

C++知识点总结(32): STL (vector)

原创

AICodeThunder

于 2024-05-05 16:33:13 发布

阅读量415


收藏 7

点赞数 3

分类专栏: C++知识点总结

文章标签: c++ java 开发语言

版权

 C++知识点总结 专栏收录该内容

7 订阅 53 篇文章

订阅专栏

动态数组 vector

- 一、概念
1. 意义

2. 优点

3. 一维动态数组

(1) 定义

(2) 功能函数

(3) 注意事项

4. 二维动态数组

(1) 二维静态数组的局限

(2) 二维动态数组操作

a. 定义

b. 初始化

5. 迭代器

(1) 概念

(2) 定义

(3) 遍历

(4) 功能函数
- 二、例题
1. 命令列表

(1) 审题

(2) 参考答案

2. 借阅表格

(1) 审题

 AICodeThunder

关注

3

7

0

专栏目录

(2) 参考答案

a. 普通写法

b. 结构体

3. 物品档案柜

(1) 审题

(2) 参考答案

一、概念

1. 意义

vector 翻译为向量，一般说成动态数组。

在插入数据或者新增数据时，数组会动态的拓展长度，即“长度根据需要而自动改变的数组”，整个过程无需人工干预，也不需要实现固定长度。

2. 优点

- 动态
- 随机访问
- 插入删除方便

3. 一维动态数组

(1) 定义

格式：`vector<数据类型>动态数组名;`

```
1 | #include <vector>
2 | vector<int>vec;
```

(2) 功能函数



AI Code Thunder

关注



3



7



0



专栏目录

方法	功能	常用程度
<code>.push_back(x)</code>	在尾部增加一个元素 x	★★★★
<code>.pop_back()</code>	删除最后一个元素	★
<code>.front()</code>	返回首元素	★
<code>.back()</code>	返回尾数组	★
<code>.size()</code>	返回元素个数	★★★★
<code>.empty()</code>	判断是否为空	★★★★
<code>.resize(n)</code>	改变实际大小变为 n	★★
<code>.begin()</code>	返回指向第一个元素的迭代器	★
<code>.end()</code>	返回指向最后一个元素的下一个位置的迭代器	★
<code>.erase()</code>	删除迭代器指向元素	★★
<code>.clear()</code>	清空所有元素	★★★★
<code>.at(pos)</code>	返回 pos 位置元素的值	★
<code>.max_size()</code>	返回最大可允许的元素数量值	★

(3) 注意事项

- `.resize(n)` 如果 n 比原来的实际大小更小，那么只会留下前面的 n 个元素。
- `.at(pos)` 如果越界会产生提示报错。

4. 二维动态数组

(1) 二维静态数组的局限

- 大小固定：在编译时就需要确定，并且无法在运行时改变。如果数组大小超出了预设的限制，就无法存储更多的数据。
- 内存浪费：在编译时就需要分配内存空间，使用的空间如果没有占满，内存就会造成极大浪费。

(2) 二维动态数组操作



AI Code Thunder

关注



3



7



0



专栏目录

a. 定义

```
1 // 方法一: vector的数组
2 vector<int> a[105];
3
4 // 方法二: vector的vector
5 vector <vector<int> > a;
6 a.resize(105);
```

b. 初始化

```
1 // 方法一: vector的vector
2 vector<vector<int> > vec={{1,2},{3,4},{5,6}};
3
4 // 方法二: vector的vector数组
5 vector<vector<int> > vec(4, vector<int>(5));
6
7 // 方法三: 全部初始化为0
8 vector<vector<int> > vec(4, vector<int>(5,0));
```

5. 迭代器

(1) 概念

迭代器的作用和指针类似, 可以通过引用 (*) 操作访问其指向的元素内容。
常用的容器 (例如 `map`, `set`, `vector` 等) 都可以使用一对迭代器来表示范围。

(2) 定义

```
1 #include <vector>
2 vector<int>::iterator it;
```

(3) 遍历

```
1 for (it = vec.begin(); it != vec.end(); it++)
2 {
3     cout << *it << " ";
4 }
```



AI Code Thunder

关注



3



7



0



专栏目录

(4) 功能函数

方法	功能	常用程度
<code>.insert(it, x)</code>	迭代器 <code>it</code> 指向的元素前添加一个元素 <code>x</code>	☆☆☆
<code>.erase(it)</code>	删除迭代器 <code>it</code> 指向的元素	☆☆☆
<code>.begin()</code>	返回指向第一个元素的迭代器	☆☆☆
<code>.end()</code>	返回指向最后一个元素的下一个位置的迭代器	☆☆☆
<code>reverse(l, r+1)</code>	翻转 <code>l</code> 到 <code>r</code> 范围的元素	☆☆
<code>.insert(it, n, x)</code>	迭代器 <code>it</code> 指向元素前增加 <code>n</code> 个相同的元素 <code>x</code>	☆☆
<code>.insert(it, l, r+1)</code>	迭代器 <code>it</code> 指向元素前插入另一个相同类型向量的 <code>l</code> 到 <code>r</code> 之间的数据	☆☆
<code>.erase(l, r+1)</code>	删除 <code>l</code> 到 <code>r</code> 范围的元素	☆☆
<code>.rbegin()</code>	反向迭代器，指向最后一个元素	☆
<code>.rend()</code>	反向迭代器，指向第一个元素之前的位置	☆

二、例题

1. 命令列表

(1) 审题

题目描述

逛公园的时候，你捡到一个神奇的对讲机，这个对讲机有一个命令列表，列表上的命令分别有三种操作：

- a) `increase`，表示向列表的最后面添加一个数字（列表中的元素唯一）；
- b) `remove`，表示删除列表的第一个元素；
- c) `least`，表示删除列表中的数字中值最小的那一个。

输入描述

第一行会给出一个数字 `N`，表示你会执行



AI Code Thunder

关注



3



7



0



专栏目录