stack S;  
int i=0,sum;  
int x,y;  
char s[100];  
InitStack(&S);  
scanf\_s("%s",s);  
while (s[i] != '\0')  
{  
if(s[i]<='9'&&s[i]>='0')  
Push(&S, s[i] - 48);  
if (s[i] == '+')  
{  
x = Pop(&S, &x);  
y = Pop(&S, &y);  
sum = x + y;  
Push(&S, sum);  
printf("加法运算结果是%d+%d=%d\n ",x,y, sum);  
}  
if (s[i] == '-')  
{  
x = Pop(&S, &x);  
y = Pop(&S, &y);  
sum = y - x;  
Push(&S, sum);  
printf("减法运算结果是%d-%d=%d\n ",y,x, sum);  
}  
if (s[i] == '\*')  
{  
x = Pop(&S, &x);  
y = Pop(&S, &y);  
sum = x \* y;  
Push(&S, sum);  
printf("乘法运算结果是%d\*%d=%d\n ",y,x,  sum);  
}  
if (s[i] == '/')  
{  
x = Pop(&S, &x);  
y = Pop(&S, &y);  
sum = y / x;  
Push(&S, sum);  
printf("除法运算结果是%d/%d=%d\n ",x,y,  sum);  
}  
i++;  
}  
printf("运算结果是%d ", sum);

这是我后来改写 调试的一个主函数，时间长了，原来弄好的代码都找不到了，我记得当时有给问题就是用两个栈分别存储输入的数字和运算符号，还是用一个栈存储数字和字符，这两种方式决定了输入的方式。