PROJECT #2 (Due November 29, 2015, Sunday, 24:00)

4-d BST

This project is a programming assignment in C which aims to build an algorithm that will store information related to students in **binary search tree** and makes necessary operations in this tree.

The entries in the input file will be similar:

(Student id, student name-surname, number of working-days of the student's summer internship)

120010018 Mehmet Oflu 10

120020011 Adem Baloğlu 60

120020021 Fatma Karatay 45

120020304 Berrin Yılmaz 28

120020056 Ezdin Yıldırım 40

…

Your program will insert these entries into a BST based on the following rule:

This will be 4-d tree. A 4-d tree is similar to an ordinary BST with only one exception is that:

Branching

at depths (0, 4, 8, 12,) is done with respect to name

at depths (1, 5, 9, 13,) is done with respect to surname

at depths (2, 6, 10, 14,) is done with respect to id

at depths (3, 7, 11, 15,) is done with respect to number of working-days of the student's summer internship.

120010018 Mehmet Oflu 10

/

120020011 **Adem** Baloğlu 60

\

120020021 Fatma **Karatay** 45

\

120020304 Berrin Yılmaz 28

\

120020056 Ezdin Yıldırım **40**

Traverse the tree in (optional: pre-order, post-order or in-order; **one traversal mode is enough; but also you can implement all of them for bonus**) and produce an output file in the following format:

Depth, number of the entries from the beginning of the root, id, name, surname, number of working-days of the student's summer internship

0 1 120010018 Mehmet Oflu 10

1 2 120020011 Adem Baloğlu 60

2 5 20020021 Fatma Karatay 45

3 9 120020304 Berrin Yılmaz 28

4 17 120020056 Ezdin Yıldırım 40

…

Also you are expected to select and display on the screen the records such that:

Names greater than **Deniz** and
Surname greater than **Balcı**
Id greater than than **120020304** and
Internship days is greater than **30**

These information (Deniz, Balcı, 120020304, 30) will be taken from the user.

In your demo, we will run your program by **using a different input file (in the same format).** We want to see if it is **working correctly or not**.

Of course, other questions based on your implementation and coding structure will be asked you during your demo. These questions will be those kinds of questions which could be answered by only the students who really implement his/her project.

The main goal of this project is to be familiar with BST. So, if you use other kinds of data structures instead of linked-lists then you will get zero, unfortunately.

In this project you are expected to develop an algorithm that is capable of finding a solution to the above problem and *implement this algorithm in ANSI C that runs under either UNIX or Windows*.

You are responsible for demonstrating your program to your TA Berna Altınel on the scheduled day that will be announced later.

NAMING OF YOUR FILE:

YOU MUST USE THIS NAMING STANDARD, OTHERWISE YOUR FILE WILL NOT BE GRADED.

ayse_yilmaz_120020056.c

CODE SUBMISSION:

You should use the following email address in order to submit your code:
datastr.mufe at gmail dot com

Your any submission after the project submission due date, will not taken into consideration.

You are required to exhibit an *individual effort* on this project. Any potential violation of this rule will lead everyone involved to **failing from the course** and necessary disciplinary actions will be taken.

Good luck!!!