**第三章作业**

**第一题（3.1） 对一个栈的输入序列 a1,a2,a3,… ,an,，称由此栈依次出栈后所得到的元素序列为栈 的合法输出序列。例如，假设栈 S 的一个输入序列为 1,2,3,4,5，则可得到多个输出序列， 例如，1,2,3,4,5 就是一个合法的输出序列，同理，5,4,3,2,1 和 3,2,1,4,5 也分别是其合 法的输出序列。分别求解下列问题：**

**（1）判断序列 1,3,4,5,2 是否是合法的输出序列。**

**是合法的输出序列。**

**（2）对输入序列 1,2,3,4,5，求出其所有的合法的输出序列。**

**54321 45321 43521 43251 43215 35421 34521 34251 34215 32541 32451 32415 32154 32145 25431 24531 24351 24315 23541 23451 23415 23154 23145 21543 21453 21435 21354 21345 15432 14532 14352 14325 13542 13452 13425 13254 13245 12543 12453 12435 12354 12345**

**（3\*）设计算法以判断对输入序列 1,2,3,… ,n，序列 a1,a2,a3,… ,an是否是该栈的合法 的输出序列（假设输出序列在数组 A 中）。**

#include<stack>

#include<iostream>

#include<string.h>

using namespace std;

int n=0,t=0;

char \*str;

void outputstack(stack<char>s,char \*output,char \*input,int in\_index,int out\_index){

if(s.empty()&&in\_index==n){

cout<<output<<" ";

if(strcmp(str,output)==0){

t=1;

}

}

else{

if(in\_index<n){

s.push(input[in\_index]);

outputstack(s,output,input,in\_index+1,out\_index);

s.pop();

}

if(!s.empty()){

output[out\_index]=s.top();

s.pop();

outputstack(s,output,input,in\_index,out\_index+1);

}

}

}

int main(){

char input[100];

cout<<"请输入输入序列：";

cin>>input;

stack<char>s;

n=strlen(input);

char \*output=new char[n];

str=new char[n];

cout<<"请输入判断序列：";

cin>>str;

cout<<"输出序列："<<endl;

outputstack(s,output,input,0,0);

cout<<endl;

if(t==1){

cout<<"判断序列是该栈的合法序列"<<endl;、/

}

else{

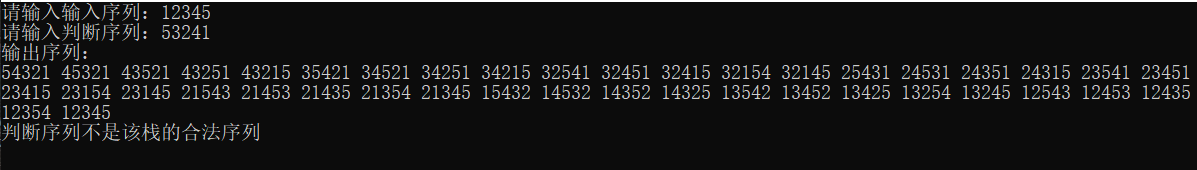
cout<<"判断序列不是该栈的合法序列"<<endl;

}

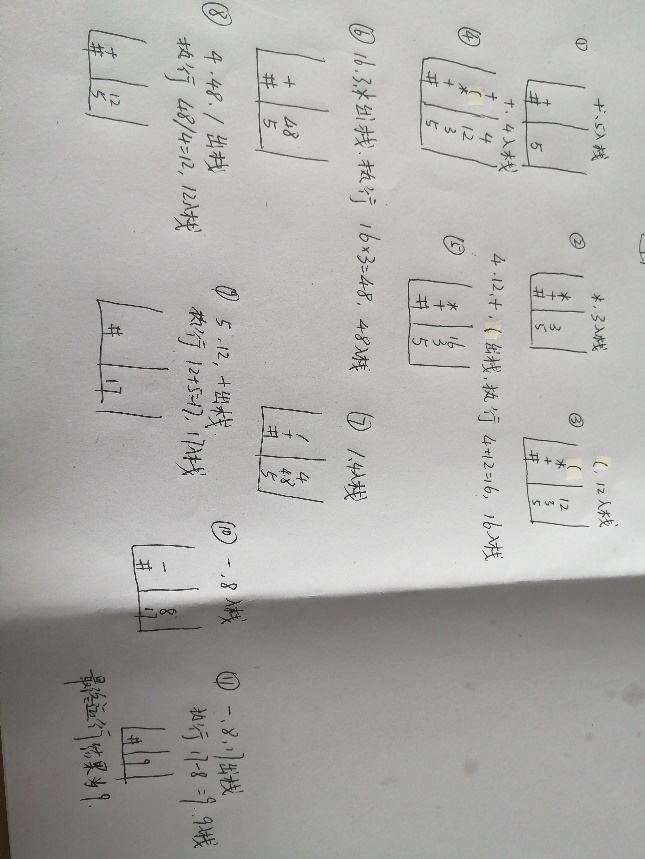
delete []output;

delete []str;

}



**第二题（3.5）对表达式 5+3\*(12+4)/4-8，依次画出在求解过程中的各步骤中的栈的状态。**

****

**第三题（3.13）**

**阅读下列程序，并写出其运行结果:**

**(1) void P(int W) {**

**if (W>0)**

**{ P(W-1);**

**cout<<W;**

**}**

**}**

**调用P（4）**

**运算结果：1234**

**（2） void P(int W) {**

**if (W>0)**

**{ cout<<W;**

**P(W-1);**

**}**

**}**

**调用 P(4);**

**运算结果：4321**

**（3） void P(int W) {**

**if (W>0)**

**{ cout<<W;**

**P(W-1);**

**cout<<W;**

**}**

**}**

**调用 P(4);**

**运算结果：43211234**

**（4）void P(int W)**

**{**

**if (W>0)**

**{**

**P(W-1);**

**P(W-1);**

**cout<<W;**

**}**

**}**

**调用 P(4);**

**运算结果：112112311211234**

**（5）int F(int N)**

**{**

**if ( N==0) F=0;**

**else if (N==1)**

**return 1;**

**else**

**return F(N-1)+2\*F(N-2);**

**}**

**调用 cout<<F(5);**

**运算结果：1112**

**(6) void P(int N,int \*F)**

**{**

**if (N==0)**

**\*F=0;**

**else { P(N-1, \*F); F=F+N; }**

**}**

**调用 P(4,\*M); cout<<\*M**

**运算结果：10**

**第四题（ppt例3.10）**

**对下面的程序代码，画出调用F(5)的执行路线图，并给出执行结果。**

**int F( int n )**

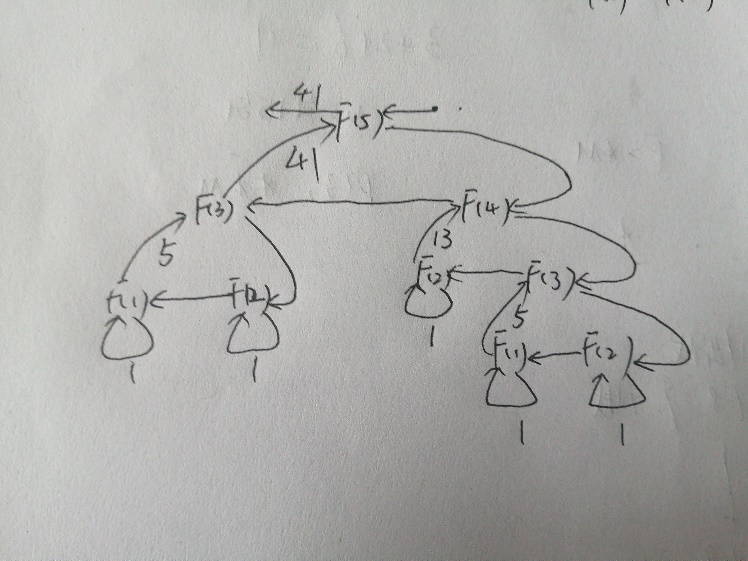
**{**

**if( n<=2 )**

**return 1;**

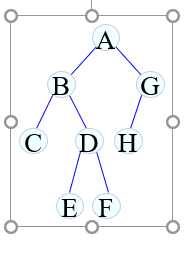
**return 2\*F(n-1) + 3\*F(n-2);**

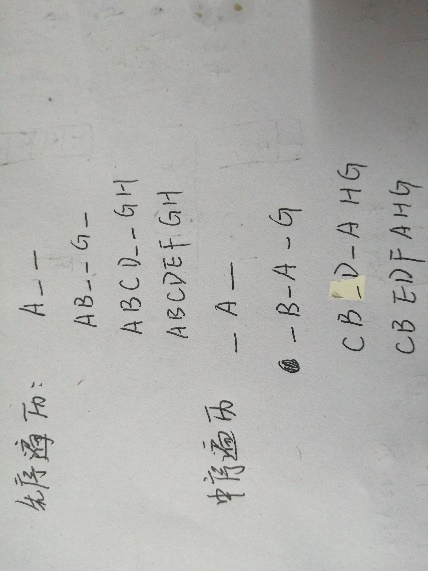
**}**



**执行结果为：41**

**第五题（二叉树ppt88页图）按照ppt方法先序，中序遍历下面这棵二叉树，需画图**



****