

C++标准程序库读书笔记

strings字符串

对string的定义是：C++标准程序库中某个字符串型别的对象，置于一般字符串，char* 或者const char*，我用的术语是C—string.

凡是以一个索引和一个长度作为参数的地方，字符串行为遵循下面两项规则：

- 索引值必须合法。
- 字符数量可为任意值。

```
namespace std
{
    template<class charT,class traits = char_traits<charT>,class
Allocator = allocator<char T> >
    class basic_string;
```

字符串操作函数的参数规格

参数	含义
const string & str	整个str字符串
const string & str	大部分情况下是指字符串str中以idx开始的num个字符
size_type idx	
size_type num	
const char* cstr	整个c-string cstr
const char* chars	字符数组chars的len个字符
size_type len	
char c	
size_type num,char c	num个字符
iterator beg iterator end	区间【beg:end】内所有字符

有三个函数可以将字符串内容转换成字符数组或c-string:

- data() 由于并未追加“\0”字符，所以返回型别并非有效的C-string
- c_str() 以C-string形式返回字符串内容，也就是在尾端添加“\0”字符
- copy() 将字符串内容复制到调用者提供的字符数组中，不添加“\0”字符

让string拥有足够的容量是很重要的，原因有二：

- 重新分配会造成所有指向string的references,pointers和iterators失效。
- 重新分配很耗时间。

以下操作会使指向字符的references和pointers失效：

- 以swap()交换两值
- 以operator>>()或getline()读入新值
- 以data()或c_str()输出内容
- 调用operator[].at() begin() rbegin() end() rend()-之外的任何non-const成员函数
- 调用任何函数并于其后跟着operator[] at() begin()rbegin() end() rend()

string和vector的差异

- vector首要目标是处理和操作容器内的元素，而非容器整体，因此实现时通常会为容器元素的操作行为进行优化
- string主要是把整个容器视为整体，进行处理和操作，因此实现时通常会为整个容器的赋值和传递进行优化。

string对迭代器的支持

string迭代器是random access迭代器，也就是说它支持随机存取。