

CS214: Data Structures

Assignment-4: BST/AVL/Heap Trees



Instructions:

About the Assignment

1. All the team members must understand all the assignment problems.
2. All the code must be in C++.
3. Any cheating in any part of the assignment is the responsibility of the whole team, and all of the team members will be punished.
5. The solution should compile, run without run-time errors, and handle all the cases.

Submission Rules

Assignment is submitted in **teams of 3** from **any group**.

1. You will upload a zipped folder that contains your code (**Don't include any .exe files in your submission**).
3. Assignment submission is on Google Classroom (**No submission through mail**).
4. Follow this convention for naming your folder: ID1_ID2_ID3_A#_G# (i.e 20200111_20200222_20200333_A2_G5_G6)
6. Deadline of the Assignment: **12 May, 2023, at 11:59 p.m.**

Failure to follow any of the above rules will result in your submission being discarded and your team being considered to have not submitted.

Task Description:

Assume having an application which stores the data of students in binary search tree and AVL trees, where each student has an id, a name, department and a GPA. In case of BST and AVL the key of the tree is the id, there are four functions as follows:

1. Add a student (write the id "from 0 to 100", name, GPA, and department)
2. Remove a student using id
3. Search for student using id (if found print the student information)
4. Print all and Department Report (all the information of students are printed sorted by id and count students per department)

The students also can be inserted in min or max heap or displayed sorted by gpa (the key is gpa) where a report of department appears, the number of students per department.

CS214: Data Structures

Assignment-4: BST/AVL/Heap Trees



<<Assignment Input/Output>>

Input File (Part 1)

```
10
1
Mohamed Ali
3.4
CS
2
Mona Samir
3.2
IT
3
Ola Maher
1.2
CS
4
Magy Madgy
2.3
DS
5
Omnia Osama
3.6
IS
```

Input File (Part 2)

```
6
Ahmed Omar
3.9
CS
7
Mai Adel
3.1
IS
8
Mohamed Saleh
2.4
CS
9
Hany Mohsen
1.8
DS
10
Mohanad Bahaa
2.9
IT
```

In the input File, first line is number of students then four lines per students, i.e. id, name, GPA and department. Store the objects from the file in four trees **then, you will show the following menu:**

((First Menu – Main menu))

Choose Data Structure:

1. BST
2. AVL
3. Min Heap
4. Max Heap
5. Exit Program

CS214: Data Structures

Assignment-4: BST/AVL/Heap Trees



((Second Menu – choice 1 BST))

Choose one of the following options:

1. Add student
2. Remove student
3. Search student
4. Print All (sorted by id)
5. Return to main menu

Example:

Enter number of option: 1

id: 50
Name: tamer said
GPA: 3.5
Department: CS

The student is added.

Enter number of option: 2

Id: 5
Student is found.
[5, Omnia Osama, 3.6, IS]
Student is deleted.

Enter number of option: 2

Id: 90
Student is not found.

Enter number of option: 3

id: 6
Student is found.
[6, Ahmed Omar, 3.9, CS]

Enter number of option: 4

Print 10 Students.
[1, Mohamed Ali, 3.4, CS]
..
..
[50, tamer said, 3.5, CS]

Students per Departments:

CS 5 Students
IT 2 Students
DS 2 Students
IS 2 Students

((Second Menu – choice 2 AVL))

Same as choice 1 but implemented using AVL.

((Second Menu – choice 3 Min Heap))

Choose one of the following options:

1. Add student
2. Print All (sorted by gpa)

CS214: Data Structures

Assignment-4: BST/AVL/Heap Trees



Example:

Enter number of option: 1

id: 11

Name: Hana Sobhy

GPA: 3.2

Department: IT

The student is added.

Enter number of option: 4

Print 11 Students.

[3, Ola Maher, 1.2, CS]

..

[6, Ahmed Omar, 3.9, CS]

((Second Menu – choice 4 Max Heap))

Choose one of the following options:

1. Add student
2. Print All (sorted by gpa)

Example:

Enter number of option: 1

id: 13

Name: Soha Habib

GPA: 1.1

Department: IT

The student is added.

Enter number of option: 4

Print 11 Students.

[6, Ahmed Omar, 3.9, CS]

..

[13, Soha Habib, 1.1, IT]

N.B.

- Any operation in one tree doesn't affect other trees.
- Implement the four trees and any function required to make the options in the menu work for each tree.