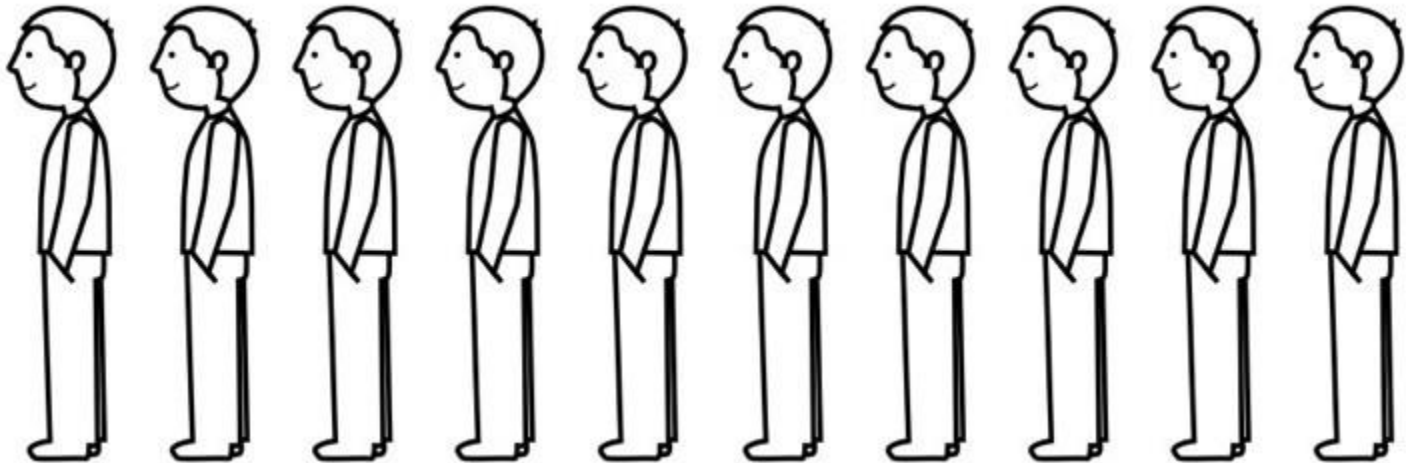


太戈编程
etiger.vip

数据结构

队列

queue





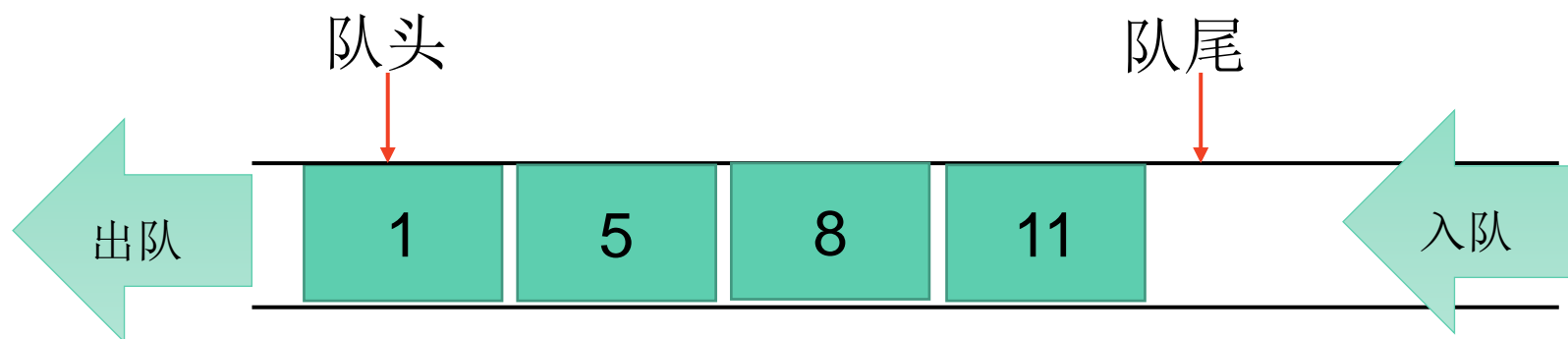
排队

插队可耻



队列的定义

队列是线性表只允许在一端进行**插入**操作，
而另一端进行**删除**操作



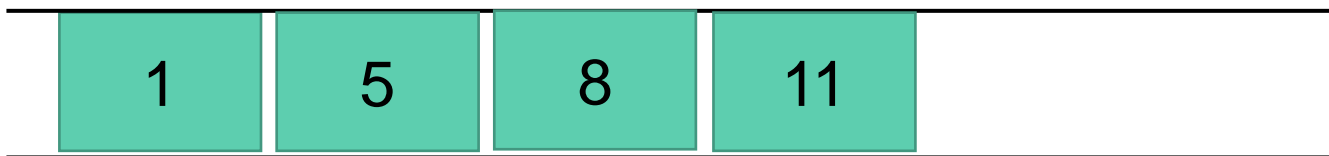
概念

空队列指不含任何数据元素的队列

允许插入（也称入队）的一端称为**队尾**，允许删除（也称出队）的一端称为**队头**

入队/出队

出队演示



特点

先进先出(First In First Out),简称FIFO。

后进后出(Last In Last Out),简称LILO。

插入操作：入队push

删除操作：出队pop

数组模拟队列

设定队列的最大容量为N

```
const int N=10009;
```

定义队列，用整数数组储存

```
int que[N];
```

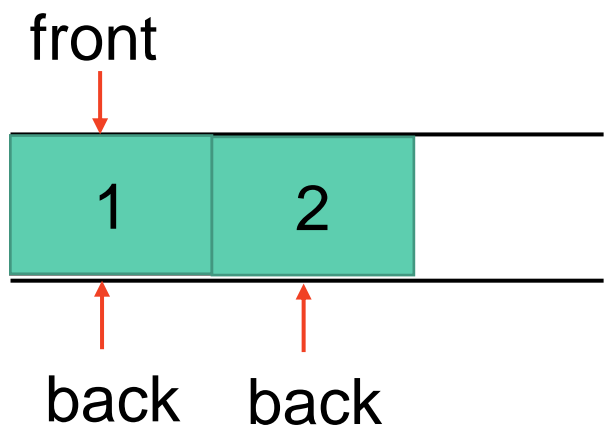
queue缩写

定义队头和队尾编号，初始为0

```
int front=0, back=0;
```

开始时元素
个数为零

入队



入队前判断队列是否已满

存入新入队列元素

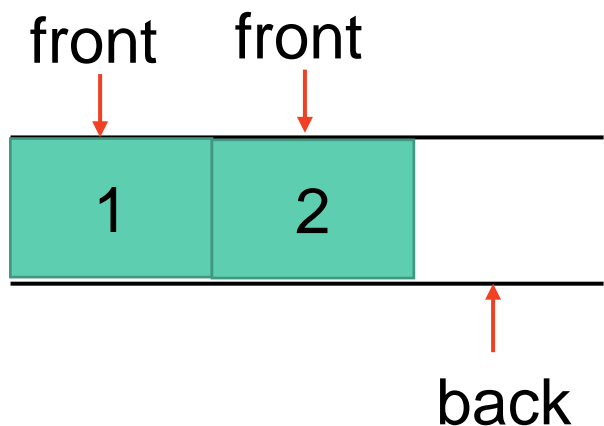
队尾向后移动一位

```
6 □ void push(int x){  
7     if(back==N)  
8         cout<<"queue is full"<<endl;  
9     else  
10        que[back++]=x;  
11 }
```

队尾处保持
没有元素

时间复杂度
 $O(1)$

出队



出队前判断队列是否为空

队头向后移动一位

移除队头元素

```
12 void pop(){  
13     if(front==back)  
14         cout<<"queue is empty"<<endl;  
15     else  
16         front++;  
17 }
```

时间复杂度
 $O(1)$

队列的基本操作

```
18 bool empty(){  
19     return front==back;  
20 }
```

判断空队列

首尾相同就是空

```
21 int size(){  
22     return back-front;  
23 }
```

求队列长度

```
24 int getFront(){  
25     return que[front];  
26 }
```

取队头元素

```
27 int getBack(){  
28     return que[back-1];  
29 }
```

取队尾元素

时间复杂度 $O(1)$

```
30 int main(){
31     push(1);
32     push(2);
33     push(3);
34     cout<<size()<<endl;
35     cout<<empty()<<endl;
36     cout<<getFront()<<endl;
37     cout<<getBack()<<endl;
38     pop();
39     cout<<size()<<endl;
40     cout<<getFront()<<endl;
41     pop();
42     cout<<size()<<endl;
43     cout<<getFront()<<endl;
44     pop();
45     cout<<size()<<endl;
46     cout<<empty()<<endl;
47     return 0;
48 }
```

请同学预测
输出结果

栈和队列 综合练习

判断出栈序列是否合法

入栈序列

1,2,3,4,5,6

1 2 3 4 5 6

出栈序列

2,3,4,1,6,5

2 3 4 1 6 5

栈

6

5

4

3

2

1

进

出

进

出

进

出

进

出

进

出

判断出栈序列是否合法

入栈序列

1,2,3,4,5,6

出栈序列

4,3,1,2,5,6

在草稿纸上画出入栈出栈序列

依次追踪入栈出栈序列变化，画出栈中元素变化

栈

判断出栈序列是否合法

入队顺序 $1, 2, 3, \dots, n$ 。按照出栈序列依次判断每个元素

如果出栈序列当前元素等于入栈序列中当前元素，通过入栈再出栈操作，可以得到该元素

如果出栈序列当前元素大于入栈序列当前元素，则将入栈序列的元素入栈暂存，直到栈顶元素等于出栈序列元素。栈顶出栈。

如果出栈序列当前元素小于入栈序列当前元素，且栈顶元素不等于出栈序列当前元素，说明无法得到该出栈序列

判断出栈序列是否合法

有六个元素1,2,3,4,5,6 依次顺序进栈，在进栈过程中可以有元素被弹出栈。哪些是合法的出栈序列，哪些是不可能的出栈序列

234165 1进 2进 2出 3进 3出 4进 4出 1出 5进 6进 6出 5出



324651

431256

543621

判断栈的最大容量

栈的初始状态为空，元素1,2,3,4,5,6依次入栈S，出栈的序列为2,4,3,6,5,1，则栈S的容量至少应该是多大

入栈序列

1,2,3,4,5,6

1 2 3 4 5 6

出栈序列

2,4,3,6,5,1

2 4 3 6 5 1

容量


3

栈


6
3
1

容量至少为3

现场挑战

(不定项选择，答案数不少于1个)对于入栈顺序为 a, b, c, d, e, f, g 的序列，下列  不可能是合法的出栈序列。

- A. a, b, c, d, e, f, g B. a, d, c, b, e, g, f
C. a, d, b, c, g, f, e D. g, f, e, d, c, b, a

今有一空栈 S，对下列待进栈的数据元素序列 a, b, c, d, e, f 依次进行进栈，进栈，出栈，进栈，进栈，出栈的操作，则此操作完成后，栈 S 的栈顶元素为  。

- A. f B. c C. a D. b

现场挑战

1023

请同学写出题目大意
已知什么求什么

限时2分钟

cnt表示什么含义?

```
4  int n,m,k,cnt=0,x;
5  cin>>n>>m;
6  cin>>k;
7  for(int i=0;i<k;i++) {
8      cin>>x;
9      if(x>0) cnt++;
10     else cnt--;
11     if(cnt>m) break;
12 }
13 if() cout<<"Error";
14 else cout<<cnt<<endl;
```

请同学跟着老师依次
翻译每一行写注释

现场挑战

1024

请同学写出题目大意
已知什么求什么

限时3分钟

请猜测变量含义

```
3  const int N=10009;  
4  int q1[N],q2[N];  
5  int f1=0;  
6  int b1=0;  
7  int f2=0;  
8  int b2=0;  
9  int n;  
10 int id=1;
```

f1为1号队列的队头编号

b1为1号队列的队尾编号

f2为2号队列的队头编号

b2为2号队列的队尾编号

id代表新入队的车号

请同学在电脑上完成前10行
以及翻译变量含义

```
12 cin>>n;
13 string s;
14 for(int i=0;i<n;i++){
15     cin>>s;
16     if(s=="arrive"){
17         if(b1-f1<=b2-f2)
18             q1[b1++]=id++;
19         else
20             
21     }else if(s=="leave1")
22         f1++;
23     else
24         
25 }
```

如果1号队列内车辆数
少于等于2号队列

请同学跟着老师依次
翻译每一行写注释

```
26  if(f1==b1)
27      cout<<0<<endl;
28  else{
29      for(int i=f1;i<=b1-2;i++)
30          cout<<q1[i]<<" ";
31      cout<<q1[b1-1]<<endl;
32  }
33  if(f2==b2)
34      cout<<0<<endl;
35  else{
36
37
38
39  }
```

注意：行末不能有空格

请同学跟着老师依次
翻译每一行写注释

太戈编程

1023

1024

拓展题

25, 32, 33