

Programming Assignment 4 - Hash Tables

[Submit Assignment](#)

Due May 16 by 11:59pm **Points** 100 **Submitting** a file upload
File Types java and cs **Available** May 5 at 8:50pm - Jun 18 at 11:59pm about 1 month

Hash Tables

For this assignment you will implement a hash table that meets the following requirements:

- the underlying data structure, `hashArray`, is a fixed size Java array (not an `ArrayList`) of type `Integer` (not `int`)
- resolve collisions with chaining
- the type of list for chain is a Java `LinkedList<Integer>`
- when the loading factor reaches 1.0, resize `hashArray` to twice it's current size and rehash
- the hash function is

```
hashValue = item % hashArray.length
```

- the initial length of the hash array is passed to the constructor as a `int` parameter
- `HashTable`, the class that defines the hash table must have the following minimal set of public methods:
 - `HashTable(int sz)` //a constructor that takes a single `int` parameter
 - `void insert(Integer key)` //takes an `Integer` parameter and stores it in the list
 - `void remove(Integer key)` // find an item in the hash table and remove it
 - `Integer search(Integer key)` // searches for an item in the hash table and if found, return it. Otherwise, return null
 - `void toString()` // outputs the contents of the hash table, one row per element in `hashArray`
 - `int size()` // return the number of elements in all of the chains
 - `int size(int index)` // return the number of elements in the indexed chain
 - `double loadFactor()` // a method the calculates and returns the loading factor to the caller
- and the following private method(s):
 - `rehash()` - instantiate an `Integer` array that is the next prime number greater than twice the size of the original hash array. Then for each element in the original hash array, rehash it into new array (don't forget that the hash function is based on the size of the hash array)

When you are finished, upload the java files to canvas.

Extra Credit

In order to earn extra credit on this assignment, you have fulfill all of the requirements of the assignment. If that's the case, the for extra credit points on this assignment if the loading factor drops below 0.2, reduce the size of hashArray by half, and rehash. (Hint: If you're adding data to an empty hash table, and the loading factor moves up to or above 0.2, that's not dropping to 0.2. The change in loading factor has to be dropping when the decision is made to reduce the size of the hash array).