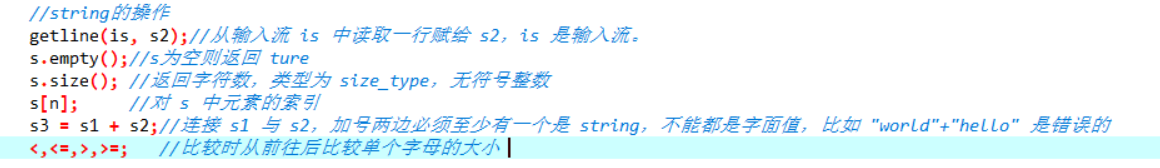
string类型：变长的字符序列，即字符数组，最后一个位置上是‘\0’

string的基本操作（string头文件）：

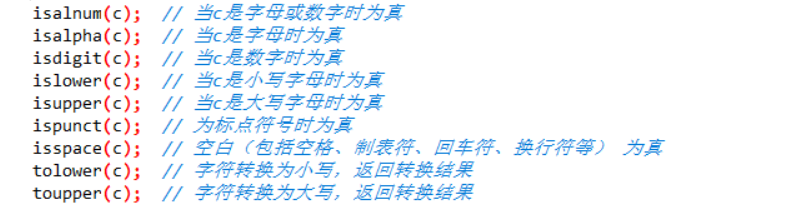


ps：

1.string的读取不能使用cin，因为cin会读取输入流直到空格；而getline() 读取输入流的内容直到遇到换行符停止

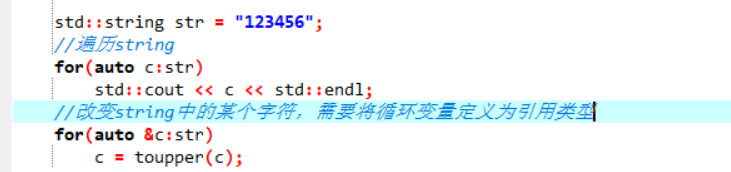
2.使用size\_type而不用unsign int是为了适应不同平台，因为其具体类型依赖于具体实现，这样可以根据平台不同使用不同的类型

对string中字符的操作（cctype头文件）：



ps：输入必须是单个字符，即ascii码在0-127或0-255区间

遍历修改字符串时，循环变量必须是char&类型

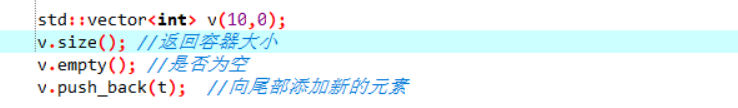


STL标准库的vector类型：

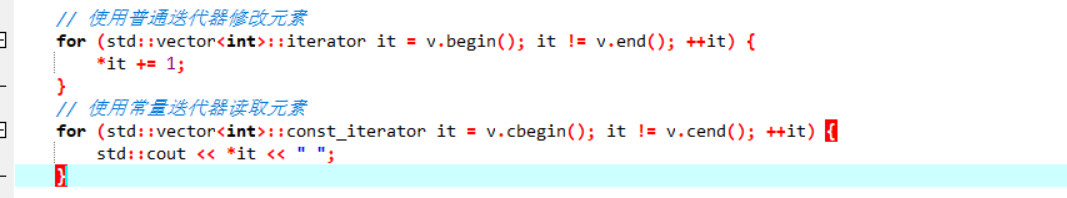
形式为vector<T>，vector为模板，T为其中存储对象的类型

特点：高速增长，向其中添加元素的速度很快。除非在初始化时就知道vector容器的大小并且其中每个元素的初始值都一样，否则定义一个空的vector再向其中添加元素会更高效。

基本操作：



ps：所有标准库容器都可以使用for循环遍历和迭代器遍历。

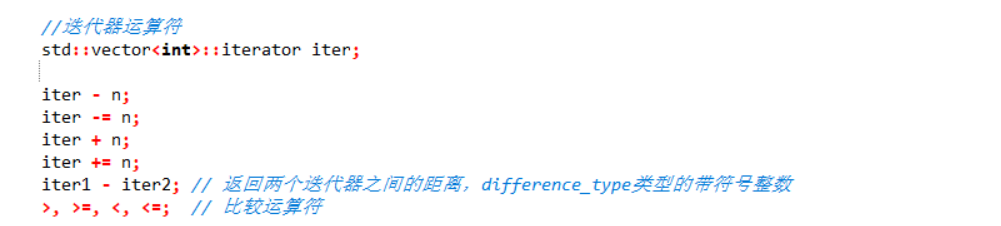


ps：

1.普通迭代器允许修改元素，而常量迭代器只允许读取元素

2.end指向“尾后元素”，即容器最后一个元素的下一个位置。

迭代器支持的运算符：



其中迭代器的加减n是当前容器中向前或向后移n个位置

数组：

1.数组大小确定不变，数组的大小需要在编译时确定，因此必须是一个常量表达式或常量

2.数组的元素必须是对象，不可以是引用（同vectore）

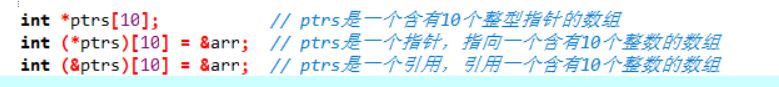
3.数组不能用auto定义。

4.与vector不同，不可以使用数组给另一个数组直接赋值或拷贝，如果需要拷贝数组，需要遍历所有元素逐一拷贝。

ps：如果是多维数组，遍历时除了最里面一层循环变量可以不为引用之外，外层的循环遍历都必须为引用类型

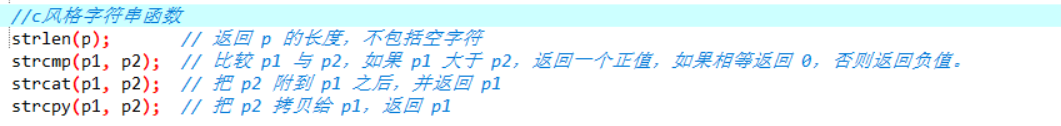
数组的类型：

根据括号按照由内向外的顺序理解数组的类型，数组的名字，例如下面的ptrs，实际上是数组第一个下标的地址，后面中括号中的下标实际上是偏移量，\*和&的运算优先级低于“[]”



C风格的字符串：将字符串放在字符数组中并以“\0”（空字符）结尾

可以使用字符数组初始化string，反过来不可以



尽量不要使用C风格字符串函数，因为这些函数不验证参数。传入参数的指针必须指向以空字符结束的数组。必须确保数组足够大