Datastructuren

October 1
Due Date: Monday, October 19

1 Programming Assignment No 3

In this programming assignment you are to generate binary search trees (BSTs) on 500 nodes randomly. Without loss of generality for the experiments you are going to carry out, you can assume that the keys are the first 500 natural numbers. The purpose of the experiment is to compute the average *internal path length (IPL)* – see page 224 in Drozdek for a definition – of binary search trees generated randomly and learn how this compares to the IPLs of perfectly balanced BSTs. In this programming assignment we restrict ourselves to BSTs constructed by insertions from randomly generated permutations of the keys. (In the following assignment we will also look at BSTs which arise from insertions and deletions. Subsequently we will also look at splay trees.)

1. Implement an algorithm for the random generation of permutations.

2. Construct 100 binary search trees of 500 nodes by generating first a random permutation of the 500 keys and subsequently construct the tree by carrying out the 500 insertions. Compute for each of the generated permutation/BST the IPL and the average path length. In addition compute the average of the 100 IPLs and the standard deviation. Interpret the results you have obtained.

Turn your C++ code and the paragraph of text with the interpretation of your results as a zipped file to Simon Zaaijer (email: szaaijer@liacs.nl) on or before Monday, October 19. Make sure both the C++ files and the interpretation contain the names of your team. Furthermore turn in a hardcopy of your files (C++ and text) in the Lecture of Monday, October 19. You can work on this assignment in pairs. Refer questions to szaaijer@liacs.nl or deutz@liacs.nl.