文本

描述已自动生成

#include <bits/stdc++.h>

#include <vector>

**using** **namespace** std;

**const** int SZ = 100001;

**struct** node {

char v;

node \*lch, \*rch;

node(char value) : v(value), lch(**nullptr**), rch(**nullptr**) {}

};

int lch[SZ] {}, rch[SZ] {};

string sq;

node\* gen(int &i) {

**if** (sq[i] == '@') {

i++;

**return** **nullptr**;

}

node\* root = **nullptr**;

root = **new** node(sq[i++]);

root->lch = gen(i);

root->rch = gen(i);

**return** root;

}

void dfs\_pre(node \*root) {

**if** (!root)

**return**;

cout << root->v;

dfs\_pre(root->lch);

dfs\_pre(root->rch);

}

void dfs\_in(node \*root) {

**if** (!root)

**return**;

dfs\_pre(root->lch);

cout << root->v;

dfs\_pre(root->rch);

}

void dfs\_post(node \*root) {

**if** (!root)

**return**;

dfs\_pre(root->lch);

dfs\_pre(root->rch);

cout << root->v;

}

bool dfs\_sch(node \*root, char trg) {

**if** (!root)

**return** false;

**if** (root->v == trg) {

cout << "lch:" << (root->lch ? root->lch->v : '^') << endl;

cout << "rch:" << (root->rch ? root->rch->v : '^') << endl;

**return** true;

}

**if** (dfs\_sch(root->lch, trg))

**return** true;

**if** (dfs\_sch(root->rch, trg))

**return** true;

**return** false;

}

int main() {

*//ABD@F@@@CE@@@*

cout << "type order:";

std::getline(cin, sq);

int tmp = 0;

node \*tree = gen(tmp);

cout << "pre:";

dfs\_pre(tree);

cout << endl;

cout << "in:";

dfs\_in(tree);

cout << endl;

cout << "post:";

dfs\_post(tree);

cout << endl;

char in;

**for**(;;) {

cout << "Type the node you want to check ($ to end)";

cin >> in;

**if** (in == '$')

**break**;

**if** (!dfs\_sch(tree, in))

cout << "node not found." << endl;

}

**return** 0;

}