

C++ String类

1 string类构造

```
1 #include <string>
2 using std::string;
3 int main(int argc, char** argv) {
4     string s1;
5     string s2("This is string 2");
6     string s3(s2); //拷贝构造
7     string s4(10, 'x'); //指定长度和字符构造
8     return 0;
9 }
```

2 string类对象的容量操作

```
1 #include <string>
2 #include <iostream>
3 using std::string;
4 using std::cout;
5 using std::endl;
6 int main(int argc, char** argv) {
7     string s("This is string");
8     cout << s.size() << endl; //返回字符串有效长度
9     cout << s.length() << endl; //返回字符串有效长度
10    cout << s.capacity() << endl; //返回空间大小
11    cout << s << endl;
12    s.clear(); //清空字符串
13    s.empty(); //判断字符串是否为空, 为空返回true, 否则返回false
14    s.resize(19, 'a'); //将s的有效字符设置为19, 多余的位置用'a'填充
15    s.resize(15); //将s的有效字符设置为15个, 多余的位置用'\0'填充
16
17    cout << s.size() << endl; //返回字符串有效长度
18    cout << s.length() << endl; //返回字符串有效长度
19    cout << s.capacity() << endl; //返回空间大小
20    cout << s << endl;
21
22    string s1;
23    s1.reserve(100); //设置空间大小
24    cout << s1.capacity() << endl; //返回空间大小
25    s1.reserve(50); //当空间小于string底层空间大小时, 不会减小
26    cout << s1.capacity() << endl; //返回空间大小
27    s1 = "This is string";
28    cout << s1 << endl;
29    return 0;
30 }
```

3 string类对象的访问及遍历操作

```
1 #include <string>
```

```

2  #include <iostream>
3  using namespace std;
4  int main(int argc, char** argv) {
5      string s = "This is string";
6      s[0] = 't';
7      cout << s << endl;
8
9      //1 for and operator[]
10     for (int i = 0; i < s.size(); i++) {
11         cout << s[i] << endl;
12     }
13
14     //2 forward iterator
15     string::iterator it = s.begin();
16     while (it != s.end()) { //end() 获取最后一个元素的下一个位置
17         cout << *it << endl;
18         it++;
19     }
20
21     //3 reverse iterator
22     string::reverse_iterator rit = s.rbegin();
23     while (rit != s.rend()) {
24         cout << *rit << endl;
25         rit++;
26     }
27
28     //4 const iterator
29     string::const_iterator cit = s.cbegin();
30     while (cit != s.cend()) {
31         cout << *cit << endl;
32         cit++;
33     }
34     return 0;
35 }

```

4 string类对象的修改操作

```

1  #include <string>
2  #include <iostream>
3  using namespace std;
4  int main(int argc, char** argv) {
5      string s;
6      s.push_back('a'); //在字符串末尾添加一个字符
7      s.append("bcd"); //在字符串末尾添加一个字符串
8
9      s += 'e'; //在字符串末尾添加一个字符
10     s += "fgh"; //在字符串末尾添加一个字符串
11
12     s.insert(s.begin(), 'x'); //在字符串的指定位置插入一个字符
13     s.erase(0, 1); //删除字符串中的指定位置的字符, 不包括1
14     cout << s << endl;
15     cout << s.c_str() << endl; //打印字符串的c风格字符串
16     return 0;
17 }

```

