

字符串类string

`std::string` 是 C++ 标准库中用于处理字符串的类，提供了便于操作字符串的丰富功能，相比于传统的 C 风格字符串（以 `null` 结尾的字符数组），`std::string` 提供了更加便捷、安全和灵活的字符串处理方法。

1.string 类的输入输出

- 包含头文件 `#include<string>`，可以直接使用 `cin` 和 `cout` 读入 `string` 类，忽略开头的空格换行制表符，再次遇到空格换行制表符的时候停止
- `getline(cin, s)` `s` 是一个字符串对象，可以用来直接读取一行的字符串并读入空格制表符

2.构造string对象和初始化

```
std::string str1 = "Hello"; // 用字符串字面值初始化
std::string str2 = str1; // 用已有的 std::string 对象初始化
std::string str3("World"); // 直接使用字符串字面值初始化
std::string str4(5, 'x'); // 重复5次字符 'x' 构造
std::string original = "Some text";
std::string str5(original.begin(), original.begin() + 4); // 使用迭代器范围构造
std::string str7 = "Hello";
str7 += " World"; // 使用 += 操作符追加字符串
str7.append("!!"); // 使用 append() 函数追加字符串
```

3.string字符元素的访问

- `s[i]` 返回 `s` 在 `i` 处的索引值，不检查是否出界
- `s.at()` 返回 `s` 在 `i` 处的索引值，检查是否出界

4.重载运算符

- 对于比较运算符，下面的功能是指使用字典序对字符串进行排序
- 对于字符串而言，字典序指的是按照字母表顺序进行排列。在英语字母表中，字母的字典序是从 A 到 Z，依次排列。当比较两个字符串时，从左到右逐个比较字符的 ASCII 码值（或 Unicode 码值），直到发现不同的字符为止。第一个不同的字符决定了两个字符串在字典序中的先后顺序。例如：
 - "apple" 在字典序中排在 "banana" 之前，因为 'a' 比 'b' 小。
 - "cat" 在字典序中排在 "car" 之前，因为在第三个位置上，'t' 比 'r' 小。

```
string s1 = "hello";
string s2("hola");
cout << (s1 == s2);
cout << (s1 != s2);
cout << (s1 >= s2);
cout << (s1 <= s2);
cout << s1 += s2;
```

5.string类的常见成员函数

5.1容量相关函数

- `size()` / `length()`：返回字符串的长度。

- `empty()`: 检查字符串是否为空。
- `resize(newSize)`: 修改字符串的长度为 `newSize`。
- `capacity()`: 返回字符串能容纳的字符数量。 **

5.2访问字符相关函数

- `at(pos)` / `operator[]`: 访问指定位置的字符。
- `front()`: 返回第一个字符。
- `back()`: 返回最后一个字符。
- `c_str()`: 返回指向以 `null` 结尾的字符数组的指针。

5.3修改字符串内容函数

- `clear()`: 清空字符串内容。
- `push_back(c)`: 在字符串末尾添加字符 `c`。
- `pop_back()`: 移除字符串末尾的字符。
- `append(str)` / `+=`: 在字符串末尾添加字符串 `str`。
- `insert(pos, str)`: 在指定位置插入字符串 `str`。
- `erase(pos, count)`: 移除指定位置开始的 `count` 个字符。
- `replace(pos, count, str)`: 用字符串 `str` 替换从 `pos` 开始的 `count` 个字符。

5.4字符串搜索和比较函数

- `find(str)` / `rfind(str)`: 在字符串中查找字符串 `str` 的第一次/最后一次出现的位置。
- `substr(pos, count)`: 返回从 `pos` 开始、包含 `count` 个字符的子串。
- `compare(str)`: 比较字符串与 `str` 是否相等。

5.5其他函数

- `sort(s, s+n)`: 对字符串内的字符按照 **字典序** 进行排序
- `swap(str)`: 交换两个字符串的内容。
- `getline(stream, str)`: 从输入流中读取一行内容存入字符串 `str`。