Readme - Functions of programs

To solve this problem, we give three codes, whose functions are as follows:

1.1 *project2.cpp*

1.1.1 Input

Includes $n + 2$ lines.

a) First line contains a single integer n.

b) In the following n lines, numbered $i = 0, 1, \ldots n - 1$, each line contains a pair of integers l and r, seperated by spaces, which represents the left and right child of node i.

c) Last line contains n intergers seperated by spaces, the elements in the BST.

1.1.2 Output

One line, n integers seperated by spaces, the breadth-first-search order of tree nodes. **Note that, in the code, add `#define Debug` command, the code will output some helpful info for you to draw the tree you've input.**

1.2 *gen.cpp*

1.2.1 Input

n and lim, the number of nodes and max element limit.

1.2.2 Output

The program will out put a random input for *project2.cpp*, and the standard answer will be output to *ans.txt*, then you can check if *project2.cpp* works well.

1.3 *checker.cpp*

1.3.1 Input

n and lim, the number of nodes and max element limit.

1.3.2 Output

The program will check *project2.cpp* with *gen.cpp* automatically. It use *gen.cpp* to generate a random input for *project.cpp*, and then run project2.cpp, finally compare the output of *project2.cpp* (redirected to *out.txt*) and *ans.txt*. If the output is correct, *checher.cpp* will output ***Accepted!***