CS 3305/W02: Data Structures

Assignment 9 - Hashing Check D2L for Due Dates (100 points total)

GENERAL SUBMISSION REQUIREMENTS

Upload all files individually as specified, not as zip files, to Assignments in D2L. Do not email files.

Make sure your program compiles, runs and produces the correct output.

Ensure you have the correct file name(s), and author header, as specified in the Assignment.

Always use meaningful labels for prompts, inputs, and outputs.

Always use comments, indentation and whitespace as shown in examples.

Note: Never hard-code test data in the test program, unless explicitly stated in the assignment. Always allow the user to enter the test data using a menu option.

Assignment 9 – Hashing ______________________ (100 points) :

There is only one Part to this assignment, Programming Exercise 27.1 from the Liang textbook, end of the Chapter Programming Exercises, re-printed below.

In the text you will find a custom Map interface to mirror java.util.Map. The name for that interface is

MyMap and there is a concrete class MyHashMap, as shown in Figure 27.9, page 995 of Liang text.

Implement MyMap using open addressing with linear probing. Hint: You must create a new concrete class that implements MyMap using open addressing with linear probing, as described on p. 989. See the lecture on linear probing. Note, you may use an array to implement the hash-table.

For simplicity, use f(key) = key % size as the hash function, where size is the hash-table size. Initially, the hash-table size is 4. The table size is doubled whenever the load factor exceeds the threshold (0.5). Note: you will need to write a different hash function ie you can't use the one in the book.

Do not forget to include author header in each submitted file as shown, no header, no points!

```
// Name: <your name>
// Class: CS 3305/ put your section number after the /
// Term: Semester Year
// Instructor: Sharon Perry
// Assignment: 9-Part-1-Hashing
```

Capture a **READABLE** screenshot(s) of your program output and paste into a word/pdf document. Readable means readable! Screenshots ***should not be an entire desktop*** – use some type of snipping tool. After your output screenshots, copy and paste the source code for your program into the word/pdf doc. Save doc as a file named LastName-A9-Part-1-Hashing.docx or .pdf. Last step is to upload word/pdf and .java files to D2L.

SUBMIT YOUR OWN CODE – Code copied from the internet will receive a score of zero.

MAKE SURE YOUR CODE HAS COMMENTS ! We are getting submissions without comments in the code. No comments = (-30) points *per Part of the assignment*

Do not submit zip files.

Late penalties of 10 % per day are in effect for this assignment.