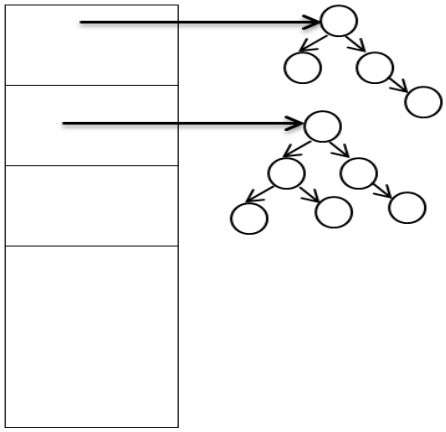
Project

Student ID at Qatar University is composed of year of admission and students number. In this project, we aim to implement a structure that improves operations of inserting and searching for a student. To enhance these operations when using tree data structure, we will build an array of trees where each tree holds only data of students admitted in a specific year. Figure 1 illustrates this concept.



Moreover, we want to speed up accessing a specific tree in the array, so we will use the year (in student's ID) as the index of the required element in the array, for example if the array is called admission_year , then a students who was enrolled in 1999, his record will be in the tree at *admission_year[1999]*. This will improve accessing any tree in the array instead of linear search from the beginning. However, we will have many locations in this array that are not used as Qatar University only was established at 1973, which means that all the elements from 0 to 1972 won't be used. Therefore, the year of admission should be mapped to a specific index as following

Index=year-1973

Implementation:

1- Each node in the tree will hold an object of type Student.
2- Student class has : id (int), name(String), address(String), GPA(double).
3- You should have one class that implements the structure above, call it *treeTable* which has the following methods:
   1- void insert(Student). This method inserts students in the appropriate tree based on his id.
   2- Student find (int). It receives an id, and returns student record with that id, or it returns null if it was not found.
   3- boolean remove(int). It receives student' id, removes his record if found and returns true, or returns false if it was not found.
   4- print (int): it receives an integer number represents the year, then prints students' data who were admitted in that year (use in order method).
   5- studentWithGPA(double): it returns array list of all students whose GPA is below the received parameter.

4- Your main application should have a menu with the following functionalities:
    1- Add new student : which prompts user to enter student's data, then insert it to the *treeTable.*
    2- Search for a student: which prompts user to enter an id then search in *treeTabel* and display the result.
    3- Delete a student: which prompts user to enter a student id, then remove his record, or display a message indicating this student was not found.
    4- Display students' data : which prompts user to enter a year, then display students' data of that year.
    5- Display students with less GPA: it prompts user to enter a number, then display students' data whose GPA is less than the input value.
    6- Save to file: which saves *treeTable* to a file.
    7- Load data: which loads data from a file to *treeTable*.

Notes:

1- This is a team-work project (2 students)
2- Deadline to submit your project is mid night **30/12/2015**.
3- Grading will be :
   8 points for correct implementation of treeTable class.
   7 points for correct implementation of main application.