Frank Hucek
Programming Project 5
EECS 132

Testing Report

Below are listed the methods that are JUnit tested within each class. The descriptions describe what each method does and how thorough the JUnit test is. Getters and setters are assumed to work and are therefore not tested.

DatabaseType interface

Contains one method stub. Only things that implement this interface can be stored in the Database Class. Because this is an interface, it cannot be tested with JUnit, rather anything implementing it will test getComparatorByTrait.

public Comparator<T> getComparatorByTrait(String trait);
Returns a comparator based on the given trait.

Database<T extends DatabaseType> implements Iterable<T> Class

Database stores anything that implements DatabaseType and has many methods for storing and looking up this entries in the database.

public boolean isEmpty()
Tests whether the list of nodes in this database is empty. Tested in JUnit with an empty database and one with entries.

public void add(T element)
Adds an element to the database. Tested in JUnit by making sure the element was actually added to the list of nodes.

public void delete(T element)
Deletes an element from the database. Tested in JUnit by making sure the element was actually deleted from the list of nodes. Every occurrence of the element is deleted.

public LinkedList<T> lookupInList(T value, Comparator<T> c)
Returns a LinkedList of all occurrences of the input value in the Database. Determines equivalency through the input comparator. Tested in JUnit by taking a Database and making sure calling this method really returns every occurance of the input value.

public LinkedList<T> lookupInIndex(T value, ArrayList<T> index, Comparator<T> c)
Returns a LinkedList of all occurrences of the input value within the input ArrayList, using the input Comparator to test for equivalency. JUnit tests are similar to the previous method's tests, but with an ArrayList and not the Database's list of nodes.

public void makeIndex(String trait)
Creates an index for the Database's data based on the given trait. Adds this created ArrayList to the Hashtable with the input trait as the Hashtable's key, and the sorted ArrayList as its value. Tested in JUnit in tests for **getList()** method by taking a database with entries and calling this method and checking the contents of the Hashtable to see if a sorted ArrayList with its corresponding trait were added to the Hashtable.

public LinkedList<T> lookup(String trait, T value)
Returns a LinkedList of the data in the database that match the value based on the value's comparator. For speed, this method first calls lookupInIndex for an associated ArrayList to the input trait within the database's hashtable. If none is found, calls lookupInList to return that LinkedList. Tested in JUnit similarly as lookupInList and lookupInIndex but cases of indexes are accounted for.

public ArrayList<T> getList(String trait)
Returns the ArrayList associated with the input trait within the Database. If none exists, It makes one by calling makeIndex, then returns the new associated ArrayList. Tested in JUnit with things that aren't already an index and things that are.

public class LLNode<T>
This class represents a node, its element, and its next node. This class was created during lecture and used throughout the course, so is not tested with JUnit. It merely contains getter/setter methods for this node's element, and its next node.

public class Contact implements DatabaseType<Contact>
This class represents a contact. It implements DatabaseType and can therefore be stored as a type within the Database class. This class stores the name, phone number, and email of a contact. The getters and setters for these values are not JUnit tested. All other methods are described below and JUnit tested.

public boolean equals(Object o)
Overrides the equals method of Object class. Determines whether two Contacts are equal based on their names, phone numbers, and emails. JUnit test checks truth of this method when tested with contacts that really aren't equal and contacts that really are equal.

public String toString()
Overrides the toString method of Object class. Simply Returns the Contact's name as a String. Tested in JUnit by checking to see if the Contact's name really was returned after calling this method.

public Comparator<Contact> getComparatorByTrait(String trait)
Overriden from DatabaseType's method stub. Based on the input trait, creates a comparator that compares 2 Contacts values based on trait. Tested in JUnit by calling this method for all different traits (name, phone, email) on various contacts and see what is returned by calling the comparator's "compare" method on two contacts.

public class ContactDatabase extends Database<Contact>
This class is a Database that stores contacts. It contains a print method to print all entries within the database. Also contains a main method for user usage. The main method is not JUnit Tested because the main method serves as its own test upon run time.

public void printList(Iterable<Contact> it)
For each contact in the input iterable, it prints out each value within the database on their own numbered lines.