Lab 5: Hash Table

This lab is designed to give you practice working with hash table. This is an individual assignment; you may not share code with other students.

Specification

Overview

Your task is to complete the definition of a `HashTable` class using separate chaining method to handle collision. Specifically, you will:

- Implement hash functions for integer and string keys
- Implement a hash table search operation
- Implement a hash table remove operation

Preliminary

Download the zipped project archive — Lab 4.zip, and import it into your preferred IDE as an existing project.

Open the `hashtabletest` package and run the `TestFrame` class. Clicking the `insert` button will insert new elements into the hash table. Note that the integer and string keys always hash to zero — you will fix this later.

Open the `hashtabletest` package and run the `HashTableTest` class . Note that the `testSearch` and `testRemove` tests should fail — you will fix this later, too.

Your Job

You need to complete the following methods in the `hashtable.HashTable` class:

- `integerHash` and `stringHash` methods: to exhibit good hashing performance for integer and string keys
- `search` method: to find and return the hash table element for a given key
- `remove` method: to remove a hash table element once it is found

Testing Your Hash Functions

After you have implemented `integerHash` and `stringHash` , test them by running the TestFrame.java class.

You should be able to insert random integers, random strings, and permuted strings with good hash function performance. Good hash function performance would generally (but perhaps not always) show 1.0 or less for a standard deviation from the mean chain length after `CAPACITY` (in this case, 11) inserts.

Testing Your Hash Table Operations

The `HashTableTest` class tests your search and remove operations. When you have successfully implemented these operations this class will run without errors.

Submission

Raise your hand to call for the teaching assistant and demonstrate your results. Create a zip file named YourStudentID.zip that contains your code project and upload your zip file to the https://wss.pet/s/3tnhqn8d5da.

You should implement your own hash table data structure and may not be confused with the java.util.HashTable class available in the Java standard library.

Deadline

26 Oct 2020 23:59 GMT+08:00

Appendix

Java Generics

In Java, a generic type is a class or interface that is parameterized over types. We use angle brackets (<>) to specify the type parameter. For example, a type of `ArrayList<String>` means the each element in the array list is of String class.