Tianqi Zhang (001056916)

**Program Structures & Algorithms**

**Spring 2021**

**Extra Assignment**

* **Task**

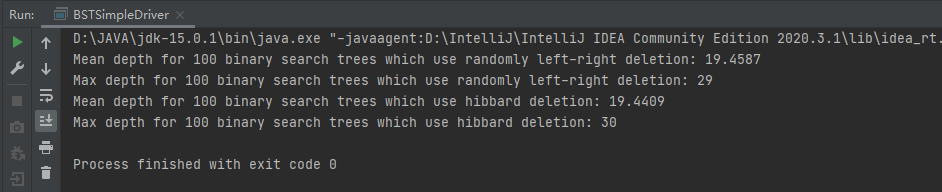
Implement and unit-test a scheme for deletion of nodes in a binary search tree that randomly picks the right-handed or the left-handed approach to deletion (what we talked about in relation to Arbitrary Substitution Principle).

You don’t have to do any benchmarks but I do need to know the following: mean depth and maximum depth for each of the following approaches:

* Just use the “Hibbard” deletion as described in the powerpoint;
* Randomly choose the right vs. left approach as described above.

In each case, you are to mix 1024 key additions randomly with 512 deletions (do not do all the additions and then all the deletions). Keys should be random.

* **Output**

****

* **Conclusion:**

It turns out that whether we choose Hibbard Deletion or Random Lfet-right Approach Deletion, the mean depth and maximum depth for these 2 methods are the same. It means that the Random Lfet-right Approach Deletion wouldn’t improve the depths when a binary search tree has been generated, though it doesn’t violate the ASP.

* **Unit tests result:**

BSTTest:

