|  |
| --- |
| 哈尔滨工业大学(深圳) |
| **《数据结构》实验报告** |
|  |
| 实验二  栈与队列及其应用  学 院: 计算机科学与技术   |  |  | | --- | --- | | 姓 名: | 段佳慧 | | 学 号: | 200110807 | | 专 业: | 计算机科学与技术 | | 日 期: | 2021-04-02 | |

# 问题分析

栈：

1. 建立一个链栈，从文件中读取数据之后计算机将其存储在栈中。再利用数组，将栈中的数据信息存储在数组中
2. 计算机根据过程1的栈，进行相关入栈、出栈、输出栈等操作

队列：

1. 利用两个链栈建立一个队列，其中一个栈为操作栈，另一个栈为存储栈，从文件中读取数据之后计算机将其存储在队列中。
2. 计算机根据过程1的队列，进行相关入队与出队，输出队列等操作

# 二、详细设计

## 2.1 设计思想

栈：

* 1. 栈的每个指针指向一个结构体，结构体包括栈顶指针。当计算机将数据元素存储在栈之后，利用数组将栈中的元素存储在数组中。
  2. 通过循环输出数组中的元素以达到输出栈中元素的目的。
  3. 对于栈来说，只能对其栈顶元素进行操作。入栈时要判断栈是否为满栈。

队列：

1. 队列的结构体包括两个栈（栈1用于操作，栈2用于存储）计算机将数据元素存储在队列之后，利用数组获取队列的一个数组拷贝
2. 通过循环输出数组中的元素以达到输出队列中元素的目的。
3. 对于队列来说，两边操作，注意两个栈的运用。

## 2.2 存储结构及操作

(1) 存储结构 存储结构如下：

（a）栈

typedef struct{

DataType data[MaxSize];

int top; // 栈顶指针

} Stack; // 结构体类型名

（b）队列

typedef struct{

Stack stack1;

Stack stack2;

} Queue;

(2) 涉及的操作（一般为自定义函数，可不写过程，但要注明该函数的含义。）

**（a）栈**

* 输出栈

void testStack(Stack S);

* 初始化一个栈

void InitStack(Stack \*s)

* 返回栈的长度

int StackLength(Stack S)

* 将栈中的数据元素存储在一个数组中

void StackToArray(Stack S, DataType \*seq)

* 判断栈是否为空

int StackEmpty(Stack S)

* 入栈操作，将元素插入到栈中

int Push(Stack \*S, DataType e)

* 出栈元素，弹出栈顶元素

int Pop(Stack \*S, DataType \*e)

* 获取栈顶元素，但不弹出

int GetTop(Stack S, DataType \*e)

**（b）队列**

* 输出队列

void testQueue(Queue Q);

* 初始化一个队列

void InitQueue(Queue \*Q);

* 获取队列头元素但不删除该元素

int GetHead(Queue Q, DataType \*e);

* 判断队列是否为空

int QueueEmpty(Queue Q);

* 入队操作，将元素插入到队列中去

int EnQueue(Queue \*Q, DataType e);

* 出队操作，从队列中取出一个元素

int DeQueue(Queue \*Q, DataType \*e);

* 清空栈

void ClearQueue(Queue \*Q);

* 获取队列的长度

int QueueLength(Queue Q);

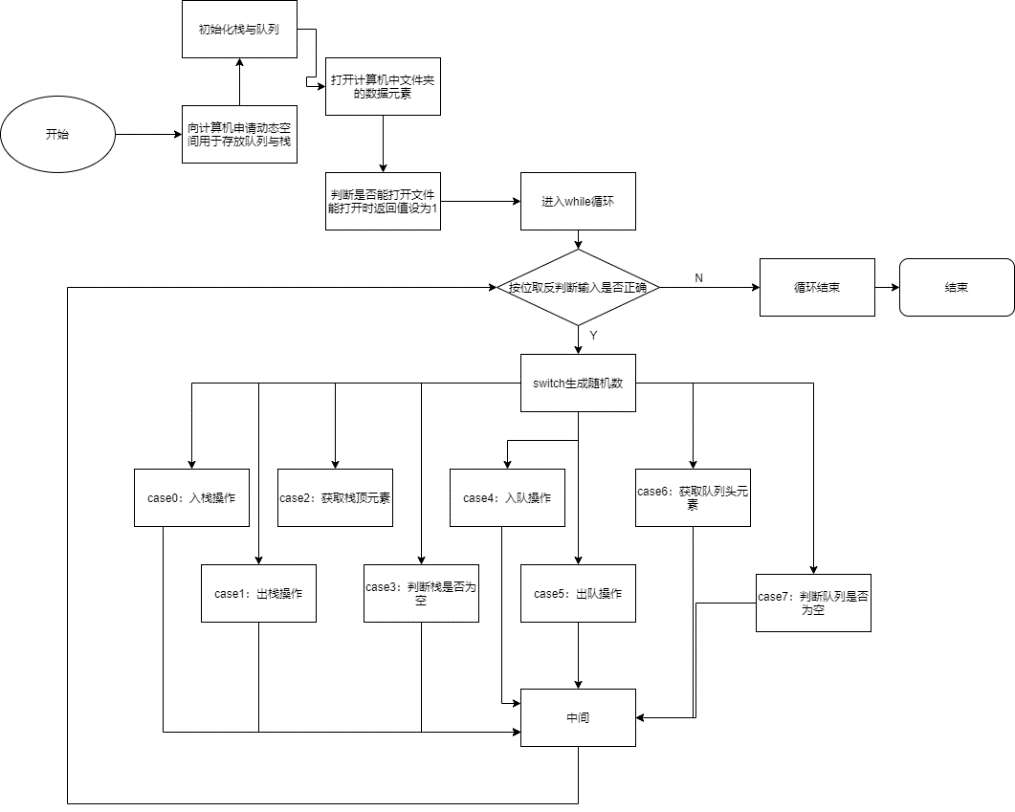
* 用一个数组存储队列中的元素

void QueueToArray(Queue Q, DataType \*seq);

## 2.3 程序整体流程

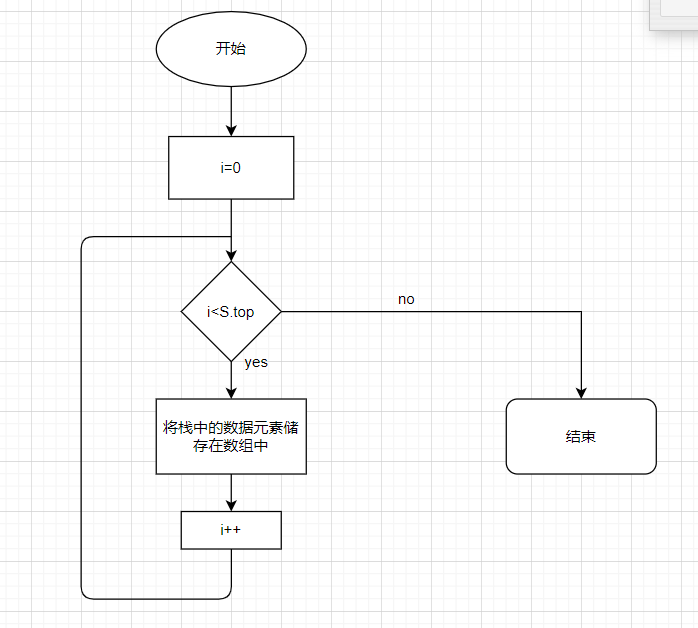
画出整体流程，及核心算法流程。

整体流程：

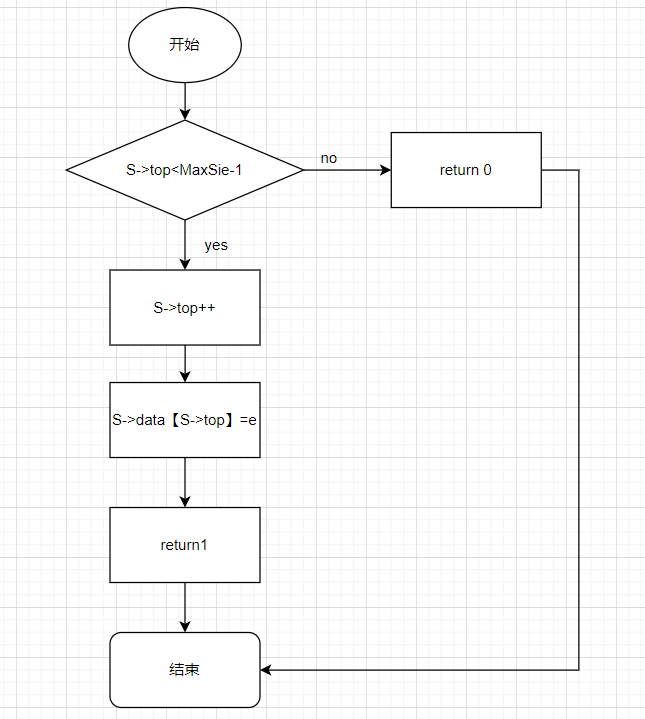


核心算法流程：

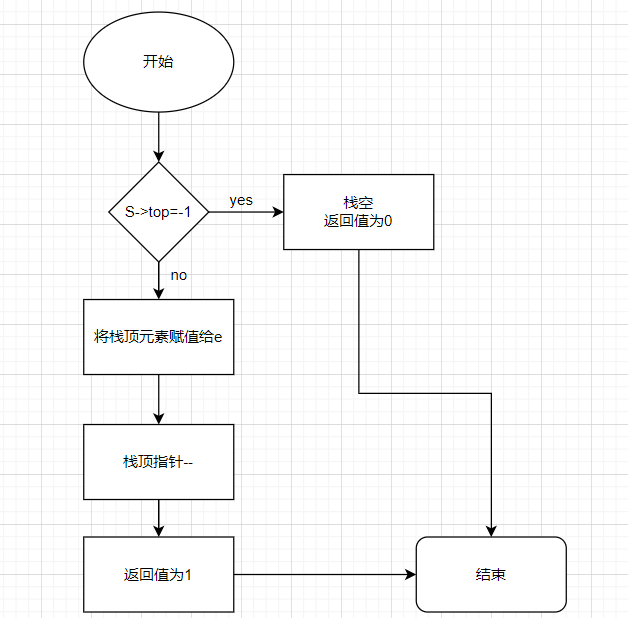
* StackToArray



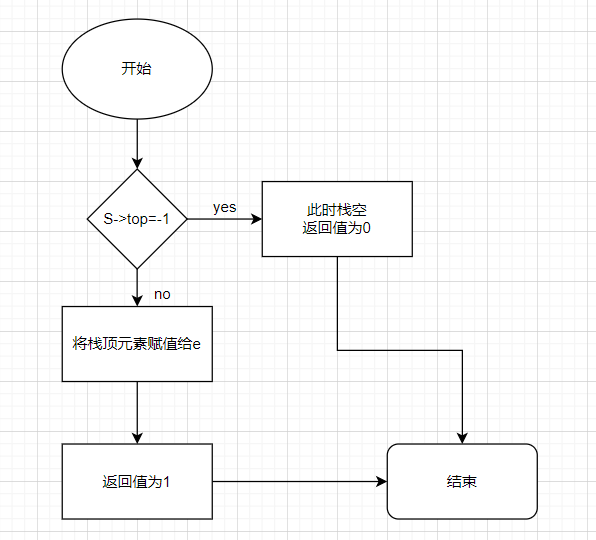
* Push



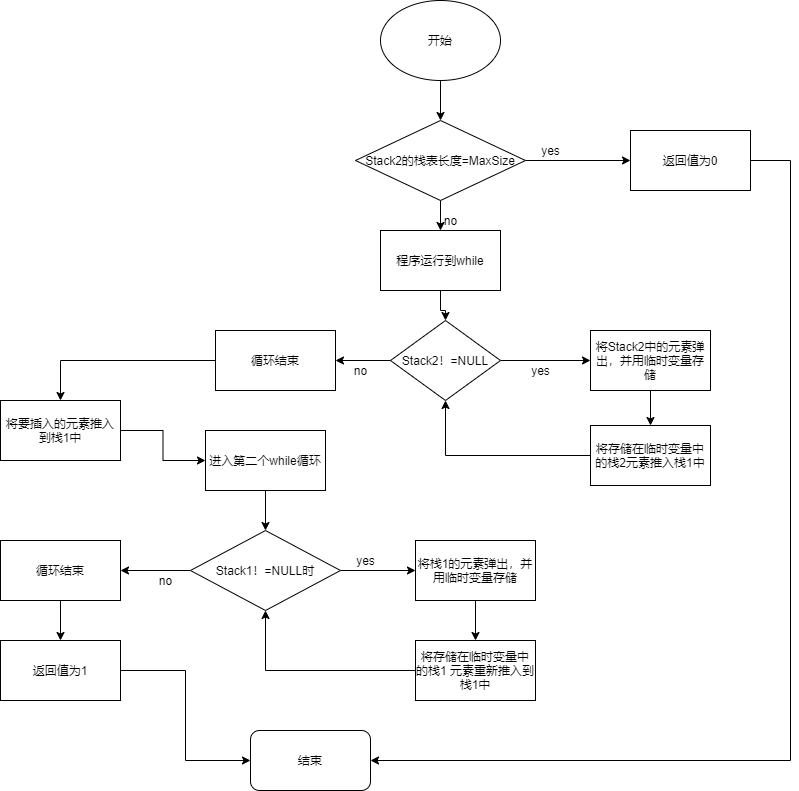
* Pop



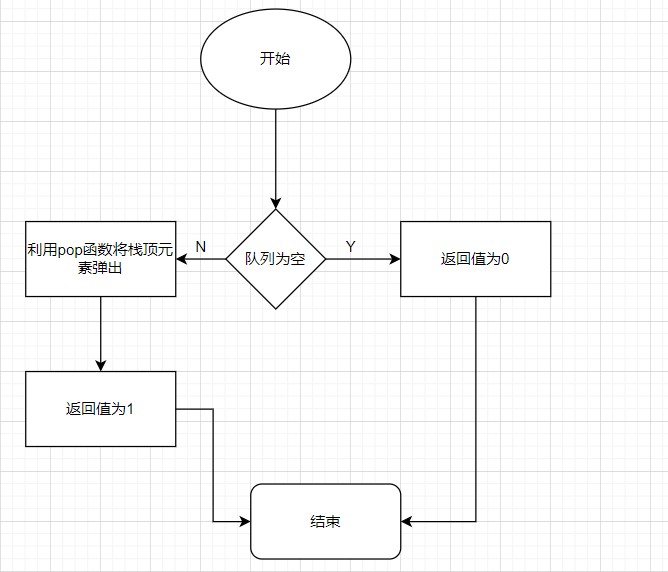
* Gettop获取栈顶元素



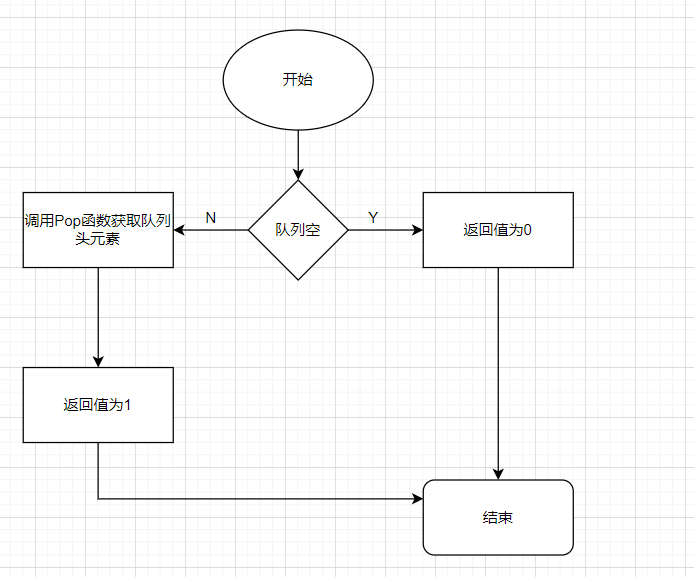
* EnQuene入队操作



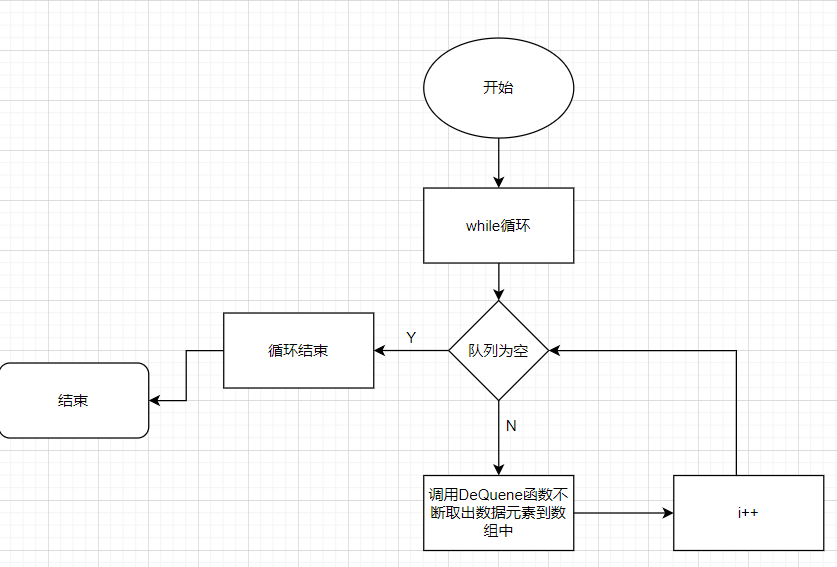
* DeQuene



* Gethead



* QueneToArray

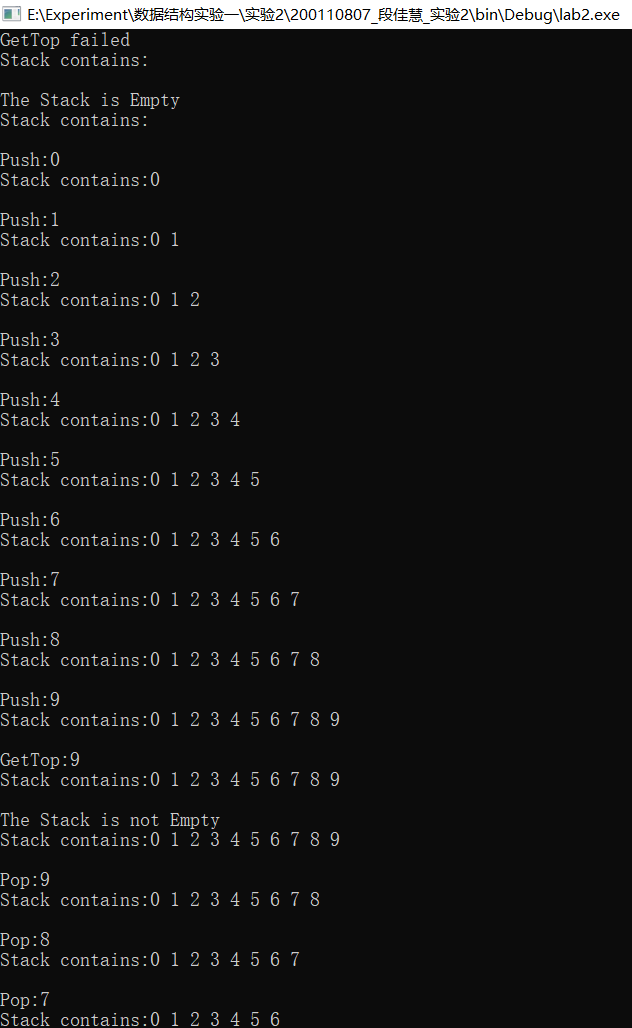


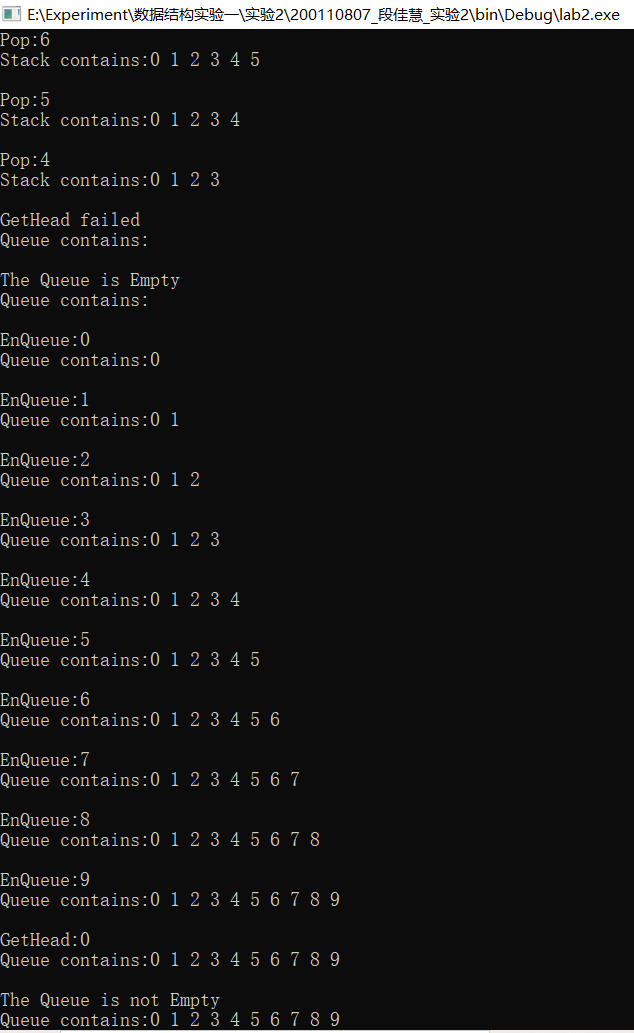
# 三、用户手册

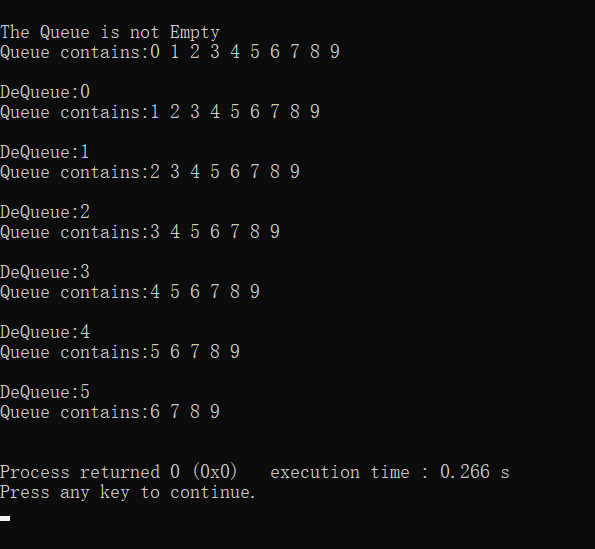
如：(1)输入数据的方式；(2)实现各种功能的操作方式等。

1. 计算机直接从目标文件中读取相关数据信息
2. 用户直接运行整个代码即可得到栈与队列的所有插入删除操作

# 四、结果







# 五、总结

该实验涉及到的数据结构和算法，以及遇到的问题和收获。

本此实验涉及的数据结构为栈与队列

在实验过程中遇到了几个问题，首先就是一开始对整个工程文件没有很清楚的认知，急于直接写代码，导致很多地方有疏漏，前后顾虑不周。最后还是要返回来看main.c里面的代码，有点浪费了时间。其次就是对于队列的理解不够，不清楚为什么要用到两个栈表来实现队列的基本操作导致刚上手写代码很多地方模棱两可，需要不停地翻看PPT和书本。

总的来说，这次实验使我对栈和队列的操作更加熟练，在以后的学习过程中，我要多多注意代码规范，追求简洁、细致的编程方式，并且要考虑边界与特殊情况，力求更进一步。