/\*\*

\* RPN.c 逆波兰计算器

\*

\* @author Darbuly 970073804@qq.com

\* @copylight 2018-2019 DBL

\*

\*

\*

\*/

#include <stdio.h>

#include <stdlib.h>

#include <ctype.h>

#define STACK\_INIT\_SIZE 20

#define STACKINCREMENT 10

#define OK 1

#define ERROR 0

#define SOVERFLOW -1

#define MAXBUFFER 10

typedef double ElemType;

typedef int Status;

typedef struct

{

ElemType \*base;

ElemType \*top;

int stacksize;

}sqStack;

Status InitStack(sqStack \*s)

{

s->base = (ElemType \*)malloc(STACK\_INIT\_SIZE\*sizeof(ElemType));

if(!s->base)return ERROR;

s->top = s->base;

s->stacksize = STACK\_INIT\_SIZE;

return OK;

}

Status Push(sqStack \*s,ElemType e)

{

if(s->top - s->base >=s->stacksize)

{

s->base = (ElemType \*)realloc(s->base,(s->stacksize+STACKINCREMENT)\*sizeof(ElemType));

if(!s->base)return ERROR;

s->top = s->base + s->stacksize;

s->stacksize +=STACKINCREMENT;

}

\*(s->top++) = e;

return OK;

}

Status Pop(sqStack \*s,ElemType \*e)

{

if(s->top == s->base)return ERROR;

\*e = \*(--s->top);

return OK;

}

int StackLen(sqStack s)

{

return(s.top - s.base);

}

int main()

{

sqStack s;

ElemType a,b,d,sum=0;

int i;

char c,str[MAXBUFFER];

InitStack(&s);

printf("Please enter RPN expression end by #:\n");

scanf("%c",&c);

while(c!='#')

{

while(isdigit(c) || c=='.')

{

str[i++]=c;

str[i]='\0';

if(i>=MAXBUFFER)

{

printf("\nError!Overflow!\n");

return -2;

}

scanf("%c",&c);

if(c==' ')

{

d = atof(str);

Push(&s,d);

i = 0;

break;

}

}

switch(c)

{

case '+':

//printf("It is a Inc\n");

Pop(&s,&b);

Pop(&s,&a);

Push(&s,a+b);

break;

case '-':

Pop(&s,&b);

Pop(&s,&a);

Push(&s,a-b);

break;

case '\*':

Pop(&s,&b);

Pop(&s,&a);

Push(&s,a\*b);

break;

case '/':

Pop(&s,&b);

Pop(&s,&a);

if(b==0)

{

printf("\nError,b can not be zero!\n");

return -1;

}

Push(&s,a/b);

break;

}

scanf("%c",&c);

}

Pop(&s,&d);

printf("\nResult is : %f\n",d);

return 0;

}

时间复杂度O(1)

1.过滤思想：有时候我们想象中的一般情况，可以考虑过滤思想过滤。