

CS 1331 Programming Quiz 3 - HashMap

July 18, 2017

Important Note: After you submit your quiz make sure you bring your buzzcard to the front of the room and get checked off by a TA. If you fail to do so your quiz will NOT be graded.

Problem description

You will be creating a simple HashMap using generics. A HashMap is a data structure that stores key-value pairs.

- You may NOT use the HashMap class in the java.util package!
- You may only use an outside data structure in the KEYSET method.

You have been provided with a few files, some of which are incomplete. Let's review them:

HashMapInterface.java

This interface contains all of the methods you will be implementing in your `HashMap`. See javadocs for method details.

Note: We have given you special rules for some of the methods so read the javadocs carefully.

HashMap.java

This class has been partially implemented for you and represents a simple `HashMap`. It **must** implement `HashMapInterface`.

You must create a constructor for your `HashMap` that takes in an `int` representing size and instantiates the backing array `backingArr` to be of that size.

You have been provided with the following:

- A private `Pair` inner class. **Do not edit this class.**
- A private instance variable of type `Pair<K,V>[]` and name `backingArr`. You **must** use this as the backing array for your `HashMap`. **Do not edit the name, type, or visibility of this variable.**

Tester.java

This class is for testing your code. Some basic tests have been provided, however, getting the optimal output does not guarantee you a specific grade. Feel free to edit this file as you wish. **This file will not be graded.**

Tips

- The warning "Note: Recompile with -Xlint:unchecked for details." is **FINE** and you will **NOT** be penalized for this.
- You **are** allowed to use the Math class.
- The modulus operator (%) may be useful.
- Remember, all objects have a hashCode() method!
- hashCode() may return ANY int, positive or negative.
- If there is some section of code that does not compile and you cannot fix it in time, comment it out. Partial credit is better than no credit!

Clarifications

Points will not be taken off for checkstyle and you do not need to write javadocs for your code.

You should not import any libraries or packages that trivialize the assignment. This includes data structures such as `List`, `Map`, etc. `System.arraycopy` is not allowed. If you are unsure of whether something is allowed, raise your hand and ask a TA. In general, if something does a large part of the assignment for you, it is probably not allowed.

The only websites you are permitted to use are T-Square (<https://t-square.gatech.edu>) to **download and submit the quiz** and the Oracle Java API (<https://docs.oracle.com/javase/8/docs/api/>). **You may not use any other references including class notes, powerpoints, or code not provided with the quiz.**

Compiling and Testing your code

Compile your code from the `PQ3` directory with the command `javac *.java`. Test your code with the command `java Tester`. Verify your output by checking the comments in the `Tester.java` file. This tester is not exhaustive so feel free to edit it.

Turn-in Procedure

YOUR SUBMISSION SHOULD ONLY CONTAIN .JAVA FILES! DO NOT SUBMIT .CLASS FILES!

Submit your `PQ3.zip` file on T-Square as an attachment. This zip file should contain `HashMap.java` and `HashMapInterface.java`. When you're ready, double-check that you have submitted and not just saved a draft.

After you submit your quiz make sure you bring your buzzcard to the front of the room and get checked off by a TA. If you fail to do so your quiz will NOT be graded.

Verify the Success of Your Submission to T-Square

Practice safe submission! Verify that your quiz files were truly submitted correctly, the upload was successful, and that your program runs with no syntax or runtime errors. It is solely your responsibility to turn in your quiz and practice this safe submission safeguard.

- After submitting the files to T-Square, return to the Assignment menu option and this quiz. It should show the submitted files.
- Download copies of your submitted files from the T-Square Assignment page placing them in a new folder.
- Re-run and test the files you downloaded from T-Square to make sure it's what you expect.
- This procedure helps guard against a few things.
 - It helps insure that you turn in the correct files.
 - It helps you realize if you omit a file or files. Missing files will not be given any credit, and non-compiling/non-running solutions will receive few to zero points. Also recall that late quizzes will not be accepted regardless of excuse. (If you do discover that you omitted a file, submit all of your files again, not just the missing one.)
 - Helps find syntax errors or runtime errors that you may have added after you last tested your code.