

## CO322 Data structures and algorithms

### Lab4: Trees

#### **Introduction:**

As the name implies, binary search trees are used for searching items with a given collection of data. Furthermore, performing an *in-order traversal* on a binary search tree will yield a sorted list of the data in the tree.

In this lab class we will use binary trees for a slightly different purpose than searching – storing information about students and groups of students (i.e batches) and obtaining sorted lists based on the subjects and overall performance of a student.

For this lab class we will consider only the 5<sup>th</sup> semester computer engineering students who are following subjects: CO321, CO322, CO323, CO324, CO325 and EE386. You have finished all these courses (so no need to worry about labs any more) and your grade for each course will be calculated randomly. Each student will have a registration number and grades for the above courses. The courses will be kept in a binary search tree ordered by the course grade. So when you perform an in-order traversal for a student you should get the courses with his highest grade to the lowest.

You should also note that all the traversals we discussed in the class will visit each node once. Your next task is to use the above observation to implement a function that can compute the GAP (grade point average) for a given student. This computation is obvious: add the grade for each course and divide by the number of courses. We will use the typical grade points given in table below.

Students in the batch are again kept in another binary tree ordered by the GAP of each student. You may get the list of students from <http://www.ce.pdn.ac.lk/?cat=52>. You can use some method to extract the list from that webpage (include, though not recommended, manually reading and entering into a file). Your program should read the list of students from this file, for each student create a tree with randomly grades for the above list of subjects and insert into the binary tree. For a student the tree should be ordered by the subject grades and for the batch students should be ordered by the GPA. Finally, your program should display the student with the highest GPA first alone side his/her grades for the above courses ordered from the lowest grade to the highest.

To summarise, your program should create a tree of students and each student should have a tree of subjects. Then it should display the student with the highest GAP to the lowest and for each such student should display the grades from lowest to highest.

#### **Assessment**

The aim of this lab class is to check your understanding of data structures and object oriented programming concepts. If you look at the above problem carefully you will not many similarities

between the student's subject tree and the batch tree. If you are careful enough you will be able to leverage this and use concepts like inheritance and overloading to implement this efficiently.

**Submission**

You are required to implement the above code using Java and submit all the source files as a single tar file. You will be given marks for your understanding of data structure and object oriented programming concepts.

Submit your code via moodle on or before 28<sup>th</sup> March 2015. (No extensions and do not take the deadline that would appear in the moodle).