify data member visibility:
     1. Private
     2. Public
     3. Package access
  4. Specify data member modifiers:
     1. Instance
     2. Static
  5. Create constructors, with or without parameters.
  6. Utilize the **this** operator for data members or redirecting to another constructor.
  7. Create class behaviors (methods)
  8. Write getters and setters.
  9. Write a class’ to string method.
  10. Create an object based on a class; utilize the objects constructors and methods.

* For question 4 you should know how to:
  1. Create a string.
  2. Obtain the substring of a string.
  3. Replace characters in a string.
  4. Split a string based on a given character.
  5. Iterate through a string.
  6. Comparing strings for equality or if one string is greater than another is.

Here is a set of questions that will help you in preparation for the logic of the questions:

1. Ask the user to enter a range of numbers. The program will then use those two numbers to fill an array with randomly generated numbers between that range.
   1. **Hint:** I will provide how to use Math.random()
   2. Average these numbers.
   3. Sum these numbers.
   4. Determine the min and max.
   5. Count the occurrences of numbers above and below the average of the numbers.
2. Write a program where the user can choose a number of students and a number of grades. This will be the rows and columns for a 2D array respectively.
   1. Have the user fill in the grades for each student.
   2. Perform operations similar in point 1.
   3. Iterate through either the rows or columns.
3. Write a class which represents a house based on the following properties:
   1. How many doors does the house have?
   2. What material is the house made of?
   3. How many windows does the house have?
   4. How many square feet is the property?
   5. What color is the house?
   6. How many rooms does the house have?
   7. How many floors does the house have?
   8. How many houses have been created in total?
   9. Calculate the cost of the house based on a mathematical equation regarding its properties.
4. Write a program where a user can enter a string and the program will count the occurrences of each letter in the string. For example: If the input is ‘aaaabbdd’. The program will output:
   1. The character ‘a’ occurs 4 times.
   2. The character ‘b’ occurs 2 times.
   3. The character ‘d’ occurs 2 times.