

STL中vector和list怎么实现的

vector

是动态数组，在堆中分配内存，元素连续存放，有保留内存，如果减少大小后，内存也不会释放；如果新增大小比当前大时才会重新分配内存

扩容操作

如果集合已满，在新增数据的时候，就要分配一块更大的内存，将原来的数据复制过来，释放之前的内存，在插入新增的元素。

所以对vector的任何操作，一旦引起空间重新配置，指向原vector的所有迭代器就都失效了；

为什么vector扩容一般都是1.5到2倍

VS2015中以1.5倍扩容，GCC以2倍扩容。

使用成倍扩容，可以保证常数的时间复杂度，而增加固定大小的容量只能达到 $O(n)$ 的时间复杂度

以2倍的方式扩容，导致下一次申请的内存必然大于之前分配内存的总和，导致之前分配的内存不能再被使用，所以最好倍增长因子设置为(1,2)之间：

list（双向链表）

元素存放在堆中，每个元素都是放在一块内存中，内存地址可以不连续，通过指针访问所以删除插入很快，但是随机访问很不方便，没有对[]重载

常见操作时间复杂度对比

vector插入、查找、删除时间复杂度分别为： $O(n)$ 、 $O(1)$ 、 $O(n)$

list插入、查找、删除时间复杂度分别为： $O(1)$ 、 $O(n)$ 、 $O(1)$

应用场景

vector适用于不在乎插入、删除的效率，需要高效的随机访问

set适用于高效的删除、插入，而不关心随机访问