**三，字符串，向量和数组**

1，命名空间的using声明

using声明具有如下的形式：using namespace::name;

头文件不应包含using声明。

2，标准库类型string

标准库类型string表示可变长的字符序列。

参考文档：初始化string对象的方式。

如果使用=号初始化一个变量，执行的是拷贝初始化，不使用等号，则执行的是直接初始化。

string.size()返回的类型为string::size\_type。这是一个无符号数，不可与带符号数混用。

字符串字面值与string是不同的类型，连接字符串时候必须确保每个加法运算符的两侧的运算对象至少有一个string。

可以使用下标访问string对象。也可以进行赋值。

3，标准库类型vector

引用不是对象，所以不存在包含引用的Vector。

4，迭代器

end表示尾元素的下一个位置。如果容器为空，begin和end返回相同。

如果对象是常量，begin和end返回const\_iterator；如果对象不是常量，返回iterator。

如果对象只需读操作而无须写操作的话最好使用常量类型，为了便于得到const\_iterator类型，可以使用cbegin和cend函数。

迭代器解引用：it->mem和(\*it).mem表达的意思相同。

5，数组

不能将数组的内容拷贝给其他数组作为其初始值，也不能用数组为其他数组赋值。

使用数组时，编译器一般会把它转换成一个指向数组首元素的指针。

获取数组的首尾：begin(arr)和end(arr)。