**15 简答题3**

**说说你常用的stl库，它们有什么优缺点。**

答：

1.向量（std::vector）:

优点:

相当于一个动态数组，可以自动调整大小。

具有随机访问元素的能力。

与数组相比更灵活，易于扩展和收缩。

缺点:

插入或删除非尾部元素相对较慢。

动态增长可能导致内存重新分配和元素复制，影响效率。

2.双端数组 (std::deque)

可以对头端进行插入删除操作

与vector区别：

vector对于头部的插入删除效率低，数据量越大，效率越低

deque相对而言，对头部的插入删除速度回比vector快

vector访问元素时的速度会比deque快,这和两者内部实现有关

3.列表（std::list）:

提供双向链表的功能，易于插入和删除元素。但不支持随机访问，访问元素的时间复杂度为O(n)。相比向量，由于要存储指向下一结点的指针，内存占用更多。

4.队列（std::queue） 和 栈（std::stack）:

简单易用，只暴露了队列中后进先出和栈中先进先出的相关接口。

但功能相对受限。

5.集合（std::set） :

底层红黑树实现，提供自动排序功能。快速查找，插入和删除（大多数操作的时间复杂度为O(log n)）。但相比哈希表（如std::unordered\_map），性能可能稍差。

6.映射（std::map）:

提供了key-value的键值关联，可以根据key值快速找到value值。

7.无序集合（std::unordered\_set）和无序映射（std::unordered\_map）:

优点:

基于哈希表，提供平均时间复杂度为O(1)的快速查找。暴力解题必备。

缺点:

相比有序集合和映射，内存使用效率较低。

8.算法（如 std::sort, std::find 等）:

提供了查找，拷贝，排序，集合等算法。比如可以用std::count\_if进行条件查找，大大提高了代码编写效率。但有时候这些函数在某些特定场景下并不是最优解，需要自己动手优化。