String基本概念



本质：

string是C++风格的字符串，而string本质上是一个类

string和char\*的区别

char\*是一个指针

string是一个类，类内部封装了char\*，管理这个字符串，是一个char\*的容器。

特点：

string类内部封装了很多成员方法

例如：查找find，拷贝copy，删除delete，替换replace，插入insert

string管理char\*所分配的内存，不用担心复制越界和取值越界等，由类内部进行负责

string构造函数

string(); //创建一个空的字符串，例如：string str;

string(const char\* s); //使用字符串s初始化

string(const string& str); //使用一个string对象初始化另一个string对象

string(int n,char c); //使用n个字符c初始化

string赋值操作

功能描述：

给string字符串进行赋值

赋值的函数原型

string& operator=(const char\* s); //char\*类型字符串赋值给当前的字符串

string& operator=(const string& s); //把字符串s赋给当前的字符串

string& operator=(char c); //字符赋值给当前的字符串

string& assign(const char\* s); //把字符串s赋给当前的字符串

string& assign(const char\* s, int n); //把字符串s的前n个字符赋给当前的字符串

string& assign(const string& s); //把字符串s赋给当前字符串

string& assign(int n, char c); //用n个字符串c赋给当前字符串

总结：string的赋值方式很多，其中operator=这种方式比较实用。

string字符串拼接

功能描述

实现在字符串末尾拼接字符串

函数原型

string& operator+=(const char\* str); //重载+=操作符

string& operator+=(const char c); //重载+=操作符

string& operator+=(const string& str); //重载+=操作符

string& append(const char\* s); //把字符串s连接到当前字符串结尾

string& append(const char\* s, int n); //把字符串s 的前n个字符链接到当前字符串结尾

string& append(const string& s); //同operator+=(const string& str)

string& append(const string& s,int pos, int n); //字符串s中从pos开始的n个字符链接到字符串结尾

string查找和替换

功能描述：

查找：查找指定字符是否存在

替换：在指定的位置替换字符串

函数原型

int find (const string& str, int pos = 0) const; // 查找 str 第一次出现位置，从 pos 开始查找int find (const char\* s, int pos = 0) const; // 查找 s 第一次出现位置，从 pos 开始查找int find (const char\* s, int pos, int n) const; // 从 pos 位置查找 s 的前 n 个字符第一次位置int find (const char c, int pos = 0) const; // 查找字符 c 第一次出现位置int rfind (const string& str, int pos = npos) const; // 查找 str 最后一次位置，从 pos 开始查找int rfind (const char\* s, int pos = npos) const; // 查找 s 最后一次出现位置，从 pos 开始查找int rfind (const char\* s, int pos, int n) const; // 从 pos 查找 s 的前 n 个字符最后一次位置int rfind (const char c, int pos = 0) const; // 查找字符 c 最后一次出现位置string& replace (int pos, int n, const string& str); // 替换从 pos 开始 n 个字符为字符串 strstring& replace (int pos, int n,const char\* s); // 替换从 pos 开始的 n 个字符为字符串 s

string字符串比较

功能描述

字符串之间的比较

比较方式

字符串比较是按字符的ASCII码进行对比

=返回0

>返回1  
 <返回-1

函数原型  
 int compare(const string& s)const; //与字符串s比较

int compare(const char\* s)const; //与字符串s比较

string字符存取

string中单个字符存取方式有两种

char& operator[](int n); //通过[]方式取字符

char& at(int n); //通过at方法获取字符

string插入和删除

功能描述：

对string字符串进行插入和删除字符操作

函数原型：

string& insert (int pos, const char\* s); // 插入字符串string& insert (int pos, const string& str); // 插入字符串string& insert (int pos, int n, char c); //在指定位置插入 n 个字符 cstring& erase (int pos, int n = npos); //删除从 Pos 开始的 n 个字符

string字串

功能描述

从字符串中获取想要的字串

函数原型

string substr(int pos = 0, int n = npos)const; //返回由pos开始的n个字符组成的字符串