CSCI 230 Data Structures and Algorithms
Problem Set 3 - Maps, Hash Tables, and Sets
Student Name

Assignment

This assignment is based on material from the course primary textbook, "Data Structures and Algorithms in Java" by Michael Goodrich, chapters:

- Chapter 10 Maps, Hash Tables, and Skip Lists

Problem 1. The use of null values in a map is problematic, as there is then no way to differentiate whether a null value returned by the call `get(k)` represents the legitimate value of an `entry(k,null)`, or designates that key `k` was not found. The `java.util.Map` interface includes method `boolean containsKey(k)`, that resolves any such ambiguity. Implement such a method for the `UnsortedTableMap` class.

Code: `UnsortedTableMap.java`

```
1  public class UnsortedTableMap<K,V> extends AbstractMap<K,V> {
2      ...
3      boolean containsKey(K key) {
4          // Implement this method
5      }
6      ...
7  }
```

Problem 2. What is the worst-case time for putting n entries in an initially empty hash table, with collisions resolved by chaining? What is the best case?

Problem 3. Describe how a sorted list implemented as a doubly linked list could be used to implement the sorted map ADT.

Problem 4. What abstraction would you use to manage a database of friends' birthdays in order to support efficient queries such as "find all friends whose birthday is today" and "find the friend who will be the next to celebrate a birthday"?

Submission Guidelines

Modify this LaTeX document by inserting your solutions into the `solution` environments above. Submit this document along with any source code files `[*.java]` and archives `[*.jar]` to the course LMS. Finally, comment out the `\input{TexFiles/SubmissionGuidelines.tex}` line in `main.tex` to hide this section.