第3次作业

1. 书133页第4题：设计算法判断一个算术表达式的圆括号是否正确配对。（提示：对表达式进行扫描，凡遇“（”就进栈，遇“）”就退掉栈顶的“）”，表达式被扫描完毕，栈应为空。）

[题目分析]题目没有说明算术表达式的存储方式。这里我们就采用最简单存储方式来解决问题。假定算术表达式以字符串的形式存储。

#define maxsize 100 //顺序栈容量

typedef char datatype; //定义堆栈的数据类型

typedef struct{

datatype elements[maxsize];

int Top;

}seqstack; //顺序栈类型名称

void SETNULLS(seqstack \*S){ //置空栈

S->Top = -1;

}

int EMPTYS(seqstack \*S){ //判断栈是否为空

if (S->Top>=0) return(0);

else return(1);

}

seqstack \*PUSHS(seqstack \*S, datatype E){//进栈

if(S->Top>=maxsize-1){

printf(“Stack Overflow”);

return(NULL);

}

else{

S->Top++;

S->elements[S->Top]=E;

}

Return(S);

}

datatype POPS(seqstack \*S){

datatype \*tmp;

if(EMPTYS(S)){

printf(“Stack Underflow”);

return NULL;

}

else{

tmp=(datatype \*)malloc(sizeof(datatype));

\*tmp=S->elements[S->Top];

S->Top--;

}

}

**void match(char \*s){ //s为字符串的首地址指针**

**seqstack \*stack; //定义一个堆栈**

**int i;**

**SETNULLS(stack); //将堆栈置空**

**for(i=0; s[i]!='\0'; i++){ //字符串以'\0'结尾**

**if( s[i]=='(' ) PUSHS(stack, s[i]); //遇到'（'就进栈**

**else if( s[i]==')' ) //若字符为'）'**

**if( EMPTYS(stack) ){ //若堆栈为空，则括号匹配错误**

**printf("no!");**

**return(0);**

**}**

**else POPS(stack); //遇到'）'且栈非空，就出栈**

**else continue; //其它字符不做处理**

**}//for循环结束，所有字符都检查完毕**

**if(EMPTYS(stack)) printf("yes!"); /\*正好空栈的情况，括号匹配正确\*/**

**else printf("no!"); //否则括号匹配错误**

**}**

1. 书143页第6题：画出下列矩阵X的三元组。



参考答案：

|  |  |  |  |
| --- | --- | --- | --- |
|  | i | j | v |
| 0 | 0 | 0 | 15 |
| 1 | 0 | 3 | 22 |
| 2 | 0 | 5 | -15 |
| 3 | 1 | 1 | 11 |
| 4 | 1 | 2 | 3 |
| 5 | 2 | 3 | -6 |
| 6 | 4 | 0 | 91 |
| 7 | 5 | 2 | 28 |