

## 顺序队列基本操作的 C++ 代码

### 顺序队列

#### 顺序队列类 AQueue 的类实现

// 构造函数

```
template< class T >
```

```
AQueue<T>::AQueue ( int MaxQueueSize ) {
```

```
    Size = MaxQueueSize ;
```

```
    QArray = new T[MaxQueueSize] ;
```

```
    front = 0 ; rear = 0 ; count = 0 ;
```

```
}
```

// 将元素 item 插入队尾

```
template< class T >
```

```
bool AQueue<T>::QInsert ( const T& item ){
```

```
    if ( isFull () ) { cout <<“Inserting into a full Queue!”<<endl ; return false ; }// 若队列
```

已满

```
        QArray[rear] = item ;           // 添加队尾元素
```

```
        rear = ( rear + 1 ) % Size ;    // 修改队尾指针
```

```
        count ++ ;                     // 队列长度加 1
```

```
        return true ;
```

```
}
```

// 删除队首元素，并将其元素值赋给变量 item

```
template< class T >
```

```
bool AQueue<T>::QDelete ( T & item ) {
```

```
    if ( isEmpty () ) { cout<<“Deleting from an empty queue!”<<endl ; return false ; }
```

```
    item = QArray[front] ;
```

```
    front = ( front + 1 ) % Size ;      // front 顺时针移动一格
```

```
    count -- ;                         // 队列长度减 1
```

```
    return true ;
```

```
}
```

// 将队首元素值赋给变量 item

```
template< class T >
```

```
bool AQueue<T>::QFront ( T & item ) {
```

```
    if ( isEmpty () ) { cout<<“Reading from an empty queue!”<<endl ; return false ; }
```

```
    item = QArray[front] ;
```

```
    return true ;
```

```
}
```