

Lab 6: Tree and Binary Search Tree

6.1. Construct a family tree organized as follows: the root node holds a parents's name (names must be unique) as a key; each name is unique. Display this family tree with PreOrder (NLR), InOrder (LNR), and PostOrder (LRN) order on console screen or output file.

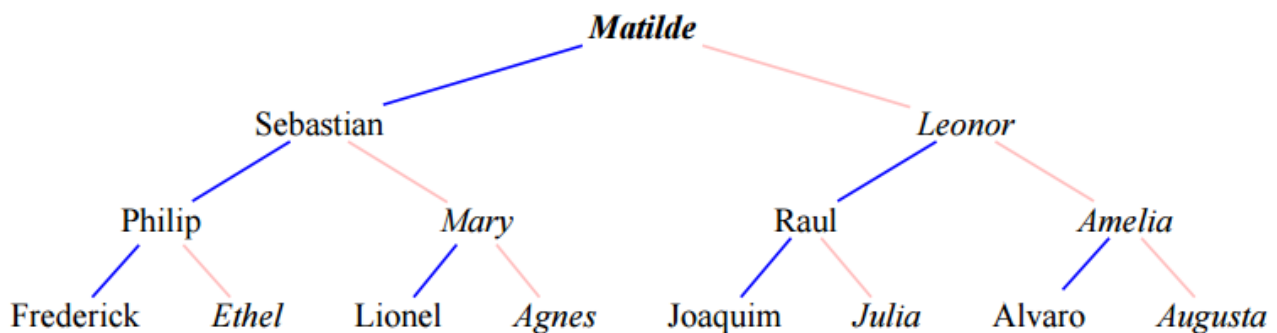
You are required read family data from input file with following structure:

- First row is number of relations
- Next rows are relations:

(< parentName > (< childName > < childName > ... < childName >))

The angle brackets ('<', '>') do not exist in the input, they just mark changeable amounts. All names (i.e. <childName>, < parentName >, <name1>, <name2 > are alphabetic strings with no spaces. '(', ')', ',' are delimiters.

For the example tree:



File input.txt:

7

(Matilde (Sebastian Leonor))

(Sebastian (Philip Mary))

(Philip (Frederick Ethel))

(Mary (Lionel Agnes))

(Leonor (Raul Amelia))

(Raul (Joaquim Julia))

(Amelia (Alvaro Augusta))

Notes: Parent can have many children (larger than 2).

6.2. You have to maintain information for school classes. For each of the students in a class the following information is kept: a unique code, student name (only first name with no space), birth date, status (undergrad, graduate). For keeping track of the students, the school secretary would use a computer program based on a binary search tree data structure. Write a program to help the secretary, by implementing following the following operations:

- Allow user create a new student info and insert this item with all its associated data to BST (Binary Search Tree).
- Find an student by his/hers unique code, and support updating of the student info if found.
- List all students in lexicographic order of their names.
- List all graduated students.
- Delete an student given by its code.
- Delete all graduates or save all students in file `student.data`.

If file `student.data` exists, your program must automatically load its contents before do above requirements.

Format in `student.data`:

- First row is number of students
- Next rows are student information: id is string, name is string, birth is string, status is boolean (0: undergrad, 1: graduated)

For example:

```
3
1512345
Lan
12/1/1997
0
1212346
Son
1/5/1995
1
```

Data Structures And Algorithms

1612347

Minh

23/9/1999

0

Notes: you must design which field is suitable key on BST.