PROGRAM 4 / CSC1310

BINARY SEARCH TREES

I Love Krabby Patties' Ultimate Love Song Music Video: https://www.youtube.com/watch?v=STufG-6QMWI

IMPORTANT DATES

Assignment Date: Thursday, November 8, 2018
Due Date: Thursday, December 6, 2018

PROGRAM ASSIGNMENT DESCRIPTION

You are writing a program for Mr. Krabs at the Krusty Krab in beautiful Bikini Bottom! The program will allow the user to choose from one of the following options:

1. Enter customer data (customer name & how many Krabby Patties they ate)
2. Remove a customer's data
3. Get statistics on Krabby Patties (customer who ate the least, customer who ate the most, and the total number of krabby patties eaten)
4. Search for a customer's data
5. End the program

The customer's data should be stored in a binary search tree. You will be writing your own BinaryTree class and instead of each tree node containing one value, your tree nodes will contain TWO values – a string containing the name of a customer & an integer containing the number of Krabby Patties the customer ate.

FILES THAT SHOULD BE INCLUDED IN YOUR SUBMISSION

- **BinaryTree.h**
- **BinaryTree.cpp**
- **KrustyKrab.cpp (driver)**
- **Makefile**
- **runProgram.bat**
- **TEST_CASE.txt**

PROGRAM SPECIFICATIONS (DIRECTIONS ON HOW TO WRITE THE PROGRAM)

BINARY TREE CLASS

The BinaryTree.h file will contain your class specification for class named BinaryTree, which is an implementation of a Binary Search Tree. The BinaryTree.cpp file will contain your class implementation for the BinaryTree class.

ATTRIBUTES

Instead of each tree node containing one value, your tree nodes will contain TWO values (in addition to the left & right pointers):

1. A string containing the name of a customer
2. An integer containing the number of Krabby Patties the customer ate

You will also need a pointer to the root of the tree.

CONSTRUCTOR & DESTRUCTOR

You need a **constructor** which will set the root pointer to NULL.

You will also need a **destructor**, which will call a private function named destroySubTree. The **destroySubTree** function should delete all nodes in the tree since they were dynamically allocated.

BINARY TREE OPERATIONS:

1. **Insert a new node** (requires two functions)
    a. The public function (**insertNode**) should accept a new customer's name and the number of Krabby Patties they ate. This function should create a new TreeNode and fill it with these new values and then call the private function (insert).
    b. The **insert** function (recursive function) inserts the node in the correct position.

2. **Delete a node** (requires three functions)
    a. The public function (**remove**) accepts a customer's name in order to identify the node that should be removed. This function prints, "You are trying to delete [name]." This function will call the private deleteNode function (a recursive function).
    b. The **deleteNode** function tries to find the node that needs deleted. If deleteNode finds the node, it calls the private makeDeletion function. If deleteNode does not find the node, it should print "I'm sorry. That customer can't be found in the Krusty Krab."
    c. The **makeDeletion** function removes the node and reattaches the branches of the tree below the node.

3. **Display names of customers** (requires two functions)
    a. The public function (**displayInOrder**) will call the private overloaded function also called displayInOrder.
    b. The private **displayInOrder** accepts a TreeNode as an argument and it is a recursive function which traverses the nodes using the inorder method and prints the customer's name from each node.
4. **Search for a customer**
    a. The **searchNode** function accepts the customer's name as an argument and then it will look at each node (traverse the tree) to find the node with that name. If the name is found, the function returns the number of Krabby Patties that customer ate. If not, then it will return -1 instead.

5. **Find the lowest** number of Krabby Patties eaten by any customer (requires two functions)
    a. The public function (**getLeastNumPatties**) accepts two arguments – a string that will eventually hold the customer's name that has the least number of eaten Krabby Patties and an integer that will eventually hold the customer's # of eaten Krabby Patties. This function sets the integer parameter (leastPatties) to the root node's # of patties and then calls the private function (getLeast).
    b. The **getLeast** function is a recursive function that traverses the nodes like the displayInOrder function except instead of printing out the customer's name, it will check to see if the current node has a smaller number then the one saved in the leastPatties variable. If so, then save this node's customer name & number of patties as the new least.

6. **Find the highest** number of Krabby Patties eaten by any customer (requires two functions)
    a. Follow the instructions for finding the lowest except find the highest instead

7. **Find the total** number of Krabby Patties eaten by ALL customers (requires two functions)
    a. The public function (getTotalNumPatties) accepts no arguments. It reates a variable to hold the total and then calls the private getTotal function, sending the root node & the total variable (send total by reference) to the function. Then, this function will return the total as an integer.
    b. The getTotal function is a recursive function that traverses the tree like the displayInOrder function except instead of printing out the customer's name, it will get a running total of Krabby Patties eaten from each node.

KRUSTYKRAB.CPP (DRIVER)

Most of this file is provided for you. However, I have placed comments in the file where you will need to add code. The main function is the only required function in this file. You will need to add a variable definition that creates a BinaryTree object and then you will be adding all the function calls to the BinaryTree member functions.

READABILITY OF OUTPUT & CODE DOCUMENTATION

- Make sure that your output looks similar to my sample output (below). When I run your program, it shouldn't make me want to scream. It should be extremely readable and user-friendly.
- **For this program, don't worry about comments except put your NAME at the top of all files that you modify!!!**

WHAT TO TURN IN

Zip ALL the files required to compile & run the program, in a single zipped file named whatever you want. Then, upload this zip file to the assignment folder in ilearn. **I will remove one point if you turn in unzipped files. Programs that do not include all the files listed in the "FILES" section above will not be graded.**

Image credit: https://www.hungryforever.com/recipe/spongebob-squarepants-krabby-patty-recipe/

SAMPLE OUTPUT

Select C:\Windows\System32\cmd.exe

```
C:\Users\acrockett\Desktop\CSC1310 Spring 2018\PROGRAMS\PROGRAM 8\solution>Krus
tyKrab



_  _  ____   _  _ (  ___ _____  _  _ (    _  _  ____     _   ___
) |) / /  _ \ ) () (  ( _( )__ __() () (   ) |) / /  _ \   )_\  \  _)
| ( (  )  ' / | \/ | _) \    | |  '.  /    | ( (  )  ' /  /( )\ |  (
)_|)_\ |_()_\ )____( )____)  )_(   /_(     )_|)_\ |_()_\ )_/ \_(/__o)

Welcome to the Krusty Krab!
Choose one of the following options:
1.  Enter customer data.
2.  Remove a customer's data.
3.  Get statistics on Krabby Patties.
4.  Search for a customer's data.
5.  End Program.
ENTER 1-5:  1
Enter a customer's name or -1 to quit entering data.
NAME:  SpongeBob Squarepants
NUMBER KRABBY PATTIES EATEN:  52
Enter a customer's name or -1 to quit entering data.
NAME:  Eugene H. Krabs
NUMBER KRABBY PATTIES EATEN:  2
Enter a customer's name or -1 to quit entering data.
NAME:  Patrick Star
NUMBER KRABBY PATTIES EATEN:  182
Enter a customer's name or -1 to quit entering data.
NAME:  Squidward Tentacles
NUMBER KRABBY PATTIES EATEN:  18
```

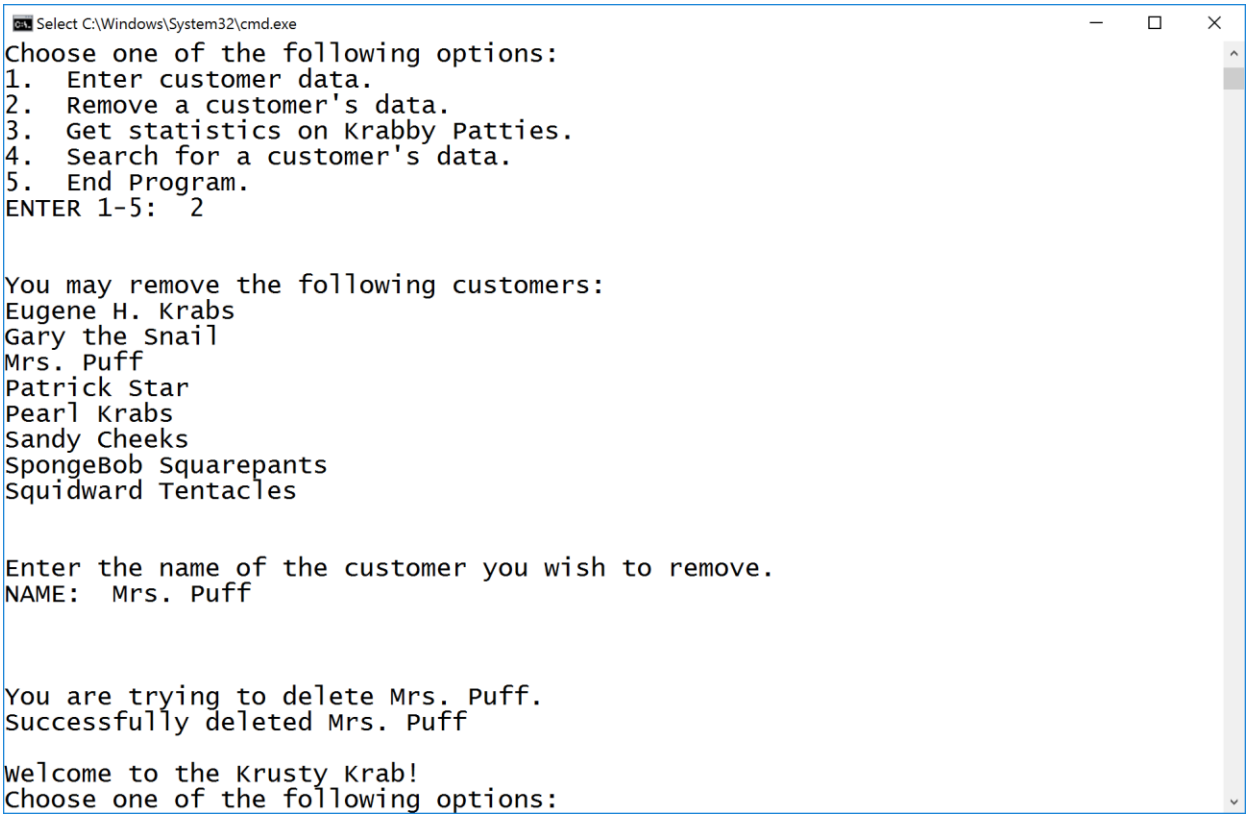
Select C:\Windows\System32\cmd.exe

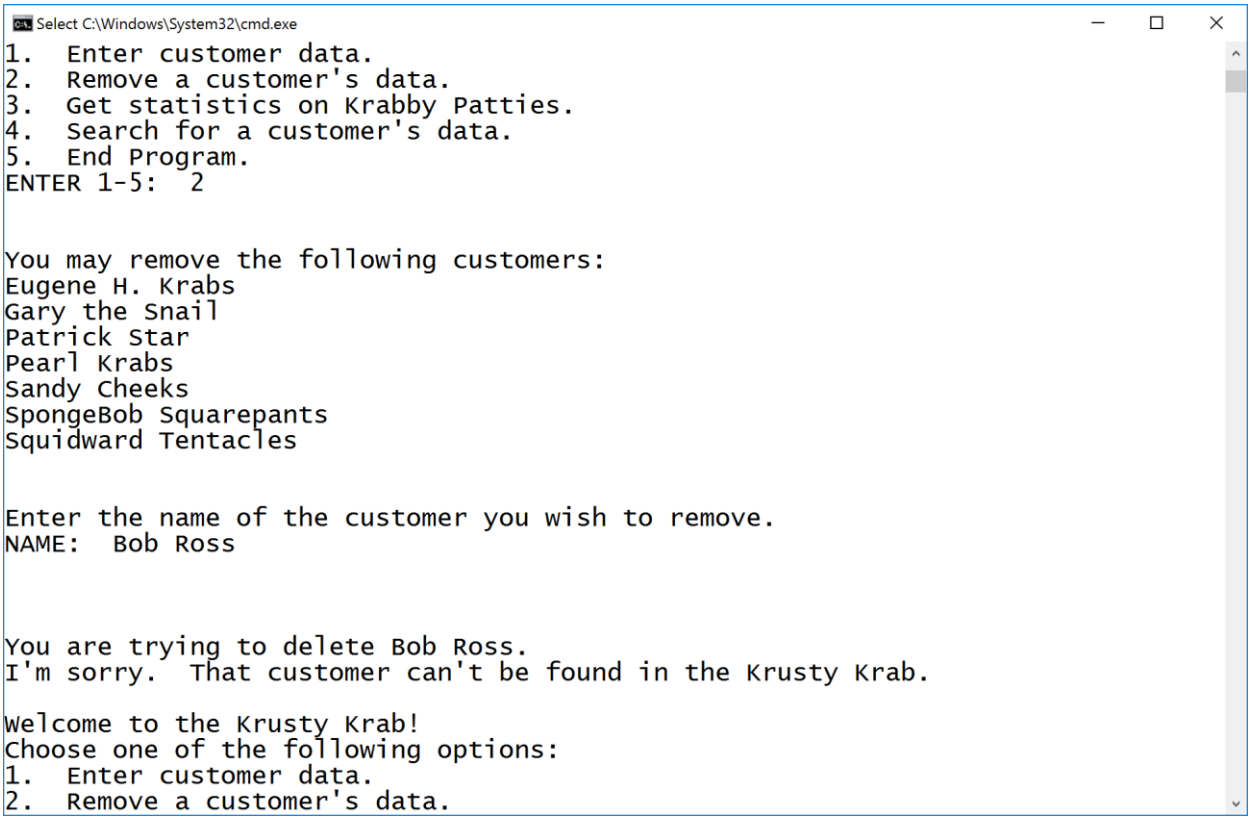
```
Enter a customer's name or -1 to quit entering data.
NAME:  Sandy Cheeks
NUMBER KRABBY PATTIES EATEN:  12
Enter a customer's name or -1 to quit entering data.
NAME:  Mrs. Puff
NUMBER KRABBY PATTIES EATEN:  34
Enter a customer's name or -1 to quit entering data.
NAME:  Pearl Krabs
NUMBER KRABBY PATTIES EATEN:  3
Enter a customer's name or -1 to quit entering data.
NAME:  Gary the Snail
NUMBER KRABBY PATTIES EATEN:  1
Enter a customer's name or -1 to quit entering data.
NAME:  -1

Welcome to the Krusty Krab!
Choose one of the following options:
1.  Enter customer data.
2.  Remove a customer's data.
3.  Get statistics on Krabby Patties.
4.  Search for a customer's data.
5.  End Program.
ENTER 1-5:  3


LEAST NUMBER OF KRABBY PATTIES EATEN:  Gary the Snail, 1 Krabby Patties
LARGEST NUMBER OF KRABBY PATTIES EATEN:  Patrick Star, 182 Krabby Patties
TOTAL NUMBER OF KRABBY PATTIES EATEN:  304

Welcome to the Krusty Krab!
```


Select C:\Windows\System32\cmd.exe
Choose one of the following options:
1. Enter customer data.
2. Remove a customer's data.
3. Get statistics on Krabby Patties.
4. Search for a customer's data.
5. End Program.
ENTER 1-5: 2
You may remove the following customers:
Eugene H. Krabs
Gary the Snail
Mrs. Puff
Patrick Star
Pearl Krabs
Sandy Cheeks
SpongeBob Squarepants
Squidward Tentacles
Enter the name of the customer you wish to remove.
NAME: Mrs. Puff
You are trying to delete Mrs. Puff.
Successfully deleted Mrs. Puff
Welcome to the Krusty Krab!
Choose one of the following options:


Select C:\Windows\System32\cmd.exe
1. Enter customer data.
2. Remove a customer's data.
3. Get statistics on Krabby Patties.
4. Search for a customer's data.
5. End Program.
ENTER 1-5: 2
You may remove the following customers:
Eugene H. Krabs
Gary the Snail
Patrick Star
Pearl Krabs
Sandy Cheeks
SpongeBob Squarepants
Squidward Tentacles
Enter the name of the customer you wish to remove.
NAME: Bob Ross
You are trying to delete Bob Ross.
I'm sorry. That customer can't be found in the Krusty Krab.
Welcome to the Krusty Krab!
Choose one of the following options:
1. Enter customer data.
2. Remove a customer's data.

```
Select C:\Windows\System32\cmd.exe
3.  Get statistics on Krabby Patties.
4.  Search for a customer's data.
5.  End Program.
ENTER 1-5:  4


Which customer are you looking for?
Eugene H. Krabs
Gary the Snail
Patrick Star
Pearl Krabs
Sandy Cheeks
SpongeBob Squarepants
Squidward Tentacles
Enter the name of the customer.
NAME:  SpongeBob Squarepants

SpongeBob Squarepants ate 52 Krabby Patties

Welcome to the Krusty Krab!
Choose one of the following options:
1.  Enter customer data.
2.  Remove a customer's data.
3.  Get statistics on Krabby Patties.
4.  Search for a customer's data.
5.  End Program.
ENTER 1-5:  4


Which customer are you looking for?
```

```
Select C:\Windows\System32\cmd.exe
Eugene H. Krabs
Gary the Snail
Patrick Star
Pearl Krabs
Sandy Cheeks
SpongeBob Squarepants
Squidward Tentacles
Enter the name of the customer.
NAME:  Bob Ross

Bob Ross is not a customer of the Krusty Krab.
Welcome to the Krusty Krab!
Choose one of the following options:
1.  Enter customer data.
2.  Remove a customer's data.
3.  Get statistics on Krabby Patties.
4.  Search for a customer's data.
5.  End Program.
ENTER 1-5:  3


LEAST NUMBER OF KRABBY PATTIES EATEN:  Gary the Snail, 1 Krabby Patties
LARGEST NUMBER OF KRABBY PATTIES EATEN:  Patrick Star, 182 Krabby Patties
TOTAL NUMBER OF KRABBY PATTIES EATEN:  270

Welcome to the Krusty Krab!
Choose one of the following options:
1.  Enter customer data.
2.  Remove a customer's data.
3.  Get statistics on Krabby Patties.
```

```
Select C:\Windows\System32\cmd.exe
4.  Search for a customer's data.
5.  End Program.
ENTER 1-5:  5


Goodbye!


C:\Users\acrockett\Desktop\CSC1310 Spring 2018\PROGRAMS\PROGRAM 8\solution>
```