C++ 标准模板库

容器名	名称	数据结构	性能	备注
string	通用字符串库	连续存放的内存块、 有保留内存 堆中分配内存	高效率的随机访问； O（1）的访问时间； 在最后增加元素时，一般不需要分配内存空间，速度快； 在中间或开始操作元素时要进行内存拷贝效率低； 支持[]操作;	
Vector	通用向量（数组）库	变长一维数组； 连续存放的内存块； 有保留内存； 堆中分配内存；	高效率的随机访问； O（1）的访问时间 在最后增加元素时，一般不需要分配内存空间，速度快； 在中间或开始操作元素时要进行内存拷贝效率低; 支持[]操作;	需要高效的随即存取，而不在乎插入和删除使用 vector
list	通用链表库	双向链表； 内存空间上可能是不连续; 无保留内存 堆中分配内存；	随机存储需要遍历指针所以效率低; 很高的效率进行任意地方删除插入; 开始和结尾元素的访问时间快,其它元素都 O（n）；	大量的插入和删除，而不关系随即存取使用 list
deque		双端队列; 在堆上分配内存; 个堆保存几个元素，而堆之间使用指针连接;	支持[]操作; 像是 list 和 vector 的结合;	关心插入和删除并关心随即存取折中使用 deque
set	通用集合库	集合; 无序的保存元素		
map	通用字典库	平衡二叉树; 一个值映射成另一个值	查找某一值是常数时间 O(1); 每次插入值的时候，会重新构成底层的平衡二叉树，效率有一定影响	创建字典时使用

在 STL 中基本容器有：string、vector、list、deque、set、map

Set 和 map 都是无序的保存元素，只能通过它提供的接口对里面的元素进行访问

Set:集合，用来判断某一个元素是不是在一个组里面，一般此容器使用的比较少。

Map:映射，相当于字典，把一个值映射成另一个值，如果想创建字典的话使用它是最好的，底层采用的是树型结构，多数使用平衡二叉树实现，查找某一值是常数时间，遍历起来效果也不错，只是每次插入值的时候，会重新构成底层的平衡二叉树，效率有一定影响。

string、vector、list、deque、set 是有序容器；

1. string

名为：通用字符串库

string 是一个处理字符串数据的类，支持变长字符串，支持随机存取

string 是 basic_string<char>实现，在内存中是连续存放的，为了提高效率，都会有保留内存，会在一开始分配一个大小于内容的内存，当内容增加时不用再分配新的内存，只有当已分内存放不下时。再分配新的内存空间。

对 string 的操作，如果是添加到最后时，一般不需要分配新的内存，所以性能最快；

如果对中间或是开始部分操作，如往那里添加元素或是删除元素，或是代替元素，这时需要进行内存复制，性能会降低。

当删除元素时，并不会立刻释放它已经分配的内存，为了是下次使用时可以高效。

由于保留内存，所以大量使用时会有内存浪费。

在堆中分配内存；

2. vector

名为：通用向量（数组）库

Std:vector, 是一个变长一维数组模板类,同内建数组一样，支持随机存取。、

实际上就是一个动态数组，随机存取任何元素都能在常数时间完成，在尾端增删元素具有较佳的性能。

元素连续存放，同样有保留内存，如果减少大小后内存不会释放，当新值大于当前内存大小时才会再分配内存；

其拥有一段连续内存，并且起始地址始终不变，因此能非常好的支持随即存取，即数组的[]操作符，但由于它的内存空间是连续的，所以在中间进行插入和删除会造成内存块的拷贝，另外，当该数组后的内存空间不够时需要重新申请一块足够大的内存进行内存拷贝，大大影响了效率。

同 string 一样，对最后元素的操作是最快的，此时一般不需要移动内存，只有保留内存不够时才需要分配内存进行内存拷贝。

对中间和开始处进行添加删除元素操作需要移动内存。如果元素是结构和类，那移动的同时还会进行构造和析构操作，所以性能不高（此时以存储类对像的指针为好）；

访问方面，由于是连续内存的存储。所以对任何元素的访问都是 O（1），即常数时间完成。

同 string 一样可以使用 capacity 看当前保留的内存，使用 swap 来减少它使用的内存。

同样在堆上分配内存；

随机访问效率很高

3. <list>

名为：通用链表库

list 是一个通用的双向链表模板类，支持任意位置的数据读写，插入和删除操作。

对 list 的访问主要通过迭代器实现。

由于是链表存储所以内存空间上可以是不连续的，通过指针进行数据的访问，随机存储需要遍历指针所以效率很低。但链表的特点，它可以很高的效率进行任意地方的删除和插入。

没有内存空间预留，所以每次增加元素都会从内存中分配空间，每次删除都会释放它占用的内存。

因为使用链表存储指针的方式，所以添加删除元素时不存在内存拷贝的问题，也没有对每类对像元素进行构造析构的问题，所以常作随机操作容器。

但就访问来说。因为双向链表，只有开始和结尾元素的访问时间快。其它元素都 O（n）

元素在堆上分配；

如果添加删除大内存元素用 list 比较好。对于 list<指针>是性能最低下的做法；

54 deque

Deque 是一个双端队列，是一个个的堆。每个堆保存几个元素，而堆之间使用指针连接，像是 list 和 vector 的结合，支持[]操作，也支持随机存取，可以有较高的随机访问速度。

在堆上分配内存；

5. <set>

名为：通用集合库

set 或 multiset 是一个按用户指定排序规则将 set 内的数据进行排序的集合库，set 内不允许有相同的数据，而 multiset 允许有重复数据。

6.<map>

名为：通用字典库

map 或 multimap 是一个字典模板类，其索引可以是任意用户指定类型。且能按用户指定排序规则将 map 内的数据进行排序的集合库，map 内不允许有相同的数据，而 multimap 允许有重复数据。