Lab 11  
Name:_________________________  Checked:_____

Objectives:
More practice using arrays:
1. Arrays of Strings and shuffling an array
2. Arrays as parameters
3. Collections

Preparation
Submit DeckOfCards.java and TrianglePanel.java through Blackboard by 8:00am the morning of the Lab.

1) An array to store a deck of cards: DeckOfCards.java  
Do as much as you can of this part (at least through instantiating the array in (c). If you get stuck, you can complete it in the lab

a. Use initializer lists to create the following arrays:
   • An array of 4 Strings to represent the suits: “hearts”, “spades”, “diamonds”, “clubs”
   • An array of 13 Strings to represent the ranks: “2”, “3”, …, “10”, “Jack”, “Queen”, “King”, “Ace”

b. Declare and instantiate a third array of 52 Strings to represent the deck of 52 playing cards (we will initialize the values of the array below).

c. Print the contents of all three arrays.
What happens if you do not initialize the array's values?

   Answer: __________________________________________________________

   d. Use the arrays for suits and ranks, above, to initialize the deck, so that it consists of Strings like “4 of hearts” or “King of clubs”, i.e., a rank, followed by the string “of” followed by a suit.

   Hints:
   • With a nested loop, combine each suit with each rank, concatenating them to make Strings such as “4 of hearts”
   • As the strings are combined, they should be stored as one of the 52 array elements representing the deck. Thus, you will need a counter to index into the deck array. You need to start that at zero and increment it each time an entry is added (in the inner loop).

<table>
<thead>
<tr>
<th>suits</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>hearts</td>
<td>spades</td>
<td>diamonds</td>
<td>clubs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ranks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>deck</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>2 of hearts</td>
</tr>
</tbody>
</table>
2) **Arrays as parameters:** *TrianglePanel.java*
   a. Run the example *TrianglePanel.java*. Note the use of arrays as parameters.
   b. Change the numbers so that you get nice blue V on a white triangle.
   c. (bonus) Take a selfie of yourself with your V and submit along with your code!

3) **Collection:** *ShoeCollection.java*
   a. Review the textbook example of a DVD database: *DVD.java, DVDCollection.java* and *Movies.java*.
   b. Aren't shoes better than movies? Maybe you disagree, but, in any event, we will create a program similar to the one above, using shoes instead of DVDs. Using the Shoe class you designed for one of your earlier projects, create a class called ShoeCollection, similar to the DVDCollection – it should maintain a database of Shoe objects, using an array of Shoes. Create a program *YouVeGotMoreShoes.java* to test *ShoeCollection.java*.
   c. In the space below, sketch the UML class diagram for your program. (Hint: It should be similar to the UML class diagram of the textbook example.)
Lab 11 Part 1 - DeckOfCards.java

a) Compare your work with your partner’s. Verify that arrays are declared, instantiated, and initialized correctly. If necessary, help each other complete part d of the preparation.

Lab partner’s signature: ________________________________

b) Add code to shuffle the deck as follows:

  • Generate two random numbers a and b in the range 0..51
  • Exchange the values (Strings) in deck[a] and deck[b]. For example, if the numbers generated were 2,50, then the “4 of hearts” and the “king of clubs” would switch places in the array.
  • Repeat many, many times (how many do you think would be enough?)

After the deck is shuffled, print it again and behold the cards in random order!

Lab 11 Part 2 - TrianglePanel.java

a) Compare your work with your partner’s. Test each other’s code.

Lab partner’s signature: ________________________________

b) Add the method addTen() to the TrianglePanel class and incorporate some code in the paintComponent() method to use addTen() so as to shift the blue V to the right.

  public void addTen(int[] a)
  {
    for (int i = 0; i < a.length; i++)
      a[i] += 10;
  }

c) Create another method addN() similar to addTen() which gives you more flexibility by using a second parameter to allow you to add any value to each element of the array. For example addN(anArray, -72) would subtract 72 from each element of anArray. Try it with TrianglePanel to draw additional triangles. Finish by creating a design of your choice.

d). Compare your design with your partner’s. Sign and optionally comment on their design – take a selfie together with your V’s – Go Cats!

Lab partner’s comments: __________________________________________________________

Lab partner’s signature: ________________________________
Lab 11 Part 3 - ShoeCollection.java

a) Check your UML class diagram with your partner.

Lab partner’s signature: ____________________________

b) Make sure that YouVeGotMoreShoes.java has at least 10 shoes added to the collection (if necessary, make up a few more to add).

c) Add a shuffle method to the ShoeCollection class. Use it in YouVeGotMoreShoes.java to shuffle the shoes and print them in some random order.
Lab 11 Comments  
Name:__________________________

Comments on this lab, please:

What was the most valuable thing you learned in this lab?

What did you like best about this lab?

Was there any particular problem?

Do you have any suggestions for improving this lab as an effective learning experience?