

**Lab report**

|  |  |
| --- | --- |
| **Course**: | Class Libraries and Data Structures |
| **Semester**: | 1st semester of the academic year **2020-2021** |
| **Major**: | Software Engineering |
| **Class**: | 2019 |
| **Student Name**: | 冯春霖 |
| **Student ID:** | 222019321062074 |
| **Teacher:** | ZHAO, Hengjun (赵恒军) |

**School of Computer and Information Science**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | | Binary Search Tree | | | |
| Date | | Jan，2021 | Type | | □Confirmatory  √ Design  □Comprehensive |
| 1. **Objective & Requirements**    1. Understand the concept and property of binary search tree    2. Get familiar with the insert, delete and find operations on binary search tree    3. Grasp the design of recursive or iterative algorithms about binary search tree | | | | | |
| 1. **Experimental environment (**platform and software**)**   Windows 7 (or higher versions) + Visual Studio 2010 (or higher versions) | | | | | |
| 1. **Experimental content and design** (Main Content, Procedure, Codes and Results)   Task 1  Generate a series of integers (of size n) randomly and insert them into an empty binary search tree, and compute the height of the tree.  Code of bst.h:  template<typename T>  int BinSearchTree<T>::height() const  {  return getHeight(root);  }  template<typename T>  int BinSearchTree<T>::getHeight(Node\* \_node) const  {  return \_node == NULL ? -1 : max(getHeight(\_node->left) + 1, getHeight(\_node->right) + 1);  }  Define a new function with reference to get the height of the tree, in which the height of the tree is obtained by recursive calls. The referenced function is called in the unreferenced function and the root node is passed in as an argument.  Screenshot of the program: | | | | | |
| 1. **Result analysis and discussion**（Analysis of experimental results and summing up the harvest and the existing problems）   Through this experiment, I have a certain understanding of the data structure of the binary search tree, and complete the function to get the height of the tree. In completing this function, I made some optimization of the recursive call to the parameter function, so that the final result is only one line of code, which makes me feel very accomplished | | | | | |
| Comments & Evaluation | Content & Design (A-E) | | |  | |
| Procedure & Codes (A-E) | | |  | |
| Results (A-E) | | |  | |
| Analysis & Discussion (A-E) | | |  | |
| Score (A-E):  Feedback comments: | | | | |