CS212 - Lab 6a

Writing and Testing the UnorderedArrayList Class

GOALS:

- Practice creating subclasses
- Gain a better understanding of how the UnorderedArrayList class is implemented
- Practice testing methods within the UnorderedArrayList class

Problem:

The purpose of this lab is to develop and test the UnorderedArrayList class. I have provided the parent class, ArrayList, along with documentation for each of the methods defined in ArrayList. You will write and test each method/constructor prior to writing the next one.

Creating the Project:

- Create a new project **Lab6_LastnameFirstinitial** and add a new package called **lab6a**.
- Copy **sqlite-jdbc-3.7.2.jar** to the project folder, and add it as a library to the Java Build Path for the project. This jar file is required for the getData method in Lab6aApp.
- Copy the **products.db** file to the project folder. This is a SQLite database file containing products.
- Copy the class files **ArrayListADT**, **ArrayList**, **Lab6aApp**, and **Product** into the project.
- Within this project, create a new class named **UnorderedArrayList**.

Exercise

In the exercises below you are asked to include code in the application class in order to test the methods defined in the UnorderedArrayList class. Leave all of this code so that I can see how each method was tested. Do not modify code that was previously used to test one method in order to test another.

Comments for the add (insert method), indexOf (search method), and remove methods are provided in ArrayList; they are correct and are intended to provide you with information about what each method is supposed to do. Make the necessary modifications to each method before beginning to test another. Include appropriate comments in the application class to explain your code.

Modify the UnorderedArrayList Class:

1. Define a default constructor and a parameterized constructor. Recall that **UnorderedArrayList is a subclass of ArrayList**.

2. Provide “stubs” (see below) for the abstract methods add, indexOf, lastIndexOf, contains, get, and remove. This will allow you to compile the class and to develop and test each one individually.

```
public boolean add(T insertItem) { return true; }
public int indexOf(T searchItem) {return 0; }
public int lastIndexOf(T searchItem) { return 0; }
public boolean contains(T searchItem) { return false; }
public T get(int index) { return null; }
public T remove(T removeItem) { return null; }
public T remove(int index) { return null; }
```

Modify Lab6aApp:

3. Define the main method.

4. Create an UnorderedArrayList object, **list1**, using the default constructor. Display the contents of the list.

5. Create a second list, **list2**, using the parameterized constructor, and specify a maximum size of 5. Display the contents of the list.

Modify the UnorderedArrayList Class:

6. Provide the necessary code for the **add** method.

Modify Lab6aApp:

7. In order to test the **add** method you will need to create references to Product objects and instantiate them using the parameterized constructor. The data below should be stored in these Product objects. Create a variable called **products** (a reference to an array of Product), call the **getData** method which returns a Product array, and store the reference returned in **products**.

   You will have to change the following statement to specify the correct path to the database file: Connection c = DriverManager.getConnection("jdbc:sqlite:I:/aabreu/products.db");

   Note: if you need to specify a different location (path) for the file, change the I:/aabreu/products.db.

8. Use the Product array from step 7 to insert each of these into **list2**. Create a loop to insert and display the resulting list after each insertion, to determine if the insert method is working. If not, make the necessary corrections and test it again until it works as expected.

Modify the UnorderedArrayList Class:

9. Provide the necessary code for the **indexOf** and **lastIndexOf** methods.

Modify Lab6aApp:

10. Create a method, **testSearch(key, ulist)**, and write the statement(s) necessary to locate the position of a product in the list. Call the method and pass the product whose id is "264j45" (an item found in the middle of the list). If it is found, display the product id and its location; otherwise display a message stating that the item was not found

11. Call the **testSearch** method and pass the product whose id is "344d97" (an item that is not in the list). If it is found, display the product id and its location; otherwise display a message stating that the item was not found.

12. You should test your **indexOf**, **lastIndexOf**, and **contains** methods with other product ids. Include additional code to search for an item based on index and key field:
    - that is found in the beginning of the list
    - that is found at the end of the list
    - in an empty list (use list1). Be sure to display an appropriate message indicating whether the product was found.

Modify the UnorderedArrayList Class:

13. Provide the necessary code for the **remove** method.

Modify Lab6aApp:

14. Create a method, **testRemove(key, ulist)**, and write the statement(s) necessary to remove a product from the list. Call the method and pass the product whose id is "355d98" (an item in the middle of the list) and the list. Display the items in the list to ensure it has been deleted.

15. Call the testRemove method and pass the product whose id is "344d97" (an item that is not in the list). Display the items in the list to ensure that the list has not changed.

16. You should test your **remove** method with other product ids. Include additional code to remove:
    - an item that is found in the beginning of the list
    - an item that is found at the end of the list
    - a list with 1 item (add a product to list1 and remove it)
    - an item from an empty list (use list1)

    be sure to display the list after attempting to remove each of these to ensure that your method is working correctly.

17. Comment out the **equals** method in the **Product** class. Run your program again. Do you get an error message? Why or why not? How does your output change (if at all)? You should include your answers to these questions in a comment at the bottom of the main method.

18. Remove the comments from the equal method. Run your program again to ensure that you haven't introduced any errors.

Submit:

Implement all remaining methods in the **UnorderedArrayList** class and test them in the Lab6aApp class. Modify the comment at the beginning of the 4 files (excluding Product) to include an appropriate title, description, and your names as the authors. **Export and submit the entire project to your lab instructor. Each person must submit the project**.