GIT Department of Computer Engineering
CSE 222/505 - Spring 2020
Homework 5_Q1/2/3/4

BARAN HASAN BOZDUMAN
171044036

Problem Approach for Q1

My filesystem is creates a filesystem hieararchy. Firstly It needs to create a root file and after that if user wants to create a file there is a field in my file node which is is directory, my add file method makes it false and also I keep Linked list to keep subdirectories so I got a tree structure But there is a problem with my search method I do not nknow why it didnt worked and there are also some spesifications such as the user can not add same file or directory names more than once that is for the remove method because since user gives only path as you determine in hw.pdf the program can not decide to delete which file.So I decide to unique names for files and folders which are in the same directory.For printing files I print it like filesystem hiearch you can look up the doc file for more details.

Q1 -Test Case Table

Test number	Number of Output Image	Test Senario	Expected Result	Actual Result	Pass/Fail/ Not Executed
1	1	Create a file system and add files and folders	There is not any error	There was not any error	PASS
2	2	Add some files and folders which exsist in fdirectory	Giving error	It works as expected	PASS
3	3	Try to add a file in a file	Giving error	Itworks as expected	PASS
4	4	Creating random files and folders to create a hierarchy	**Showing the hierarchy**	It works as expected	PASS
5	5	Removing a folder which has not sub directory	Warnn to removes a folder and remove it	It works as expected	PASS
6	5	Removing File in root subdirectory	Warn to remove a file and remove it	It works as expected	PASS
7	6	Remove files which is in more depth and remove it	Warn it and remove it	Itworks as expected	PASS
8	6	Remove folder which is in more depth	Warn and remove it	Itworks as expected	PASS
9	7	Remove a file which is not in the filesystem	Gives error	Itworks as expected	PASS
10	8	Removing root files and print the file hierarchy	Deletes all files and prints	Itworks as expected	PASS
11	9	Search method	Prints the all files and directory types	Not working [roperly	Failed

Problem Approach for Q2

The purpose of this part take prefix and postfix expression from user and construct a expression treesince the read binary tree method almost do the same thing for expression tree. I did not change too much there would be to base cases for the my read method one of it for the end of the string that means it stops to put tree the strings other one if the string which is got is a number thatmeans it cannot have children so if it takes the number it put in the tree but it assigns its childs to null because it cannot have more children so thats the point of part and for evaluation it goes until the leaf of tree and it implies the operation until root for detail explain you can look up the java doc.

Q2 -Test Case Table

Test number	Number of Output Image	Test Senario	Test data	Expected Result	Actual Result	Pass/Fail/ Not Executed
1	1	Calling constructor for prefix	+ + 10 * 5 15 20	Add elements to tree	There was not any error	PASS
2	2	Call to string method	+ + 10 * 5 15 20	+20 + * 15 5 10	It works as expected	PASS
3	3	Call toString2 method	+ + 10 * 5 15 20	20 15 5 * 10 + +	Itworks as expected	PASS
4	4	Call evaluate method	+ + 10 * 5 15 20	**105**	It works as expected	PASS
5	1	Calling constructor for postfix	20 15 5 * 10 + +	Add elements to tree	There was no error	PASS
6	5	Call to string method	20 15 5 * 10 + +	+ + 10 * 5 15 20	It works as expected	PASS
7	6	Call toString2 method	20 15 5 * 10 + +	10 5 15 * + 20 +	Itworks as expected	PASS
8	7	Call evaluate method	20 15 5 * 10 + +	105	Itworks as expected	PASS
9	8	Call constructor for an invalid expression	10 5 15 * + 20 +	Gives a warning	Itworks as expected	PASS
10	9	Call constructor for different postfix and evaluate	10 2 8 * + 3 -	23	Itworks as expected	PASS
11	10	Call constructor for different prefix and evaluate	+ 9 * 2 6	21	Itworks as expected	PASS

Problem Approach for Q3

To create min heap which is uses ageData. Firstly I create an interface name is BinarySearch and I imlement them bst methods in to SearchTree class and my AgeSearch tree class extends from searchtree class, actually I use agesearchtree class to manuplate heap operations for method details you can check the java doc alsoThere is two problem with my program the younger than method only checks the one side it behaves like binary search I could not fix it for the heap and older than method since I firstly find the smallest element after the given age it does not check the other side so it counts only one side so it does not compensate the result probabl if I keep two smallest elements which are bigger than the given age it would be solve but since you do not allowed us to traverse all heap probably I would not get full credit again

Q3 -Test Case Table

Test number	Number of Output Image	Test Senario	Test data	Expected Result	Actual Result	Pass/Fail/ Not Executed
1	1	Adding elements which are different each other	10,5,70,15	Adding the elements in order	Itworks as expected	PASS
2	2	Adding an element which is on the heap	Adding 70,70,5,10	It carrise the 70 top of list	It works as expected	PASS
3	3	Removing an element which is in heap	Remove 5	It carries the 5 below 10	Itworks as expected	PASS
4	4	Removing an element which is only have one element in the heap	Remove 5	It removes 5 from list	It works as expected	PASS
5	5	Removing an element which is not in the heap	Remove 73	Gives a warning	Itworks as expected	PASS
6	6	Call add method and fill the heap againg	Adding 5,5,5,15,23,45,11	Arranges the apropriate hieararchy	It works as expected	PASS
7	7	Call Find method for top of the heap	Find 70	Retrieves the age and number of people	Itworks as expected	PASS
8	8	Call find for the any	Find 10	Retrieves	It works as	PASS

		middle of the heap		age and number of people	expected	
9	9	Call find for any last level of element heap	Find 11	Retrieves age and number of people	Itworks as expected	PASS
10	10	Call find for element which is not in the heap	Find 99	Gives a warn message	It works as expected	PASS
11	11	Call youngerThan method with elements which is not in the heap	Younger than 5	Prints 0	Itworks as expected	PASS
12	-	Younger than methot with an elementh	Youngerthan 45	Prints 10	Prints 5	Failed
13	-	Youngerthan method with in ages between two age	Youngerthan 13	Prints 6	IPrints 5	Failed
14	12	Call olderThan method with elements which is not in the heap	Older than 70	Prints 0	It works as expected	PASS
15	15	older than methot with an elementh	Olderthan 45	Prints 3	Itworks as expected	PASS
16	-	olderthan method with in ages between two age	Olderthan 44	Prints 4	Prints 3	Failed

Problem Approach for Q4

The purpose of this part the solve maxHeap problem by using arraylist which is the one of the shown of max heap and the key value is number of people at this time so the age max number of people must be beginning of the listand if user adds new age which is inte heap after add operation we must check the list is it in order or not other wise we swap them according to parent and child relationship and after add element I check from added index and if it is not in ordered I swap the with appropriate order and I use for some algrithm from book.

Q4 -Test Case Table

Test number	Number of Output Image	Test Senario	Test data	Expected Result	Actual Result	Pass/Fail/ Not Executed
1	1	Adding elements which are different each other	10,5,70,15	Adding the elements in order	Itworks as expected	PASS
2	2	Adding an element which is on the heap	Adding 70,70,5,10	It carrise the 70 top of list	It works as expected	PASS
3	3	Removing an element which is in heap	Remove 5	It carries the 5 below 10	Itworks as expected	PASS
4	4	Removing an element which is only have one element in the heap	Remove 5	It removes 5 from list	It works as expected	PASS
5	5	Removing an element which is not in the heap	Remove 73	Gives a warning	Itworks as expected	PASS
6	6	Call add method and fill the heap aging	Adding 5,5,5,15,23,45,11	Arranges the aprropriate hieararchy	It works as expected	PASS
7	7	Call Find method for top of the heap	Find 70	Retrieves the age and number of people	Itworks as expected	PASS
8	8	Call find for the any middle of the heap	Find 10	Retrieves age and number of people	It works as expected	PASS

9	9	Call find for any last level of element heap	Find 11	Retrieves age and number of people	Itworks as expected	PASS
10	10	Call find for element which is not in the heap	Find 99	Gives a warn message	It works as expected	PASS
11	11	Call youngerThan method with elements which is not in the heap	Younger than 5	Prints 0	Itworks as expected	PASS
12	12	Younger than methot with an elementh	Youngerthan 45	Prints 10	It works as expected	PASS
13	13	Youngerthan method with in ages between two age	Youngerthan 13	Prints 6	Itworks as expected	PASS
14	14	Call olderThan method with elements which is not in the heap	Older than 70	Prints 0	It works as expected	PASS
15	15	older than methot with an elementh	Olderthan 45	Prints 3	Itworks as expected	PASS
16	16	olderthan method with in ages between two age	Olderthan 44	Prints 4	It works as expected	PASS

PS:

--->For images of the testcases you can look up in report directory witth related runningResults directory.

--->For uml classdiagrams you can look up the report directories.

--->For the java doc files you can look up the related question directory.