CSC2001F Assignment 1 Report

Brandon Gower-Winter

Problem:

The problem asked of this Assignment is to test the efficiency of two data structures (Array and Binary Search Tree(BST)) and compare them using the data file provided.

Application Design:

I designed the application to be as modular as possible. I created my own generic BST that uses well-known algorithms to carry out its functions. It also makes use of the comparable interface to determine what is bigger and what is smaller thus a Type T must implement the Comparable interface in order to utilize the BST.

I created a person class to act as a data model to allow me to store the name, telephone number and address of all the people in the data file. The interface implements the comparable interface to allow for use in the BST mentioned above.

SearchItLinear is one of the application classes created. It acts in such a way as to add data from a data file into an array list and then read in queries from a query file to perform searches. The queries are searched for in the array list and an appropriate result is printed to the output stream. (See SearchItLinear Output)

PrintIt is another application class created with the purpose of creating a BST data structure and adding in the entries from the data file into the BST. Once that is complete it then will call for the BST to print out the data in order. This is when the BST uses a in order traversal algorithm to print out the values stored within the BST in an order that is defined by the Type utilizing the BST's compare to function. In this case, the person class is use as the Type and the result is that the data entries are printed out in alphabetical order sorted by the name. (See Output PrintIT)

SearchIt is the final application class created and its purpose is to store the data from the data file in a BST and then read in queries from a query file and return a suitable result as to whether the query exists in the data file. The query is read and the find function is called on the BST and an appropriate result is returned. The person object if the value is found and null if no one found.

PrintIt Output:

```
Abbott Alec    489-848-7299   03707 Botsford Fork, Lima
Abbott Alexandria      318.679.5603 x712      44812 Wilderman Mountain,
Vallejo
Abbott Alia    507.340.1186   76400 Barton Fields #044, Cerritos
Abbott Brando  602.992.4016   02519 Zackery Village, San Mateo
Abbott Elwyn   788.603.8604   88126 Bruen Common, Beverly Hills
Abbott Hosea   1-035-079-0176 x61480  51832 Bayer Pass, Simi Valley
Abbott Ima     823.283.2198 x7192     87191 Suite Z, Selma
Abbott Josh    822.752.1004   27010 Sanford Center, Stanton
Abbott Leann   516-835-0116   17296 Elta Crossroad #362, Newport Beach
Abbott Meda    1-117-789-3061 18565 Suite B, Fountain Valley
Abbott Murray  1-654-279-2374 22345 Runte Garden, Steubenville
Abbott Novella 297-763-2822   32763 Langosh Route, San Diego
Abbott Rahsaan (681)856-6604 x642     90282 Haag Keys, Garden Grove
Abbott Sadye   (961)238-9093  52000 Marques Loaf #288, Placentia
Abbott Santina 1-515-459-1556 78469 Renner Mill, Agoura Hills
```

```
Abernathy Amparo       1-052-394-1236 x29668  96179 Feil Tunnel #352, Canton
Abernathy Austyn       1-486-893-0367  98827 Gerlach Pike Apt. 743, Apple
Valley
Abernathy Catalina     1-331-934-0147  14576 Harber Knolls, Riverside
Abernathy Chadd(552)753-8320 x85031   23694 Pier F, Tempe
Abernathy Cicero       (637)882-6835 x72457   36296 Batz Walk, San Francisco
```

SearchIt and SearchItLinear Output:

```
Mayert Cathy   791-772-8120 x42168    90125 Raven Circle #864, Downey
Gower-Winter Brandon was not found.
Hickle Leone   018-594-2935 x716      17386 Stephanie Parks, Palm Springs
West Ramon     1-702-852-5634  50773 Schinner Extensions, Zanesville
Mikazuki Augus was not found.
Setsuna F. Seiei was not found.
Langosh Matt   (682)669-6865 x500     73754 Leffler Squares, Atwater
Ondricka Luz   (522)447-5098 x1929    68014 Jermain Street, Springfield
Labadie Leanna (896)176-7008  37773 Durgan Parkways Suite 558, San Dimas
Lockon Stratos was not found.
Jacobs Wallace (174)976-6745 x0539    06395 Cormier Crest Suite 404, West
Memphis
Orga Itsuka was not found.
Medhurst Sydnie was not found.
Haag Abraham   (933)453-8588 x7220    04266 Missouri Junction, Burlingame
Ryan Alicia    950-938-7050 x27619    76980 Side, Auburn
Alleluyah Haptism was not found.
Daugherty Elijah       (549)540-0126 x715     26255 Marvin Way #268, Decatur
Abernathy Houston      757.248.9579 x9418     24902 O'Conner Creek, Homer
Lehner Alberta (460)301-1274 x351     67100 Schumm Pines, Barrow
Fahey Lane     1-220-139-2838 x649    04923 Flatley Island Suite 476, San
Clemente
```

Experimental Design:

The designed my experiment to be take on three different categories of the two-data structure. 1. What if the queries were in the first 20 entries. 2. What if the queries were the last 20 entries. 3. What if the entries were randomly chosen. This to make sure both extremes and a real life example are all tested.

The data collected from the testing is the time taken for the application to run. Ie: How long it taken for SearchIt to run.

Both SearchIt and SearchItLinear go through 20 queires and were tested have twenty results obtained.

Results:

	SearchIt	SearchItLinear
Time:	0.68 0.02 0.70 0.02 0.68 0.00 0.65 0.03 0.64 0.02 0.67 0.02 0.65 0.02 0.67 0.01 0.68 0.00 0.71 0.02 0.70 0.00 0.67 0.02 0.64 0.02 0.66 0.00 0.65 0.03 0.67 0.03 0.70 0.00 0.64 0.02 0.65 0.02 0.62 0.03	0.66 0.01 0.67 0.01 0.64 0.02 0.62 0.03 0.67 0.02 0.65 0.02 0.64 0.02 0.67 0.02 0.63 0.01 0.64 0.02 0.64 0.02 0.62 0.02 0.69 0.01 0.66 0.02 0.64 0.02 0.63 0.02 0.64 0.02 0.66 0.00 0.66 0.01 0.65 0.04

Conclusion:

A BST data structure is only more efficient when querying for data at the end of a large data file otherwise both the BST and ArrayList perform as efficiently for small data values.

GIT Log:

- **Initial Commit**
- **Added libraries**
- **Added assigment 1**
- **Finished programming all classes for the assignment**
- **Javadoc for person setup and makefile now compiles all java classes**
- **Added JavaDocs for BinarySearchTree and BinarySearchTree**
- **Javadocs for all classes are complete**
- **Added Unit Testing for Binary Search Tree**
- **Completed Overall Testing**

Junit stats:

Runner class returned true for no error.