

C++ 容器类 <queue>

C++ 标准库中的 <queue> 头文件提供了队列（Queue）数据结构的实现。队列是一种先进先出（FIFO, First In First Out）的数据结构，它允许在一端添加元素（称为队尾），并在另一端移除元素（称为队首）。

队列是一种线性数据结构，它遵循以下规则：

元素只能从队尾添加。

元素只能从队首移除。

语法

在 C++ 中，队列的语法如下：

```
#include <queue>

// 声明队列
std::queue<Type> q;
```

这里 Type 是队列中存储元素的数据类型。

常用操作

队列提供了以下常用操作：

empty(): 检查队列是否为空。

size(): 返回队列中的元素数量。

front(): 返回队首元素的引用。

back(): 返回队尾元素的引用。

push(): 在队尾添加一个元素。

pop(): 移除队首元素。

实例

下面是一个使用 C++ 标准库 <queue> 的简单实例：

实例

```
#include <iostream>
#include <queue>

int main() {
    // 创建一个整数队列
    std::queue<int> q;

    // 向队列中添加元素
    q.push(10);
```

```
q.push(20);
q.push(30);

// 打印队列中的元素数量
std::cout << "队列中的元素数量: " << q.size() << std::endl;

// 打印队首元素
std::cout << "队首元素: " << q.front() << std::endl;

// 打印队尾元素
std::cout << "队尾元素: " << q.back() << std::endl;

// 移除队首元素
q.pop();
std::cout << "移除队首元素后, 队首元素: " << q.front() << std::endl;

// 再次打印队列中的元素数量
std::cout << "队列中的元素数量: " << q.size() << std::endl;

return 0;
}
```

输出结果:

```
队列中的元素数量: 3
队首元素: 10
队尾元素: 30
移除队首元素后, 队首元素: 20
队列中的元素数量: 2
```

C++ 的 `<queue>` 标准库提供了一种方便的方式来实现队列数据结构。通过使用队列，我们可以有效地管理需要按照特定顺序处理的元素集合。希望这篇文章能帮助初学者更好地理解和使用 C++ 中的队列。

注意事项

队列不允许随机访问元素，即不能直接通过索引访问队列中的元素。

队列的实现通常使用链表或动态数组，这取决于具体的实现。