Abiola Gabriel Olofin

CS 150-02

Professor Falconi

02/09/2020

Lab 3 Report

Introduction

In this lab, we were asked to create a Cell class that allows us to add an integer value to the cell and link us to the next Cell object. We then had to create an IntegerList class that extend an Integer abstract data type and in this IntegerList class, we had to adds integers to Cell objects and has different methods. In other words, we are creating our own linked list

Approach

When I was designing the program, I only created one class that ran the entire program called ExperimentController and created the Cell class to make the fundamental linked list structure. I also created an Integerlist and an IntegerlistADT, which is an abstract class for the IntegerList class. I imported the java.util.\* in all 3 classes which imports all the libraries in Java to make it easier to call certain classes and methods of different classes. In the ExperimentController class, I created 2 methods to test the time that it takes each method in the IntegerList class to complete the task. I also created a main method that would run each of those methods to generate the answers needed. In the main method, I tested each of the methods through the use of while loops in order to test how different lengths of Cell objects are populated and how much time it takes depending on the method. In the Cell class, I created 3 methods that added the values to the Cell objects and I created a toString method that was recursive and printed out the values of each Cell object. In the Integerlist class, I created 4 methods to that added values to or created new Cell objects, returned values of cells as strings, and check to see if a cell was empty. They were all abstracted methods instantiated in the IntegerListADT but were implemented in the IntegerList class.

Methods

The methods I wrote in the Cell class are append (int x) which added an integer value to the current cell and created a new cell and toString() which printed out the value of each cell in the cell linked list. In the IntergerList class I wrote append() and toString() methods as well which were based on the append() and toString() methods in the Cell class and were declared in this class because they could not be declared in the abstract class. I also unit tested the Cell and IntegerList class to assert that every method worked exactly the way I wanted to and was the correct way a linked list should work.

Data

I collected data to graph by running multiple trials of the two methods to see how long it takes the JVM on average to run the methods and how much time different inputs and added elements took more time to complete.



A screenshot of a cell phone

Description automatically generated

Conclusion

In conclusion, this lab was used to make us understand another data structure – linked list. This data structures involves using nodes and linking the nodes together through the use of pointers that come from one node pointing to another node. This will allow us to understand the fundamentals of linked lists and doubly linked lists. Also, we learned about how abstract classes work and how other classes can implement methods in the abstract classes.

References

Collections API

Random API

<file:///Users/abiolaolofin/Desktop/CS150Labs/Lab3/doc/Cell.html>

<file:///Users/abiolaolofin/Desktop/CS150Labs/Lab3/doc/IntegerList.html>

<file:///Users/abiolaolofin/Desktop/CS150Labs/Lab3/doc/IntegerListADT.html>