数据结构实验报告

院系：信息工程学院

专业：软件工程二班

学号：20122480212

姓名：霍静静

上机时间：10月18号

栈的实验报告

**实验三：栈的基本操作**

**一、实验目的：**了解栈的逻辑结构特征，熟练掌握栈存储结构的描述方法，熟练掌握栈的基本操作。

**二、实验要求**：

1. 建预定义常量和类型的头文件；

2）定义栈类型；

3）创建栈函数，实现9个基本操作；

4）写函数完成实验内容要求；

5）写出主函数调用，调试运行程序完成整个实验，显示结果；

**三、实验内容：**

1. 栈的9个基本操作和进制之间的转换。

**四、算法思想：**

**首先给出栈的结构类型，申请一个连续的空间构造一个空栈（其数据元素的类型为整形），然后实现DestryStack();ClearStake();等9个函数的操作，最后在主函数中调用各函数实现十进制数转换为八进制数。**

**主要是利用栈的后进先出的特性还完成数值之间的转换的。**

**五、主要函数的实现：**

#include<stdio.h>

#include<stdlib.h>

#include<malloc.h>

#define STACK\_INIT\_SIZE 10 //初始时给的空间

#define STACK\_INCREMENT\_SIZE 10 //内存申请不足时扩展的大小

#define OVERFLOW 0

#define FALSE 0

#define TRUE 1

#define ERROR 0

#define OK 1

typedef int SElemType;

typedef int Status;

typedef struct{

int \*base;

int \*top;

int stacksize;

}Sqstack; //顺序栈

//----The function prototype of basic operation----

Status InitStack(Sqstack &s);

Status DestroyStack(Sqstack &s);

Status ClearStack(Sqstack &s);

Status StackEmpty(Sqstack s);

int StackLength(Sqstack s);

Status GetTop(Sqstack &s,SElemType &e);

Status Push(Sqstack &s,SElemType e);

Status Pop(Sqstack &s,SElemType &e);

Status StackTraverse(Sqstack &s);

void conversion(Sqstack &s,int N);

//----Algorithm Description of basic operation---

Status InitStack(Sqstack &s)

{

s.base=(SElemType\*)malloc(sizeof(SElemType) \* STACK\_INIT\_SIZE);

if(!s.base) exit(OVERFLOW); //分配空间失败

s.top=s.base;

s.stacksize=STACK\_INIT\_SIZE;

return OK;

}

Status StackEmpty(Sqstack s)

{

if(s.top==s.base)

return TRUE;

else return FALSE;

}

int StackLength(Sqstack &s)

{

printf("\n%d\n",s.top-s.base);

}

Status GetTop(Sqstack &s,SElemType &e)

{

if (s.top == s.base) //如果空栈，报错

return ERROR;

else {

e=\*(s.top-1);

return OK;

}

}

Status DestroyStack(Sqstack &s)

{

free(s.base);

s.base=NULL;

return OK;

}

Status ClearStack(Sqstack &s)

{

s.top=s.base;

return OK;

}

Status Push(Sqstack &s,SElemType e)

{

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

{

s.base=(SElemType\*)realloc(s.base,(s.stacksize+STACK\_INCREMENT\_SIZE)\*sizeof(SElemType));

if(!s.base) exit(OVERFLOW);

s.top=s.base+s.stacksize;

s.stacksize+=STACK\_INCREMENT\_SIZE;

}//栈满，追加空间

\*s.top++=e;

return OK;

}

Status Pop(Sqstack &s,SElemType &e)

{

if(s.top==s.base) return ERROR;

e=\*--s.top;

return OK;

}

Status visit(SElemType e)

{

printf("e=%d\n",e);

return OK;

}

Status StackTraverse(Sqstack &s, Status (\*visit)(SElemType))

{

SElemType \*p;

if (s.top == s.base)

return ERROR;

p=s.top; //p指向栈顶

while( p>s.base )

visit(\* --p); //通过循环对栈的元素遍历

return OK;

}

void conversion(Sqstack &s,int N)

{

while(N)

{

Push(s,N%8);

N=N/8;

}

while(!StackEmpty(s))

{

SElemType e;

Pop(s,e);

printf("%d",e);

}

}

int main()

{

int N;

printf("请输入一个要转换的数字：");

scanf("%d",&N);

Sqstack s;

InitStack(s);

conversion(s,N);

getchar();getchar();

}

**六、测试结果：**

**该数值转换的算法时间复杂度是是O(n);**

**当输入数据是123456时，输出为：361100；**

**七、实践总结：**

**1.写程序出的最大的一个问题是在调试的时候自己没能及时找出scanf()函数里面的参量没有加地址符这个错误；**

**2.刚开始不知道怎么用status，现在用的还不够熟练；**

**3.如果能把栈的基本操作做成头文件就更好了，但自己现在还不会做；**