

# PERFORMANCE, DATA STRUCTURES AND ALGORITHMS

Exercise 30

Exercise 30

Naïve Binary Tree Performance

# Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

PURPOSE

The purpose of this exercise is to explore the impact of the type of data used to build naïve binary search trees.

For this exercise, you are given the solution to Exercise 29, which builds a naïve binary search tree. In addition, you are given a new mainline class that builds a tree from different sets of data and then displays information about the resulting trees.

Study the new mainline class, TestBinarySearchTreePerformance.java, to familiarize yourself with what it does. Run this program with varying values for the variables *maxSize* and *step* and answer the questions below.

ACTIVITIES

Perform each of the following activities. If you have questions, issues, or doubts, please ask for help and do not just guess.

1. What is the relationship between the degree to which a data set is sorted and the height of the resulting tree?

If sorted

Height will be N;

if Randomly shuffled

Height will be about log(N);

1. What are your predictions for relative search time for trees built from random data versus sorted data?

Random data = O(logN);

Sorted data = O(N);

IF NOT BALANCED!

1. What effect will the degree of sorting of the input data have on search time? Why?

It causes increase of search time, because tree will grow only in one direction (right or left, depends on if data ordered ASC or DESC) and height of tree will be N, therefore search time will be N;

1. Save and archive this document along with your answers and upload it to the LMS.