//习题10-3 建立一个向量容器的实例s，不断对s调用push\_back向其中增加新的元素

//观察在此过程中s.capacity()的变化

//习题10-5 约瑟夫问题：n个骑士编号1, 2, ..., n, 围坐在圆桌旁

//编号为1的骑士从1开始报数，报到m的骑士出列，然后下一个位置再从1开始报数

//找出最后留在圆桌旁的骑士编号

//(1) 编写一个函数模板。以一种顺序容器的类型作为模板参数

//    在模板中使用指定类型的顺序容器求解约瑟夫问题，m, n是该函数模板的形参

//(2) 分别以vector<int>, deque<int>, list<int>作为类型参数调用该函数模板

//    调用时将n设为较大的数，将m设为较小的数(例如令n=10000, m=5)

//    观察3种情况下的调用该函数模板所需花费的时间

//习题10-6 编写一个具有以下原型的函数模板

//template<class T>

//void exchange(list<T>& l1, list<T>::iterator p1, list<T>& l2, list<T>::iterator p2);

//该模板用于将l1链表的[p1, l1.end())区间和l2链表[p2, l2.end())区间的内容交换

//在主函数中调用该模板，以测试该模板的正确性

//习题10-7 分别对stack<int>, queue<int>, priority\_queue<int>的实例执行下面的操作:

//调用push函数分别将5, 1, 4, 6压入;

//调用两次pop函数;

//调用push函数分别将2, 3压入;

//调用两次pop函数。

//请问对于三类容器适配器，每次调用pop函数时弹出的元素分别是什么?

//请编写程序验证自己的判断

//习题10-8 编写一个程序，从键盘输入一个个单词后

//输出该单词是否曾经出现过以及出现次数

//可以尝试分别用多重集合(multiset)和映射(map)两种途径实现

//将二者进行比较

//习题10-9 编写程序对比STL中的3个元素交换函数

//swap, iter\_swap和swap\_ranges对数组中的元素进行的交换操作

//习题10-10 编写一个程序，从键盘输入两组整数(可以看作是两个集合)

//分别输出同属于两组的整数(即两个集合的交集)

//属于至少一组的整数(即两个集合的并集)

//属于第一组但不属于第二组的整数(即两个集合的差集)

//程序中需要用到set\_intersection, set\_union, set\_difference等算法

//习题10-11 下面的程序段首先构造了一个元素按升序排列的向量容器s

//然后试图调用unique算法去掉其中的重复元素，并将结果输出

//int arr[] = {1, 1, 4, 4, 5};

//vector<int> s(arr, arr+5);

//unique(s.begin(), s.end());

//copy(s.begin(), s.end(), ostream\_iterator<int>(cout, "\n"));

//(1) 以上的输出结果是什么？是否真正达到了去除重复元素的目的？

//    如果未达到目的，应如何对程序进行修改？

//(2) 如果s是列表，是否有更方便高效的方法？

//习题10-12 编写一个产生器，用来产生0~9范围内的随机数

//建立一个顺序容器，使用该产生器和generate算法为该容器的元素赋值

//习题10-13 编写一个二元函数对象，用来计算x的y次方，其中x和y都是整数

//利用该函数对象和transform算法，并结合适当的函数适配器

//对于10-12所生成的整数序列中的每个元素n, 分别输出5^n, n^7和n^n

10-14 为例9-3的Array类增加一个成员函数swap

10-15 对例10-25中的IncrementIterator进行扩充，使它成为一个随机访问迭代器

10-16 对例10-26中的mySort算法进一步改进，使得当传入的第三个参数为随机访问迭代器时，直接在输出的区间中进行排序，避免使用s作为中转，从而节省时间和空间

//习题10-17 登陆Boost官方网站，下载最新的Boost程序库，阅读文档，完成以下任务

//(1) 修改例10-3，将主函数中的静态数组a变为array<double, 5>类型的对象

//    对与a相关的代码进行相应修改，比较修改前后的程序

//(2) 修改例10-17，使用Bind程序库的bind代替bind2nd

//(3) 用另一种方式修改例10-17，

//    使用Lambda程序库的相关功能来生成判断一个数是否大于40的函数对象

//(4) 改用unordered\_map实现例10-10的功能