Computer Science 245
Test 2. Data structures; patterns.

Java features.

- Generics – know how to write a basic generic class (`class C <T extends A>`)

- Private and inner classes – know how to write an anonymous inner class, as for an iterator or action listener.

Data structures.

- Know stacks and queues, their operations, implementations (linked and array), and applications.

- Know binary trees, their linked and array-based implementations, and basic traversal strategies

- Review how one would define a tree class to represent expressions in a grammar, and how to build a tree by parsing an expression in such a language.

- Understand the purpose of hashing and how a hash table works.

Design patterns. Be able to identify an appropriate pattern to use to solve a given problem. Be able to draw a UML diagram for a solution using one of these patterns.

- Factory method

- Singleton

- Template method

- Iterator

- Strategy

- State

- Adaptor

- Decorator

Practice problem.

Write a class that implements a hash map, `YourHashMap`, generic in both the key and value types. Write it so that it works in the following program.

```
import java.util.Iterator;
import java.util.Scanner;

public class MapTest {
    public static void main(String[] args) {

        Scanner keyboard = new Scanner(System.in);

        YourHashMap<String,String> dictionary =
                new YourHashMap<String,String>(10);

        for (;;) {
            System.out.print("Please enter a word, blank to quit-> ");
            String word = keyboard.nextLine();
            if (word.equals("")) break;
            System.out.print("Please enter a definition for " + word + "-> ");
            String def = keyboard.nextLine();
            dictionary.put(word, def);
        }

        System.out.print("Please enter a word you would like to look up-> ");
        String word = keyboard.nextLine();

        if (dictionary.containsKey(word))
            System.out.println(dictionary.get(word));
        else
            System.out.println(word + " is not in the dictionary");

        for (Iterator<String> it = dictionary.keyIterator(); it.hasNext(); ) {
            String currentWord = it.next();
            System.out.println(currentWord + ": " + dictionary.get(currentWord));
        }
    }
}
```

Thus your class should implement the methods `put()`, `containsKey()`, `get()`, and `keyIterator()`. The constructor takes an `int` which indicates the size of array to use. Use the `hashCode()` method from the `Object` class as a hash function.