

作业十二

概念题

1. STL中存在哪些主要容器？它们的应用场景分别是什么？

- 1 | vector：动态数组，适用于需要快速随机访问和可能需要动态扩展大小的场景。
- 2 | deque：双端队列，提供在队列两端快速插入和删除操作的能力，适用于需要从两端高效操作的场景。
- 3 | list：双向链表，提供快速的插入和删除操作，适用于频繁插入和删除的场景。
- 4 | array：固定大小的数组，适用于需要固定大小连续内存块的场景。
- 5 | stack：后进先出（LIFO）的栈，通常基于deque或vector实现，适用于需要栈操作的场景。
- 6 | queue：先进先出（FIFO）的队列，通常基于deque实现，适用于需要队列操作的场景。

2. 什么是迭代器？STL中基本的五种迭代器类型以及各自的可以进行的操作是什么？

- 1 | 输入迭代器：
- 2 | 可以读取元素的值。
- 3 | 可以进行前移操作（++）。
- 4 | 不能写入元素的值。
- 5 | 不能比较两个迭代器是否相等或不等。
- 6 |
- 7 | 输出迭代器：
- 8 | 可以写入元素的值。
- 9 | 可以进行前移操作（++）。
- 10 | 不能读取元素的值。
- 11 | 不能比较两个迭代器是否相等或不等。
- 12 |
- 13 | 正向迭代器：
- 14 | 可以读取元素的值。
- 15 | 可以写入元素的值。
- 16 | 可以进行前移操作（++）。
- 17 | 可以比较两个迭代器是否相等或不等。
- 18 |
- 19 | 双向迭代器：
- 20 | 拥有正向迭代器的所有特性。
- 21 | 可以进行后移操作（--）。
- 22 |
- 23 | 随机访问迭代器：
- 24 | 拥有双向迭代器的所有特性。
- 25 | 支持随机访问，可以通过下标直接访问元素。
- 26 | 支持与下标的比较。
- 27 | 支持减法操作，可以计算两个迭代器之间的距离。

3. 上述主要容器类型与迭代器之间的关联关系如何？

- 1 | vector 提供随机访问迭代器，因为它的元素存储在连续的内存块中，允许快速的随机访问。
- 2 | list 提供双向迭代器，因为它是基于链表实现的，元素不是连续存储的，但可以高效地进行插入和删除操作。

4. 如何理解STL中迭代器的作用？

1. 所有容器都提供了迭代器，可以使用相同的方法来遍历不同类型的容器。这提高了代码的可重用性和可读性。
2. 迭代器可以针对特定类型的容器进行优化，以提供更高效的数据访问方式。例如，对于基于连续内存的容器（如 `std::vector`），迭代器可以利用内存的局部性原理来提高缓存利用率。
3. 迭代器使得容器和算法之间的耦合度降低，算法不依赖于特定类型的容器，而是依赖于迭代器提供的接口。

编程题

请完成课程网站上高级程序设计24春-作业十二-编程题

提交注意事项

截止时间：2024-5-28 23:59

文件格式：姓名-学号.pdf

提交方式：南大计科在线实验教学平台

请同学们于截止时间前在南大计科在线实验教学平台上提交，每次作业最终只需要提交一个pdf文件即可，以“姓名-学号.pdf”的方式命名。

注意：请按要求命名文件，并且只提交一个PDF文件。任何错误的命名和文件格式将影响你的作业得分。