**树的综合查询实验报告**

Class:计科201

Experiment Date:2021-12-07

Student ID 1:20401090106 Name 1:周丽菲

Student ID 2：20401010120 Name 2：张诚林

Student ID 3：20401010123 Name 3：吴健康

**One. Experimental purpose**

1、Do linear list operations by using basic operations of tree or binary tree.

2、Handle file operations.

3、Deep the understanding of tree and binary tree, gradually develop the programming ability to solve practical problems.

**Two. Experimental environment**

Computers equipped with Visual C6.0/CFree.

The experiment lasted for 4 hours.

**Three. Experimental content**

（Choose one of the following two:）

1. The directory structure of the specified directory file system is given, you should write into the file dir\_structure.txt by the indentation method, and calculate the storage area of the directory.。（The indentation method is shown in P93）
2. design a "automatic calculator" as follows：

（1）The expression that needs to be calculated is stored in the text file in the TXT text；

（2）Each line in the text is an expression；

（3）Expressions include operands, operators such as addition, subtraction, multiplication and division, and parentheses；

For example： （34-72.3）\*54.7-82.4

（4）"Automatic calculator" calculates each expression in the text file according to the input file name, and writes every expression of the result to the original file name in the \_out.txt, you should use the method of covering and when saving the records. The format of each row is：

expression = result。

For example：the original file is: A1.txt

The file for output is： A1\_out.txt

The format of the text in A1\_out.txt is：

（34-72.3）\*54.7-82.4 = -2177.41

For all the calculated results, you'd keep 4 digits after the decimal point if it is decimal.

（5）Generate a statistical document after the calculation, its content is：

Execution time：xxxx-xx-xx hh:mm:ss

The total number of expressions is：XXX

The number of correct expressions is：XXX

The number of error expressions is：XXX

Naming rules for filenames：original file name :\_log.txt，Write files with append write method。

For example：A1.txt corresponding to the statistical file：A1\_log.txt

**Special remind**：★The calculation process requires transform the infix expression to the postfix expression and then transform the postfix expression to expression tree. Finally get the result by calculating the expressions.

(If you are getting into trouble in calculating decimal, you can only consider integer calculation.)

**Four. Important data structures**

#include<iostream>

#include<cstring>

using namespace std;

typedef struct node{

char data;

struct node \*lchild;

struct node \*rchild;

} BTNode;

void CreateBTNode(node\* b,char \*str){

int len=strlen(str);

node\* list[99];

node\* p=nullptr;

int top=-1;

char decide=' ';

for(int i=0;i<len;i++){

int c=str[i];

if(c=='('){

list[++top]=p;

decide='l';

}else if(')'){

top--;

}else if(','){

decide='r';

}else{

p=new node();

p->data=c;

if(b==nullptr){

b=p;

}else{

if(decide='l'){

list[top]->lchild=p;

}else if(decide='r'){

list[top]->rchild=p;

}

}

}

}

}

node\* findNode(node \*b,char x){

node\* p;

if (b==nullptr) return nullptr;

else if (b->data==x) return b;

else

{ p=findNode(b->lchild,x);

if (p!=nullptr) return p;

else return findNode(b->rchild,x);

}

}

node\* lchildNode(node \*p)

{

return p->lchild;

}

node\* rchildNode(node \*p)

{

return p->rchild;

}

int depth(node \*b)

{ int lchilddep,rchilddep;

if (b==nullptr) return 0;

else

{

lchilddep=depth(b->lchild);

rchilddep=depth(b->rchild);

return(lchilddep>rchilddep)?

(lchilddep+1):(rchilddep+1);

}

}

void display(node \*b) {

if (b!=nullptr){

cout<<b->data;

if (b->lchild!=nullptr || b->rchild!=nullptr)

{ printf("(");

display(b->lchild);

if (b->rchild!=nullptr) cout<<",";

display(b->rchild);

cout<<")";

}

}

}

**Five. Realization idea analysis**

指针指向需要清楚理解，返回值要明确

**Six. Program debugging problem analysis**

对于函数运用及指针的熟练度要加强

**Seven. Experimental summary**

树的查询需要对数据结构认知清楚，代码结构调理逻辑清晰，多写，深度理解

**Eight. Crew Division**

|  |  |  |
| --- | --- | --- |
| **Group division** | | |
| **Member name** | **Work done** | **Completion situation** |
| **周丽菲** | **资料查找及算法分析** | **完成** |
| **张诚林** | **代码书写及结构理解** | **完成** |
| **吴健康** | **报告书写及代码测试** | **完成** |