

7-1 表达式转换 (25 分)

算术表达式有前缀表示法、中缀表示法和后缀表示法等形式。日常使用的算术表达式是采用中缀表示法，即二元运算符位于两个运算数中间。请设计程序将中缀表达式转换为后缀表达式。

输入格式：

输入在一行中给出不含空格的中缀表达式，可包含+、-、*、\以及左右括号()，表达式不超过20个字符。

输出格式：

在一行中输出转换后的后缀表达式，要求不同对象（运算数、运算符号）之间以空格分隔，但结尾不得有多余空格。

输入样例：

2+3*(7-4)+8/4

输出样例：

2 3 7 4 - * + 8 4 / +

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<map>
#include<iostream>
using namespace std;

int main(){
    char str[20];
    char stack[20];
    char *copy;
    copy=(char*)malloc(sizeof(char)*20);
    map<char,int> p;
    p['+']=p['-']=1;
    p['*']=p['/']=2;
    p['(']=p[')']=3;
    int top;
    int base;
    int count=0;
    base=top=0;
    scanf("%s",&str);
    for(int i=0;i<strlen(str);i++){
        if(str[i]=='+'||str[i]=='-'||str[i]=='*'||str[i]=='/'||str[i]=='('){
            if(top==base){
                stack[++top]=str[i];
            }else if(p[str[i]]>p[stack[top]]){
                stack[++top]=str[i];
            }else{
                while(top>base&&
                    p[stack[top]]>p[str[i]]){
                    top--;
                }
                stack[++top]=str[i];
            }
        }else{
            copy[count++]=str[i];
        }
    }
    copy[count]='\0';
    printf("%s",copy);
}
```

```

        if(stack[top]!=''){
            while(top!=base&& p[stack[top]]>=p[str[i]]){
                copy[count++]=stack[top--];
            }
            stack[++top]=str[i];
        }else{
            stack[++top]=str[i];
        }
    }
    //printf("1");
}else if(str[i]==')'){
    while(stack[top]!=''){
        copy[count++]=stack[top];
        top--;
    }
    top--;
}else{
    copy[count++]=str[i];

    //printf("3");
}
}
while(top!=base){
    copy[count++]=stack[top--];
}
printf("%s",copy);
}

```

```

#include<stdio.h>
#include<stdlib.h>
#include<string.h>

```

```

int main(){
    char str[20];
    char stack[20];
    char *copy;
    copy=(char*)malloc(sizeof(char)*20);
    char p[3];
    p['+']=p['-']=1;
    p['*']=p['/']=2;
    p['(']=p[')']=3;
}

```

```

int top;
int base;
int count=0;
base=top=0;
scanf("%s",&str);
for(int i=0;i<strlen(str);i++){
    if(str[i]=='+'||str[i]=='-'||str[i]=='*'||str[i]=='/'||str[i]=='('){
        if(top==base){
            stack[++top]=str[i];
        }else if(p[str[i]]>p[stack[top]]){
            stack[++top]=str[i];

        }else{
            if(stack[top]!=''){
                while(top!=base&&p[stack[top]]>=p[str[i]]){
                    copy[count++]=stack[top--];

                }
                stack[++top]=str[i];
            }else{
                stack[++top]=str[i];
            }
        }
        //printf("1");
    }else if(str[i]==')'){
        while(stack[top]!='('){
            copy[count++]=stack[top];
            top--;

        }
        top--;
    }else{
        copy[count++]=str[i];

        //printf("3");
    }
}
while(top!=base){
    copy[count++]=stack[top--];
}
printf("%s",copy);

}

```

