Data Structures – Assignment 8

IDC, Spring 2020

Submission day: 2.7.2020– (you can use an extension and submit by 5.7.2020)

Honor code:
1. Do not copy the answers from any source
2. You may work in small groups but write your own.
3. Cheating students will face Committee on Discipline.

Following the successful COVID-19 vaccination processed initiated in Assignment 2, the government decided it might be a good idea to keep records of the population. Specially, the records should include information about which subjects are still awaiting vaccination and which subjects were already vaccinated. By implementing this data structure in Java you may yet bring a resolution to the crisis.

The data structure needs to support the following operations:

Operation	Description	Runtime
Insert(Person p)	Insert a new Person into the data structure to await vaccination.	$O(\log n)$
checkVaccinated(Person p)	Look for a Person in the data structure and say if it was vaccinated or not.	$O(\log n)$
listNonVaccinatedSubjects()	Lists all Person objects which await vaccination, sorted according to their ID numbers.	$O(n)$
listVaccinatedSubjects()	Lists all Person objects which were Vaccinated, sorted according to their ID numbers.	$O(n)$
vaccinateSubjects()	Vaccinate all Person objects which await vaccination.	$O(n \log n)$

Implement the data structure in the following way:

- The subjects (represented by the supplied Person class) will be held in 2 AVL trees, ordered according to their ID numbers. The first tree stores subjects that await termination while the second tree stores all the vaccinated subjects.

- New subjects are always inserted into the first tree.

- Whenever the vaccinateSubjects() function is called all subjects from the first tree are moved into the second tree.

The supplied classes contain a detailed API for the functions you will need. You may add any number of functions or fields as you see you fit. **You are not allowed to use the import statement. Do not change the package declaration in the files.**

<u>Submission:</u>

- o You may submit the assignment in pairs, this is not mandatory but recommended.
- o Make sure your code is presentable and is written in good format. Any deviations from these guidelines will result in a point penalty.
- o Make sure your code can be compiled. **Code which does not compile will not be graded.**
- o Submit a zip file with the following file only:
  - Person.java
  - AVLNode.java
  - AVLTree.java
  - VaccinationRecord.java
- o The name of the zip file must be in the following format "ID-NAME.zip", where "ID" is your id and "NAME" is your full.
- o For example, "03545116-Allen_Poe.zip".
- o If you submit as a pair the zip file should be named in the following format "ID-NAME-ID-NAME.zip".
- o For example, "03545116-Allen_Poe-02238761-Paul_Dib.zip".