

## 实验 5 队列的建立与基本操作

### 一、实验目的

- (1) 通过实验掌握顺序循环队列的基本操作方法
- (2) 通过实验掌握链队列的基本操作方法

### 二、实验内容

- (1) 顺序循环队列的基本操作练习
- (2) 链队列的基本操作练习

### 三、实验步骤

1.完成循环队列的基本操作。

查看 SeqQueue.c 文件，补齐里面需要添加的代码：

```
60 void push(SeqQueue *q, DATATYPE data)
61 {
62     if(isFull(q)){
63         printf("队列已满，不能入队列\n");
64         return;
65     }
66     // 添加代码
67
68 }
69
70 DATATYPE pop(SeqQueue *q)
71 {
72     if(isEmpty(q)){
73         printf("队列为空，不能出队列，如果出队列元素值为： %d，表示是状态值----", ERROR);
74         return ERROR;
75     }
76     // 添加代码并修改返回值
77
78     return 0;
79 }
80
81 void print(SeqQueue *q)
82 {
83     // 添加代码
84
85     printf("\n");
86 }
```

```
81 void print(SeqQueue *q)
82 {
83     // 添加代码
84
85     printf("\n");
86 }
87
88 int isEmpty(SeqQueue *q)
89 {
90     return q->front == q->rear ? 1 : 0;
91 }
92
93 int isFull(SeqQueue *q)
94 {
95     //修改代码 (修改返回值)
96     return 0;
97 }
98
99 DATATYPE getFront(SeqQueue *q)
100 {
101     return q->a[q->front];
102 }
103
104 DATATYPE getRear(SeqQueue *q)
105 {
106     //修改代码 (修改返回值)
107     return 0;
108 }
109
110 int getLength(SeqQueue *q){
111     //添加或修改代码 (修改返回值)
112     return 0;
113 }
```

2.完成循环队列的基本操作。

查看 LinkQueue.c 文件，补齐里面需要添加的代码：

```
62 void push(LinkQueue *q, DATATYPE data)
63 {
64     // 添加代码
65 }
66
67
68 DATATYPE pop(LinkQueue *q)
69 {
70     if(isEmpty(q)){
71         printf("队列为空，不能出队列，如果出队列元素值为：%d，表示是状态值----", ERROR);
72         return ERROR;
73     }
74     // 添加代码并修改返回值
75     return 0;
76 }
77
78 void print(LinkQueue *q)
79 {
80     // 添加代码
81
82     printf("\n");
83 }
```

```
90 DATATYPE getFront(LinkQueue *q)
91 {
92     //修改代码（修改返回值）
93     return 0;
94 }
95
96 DATATYPE getRear(LinkQueue *q)
97 {
98     //修改代码（修改返回值）
99     return 0;
100 }
101
102 int getLength(LinkQueue *q){
103     //添加或修改代码（修改返回值）
104     return 0;
105 }
```

3.完成本次实验之后需要在云班课中提交 SeqQueue.c 和 LinkQueue.c 的文件。