Lab 10

Reminder: Your code is to be designed and written by only you and not to be shared with anyone else. See the Course Outline for details explaining the policies on Academic Integrity. Submissions that violate the Academic Integrity policy will be forwarded directly to the Computer Science Academic Integrity Committee.
All materials provided to you for this work are copyrighted, these and all solutions you create for this work cannot be shared in any form (digital, printed or otherwise). Any violations of this will be investigated and reported to Academic Integrity.

Objectives

- Exposure to hash tables
- Practice with insertion into hashtables

Part I

1. Download `Lab10Tester.java`, `Hashtable.java` and `Student.java` to your Lab10 working directory. You will be implementing insert and get in 2 ways:
   1) no collision handling and 2) collision handling with open addressing using linear probing.
2. `Lab10Tester.java` has 5 TODO notes. Follow the instructions in each TODO note.

NOTE: `Student.java` has a `hashCode` function. You should call this function to get the hashcode for a given student. Use the % operator to convert that hashcode to an index within the bounds of your table.
Example:

```
int hashcode = someStudentObject.hashcode();
int index    = hashcode % SIZE_OF_TABLE;
```

CHECK POINT – get help from your lab TA if you are unable to complete these checkpoints:

CHECK POINT 1: TODO 1,2: implement and test `insertCollisions` and `getCollisions` methods
CHECK POINT 2: TODO 3: implement and test `insertLinearProbing` method
CHECK POINT 3: TODOs 4,5: implement and test `getLinearProbing` method

Finished early – start your Assignment!