

# C++语言程序设计

贺利坚 主讲

## 容器的分类与基本功能

# 容器的分类

► 容器类是容纳、包含一组元素或元素集合的对象

► 七种基本容器：

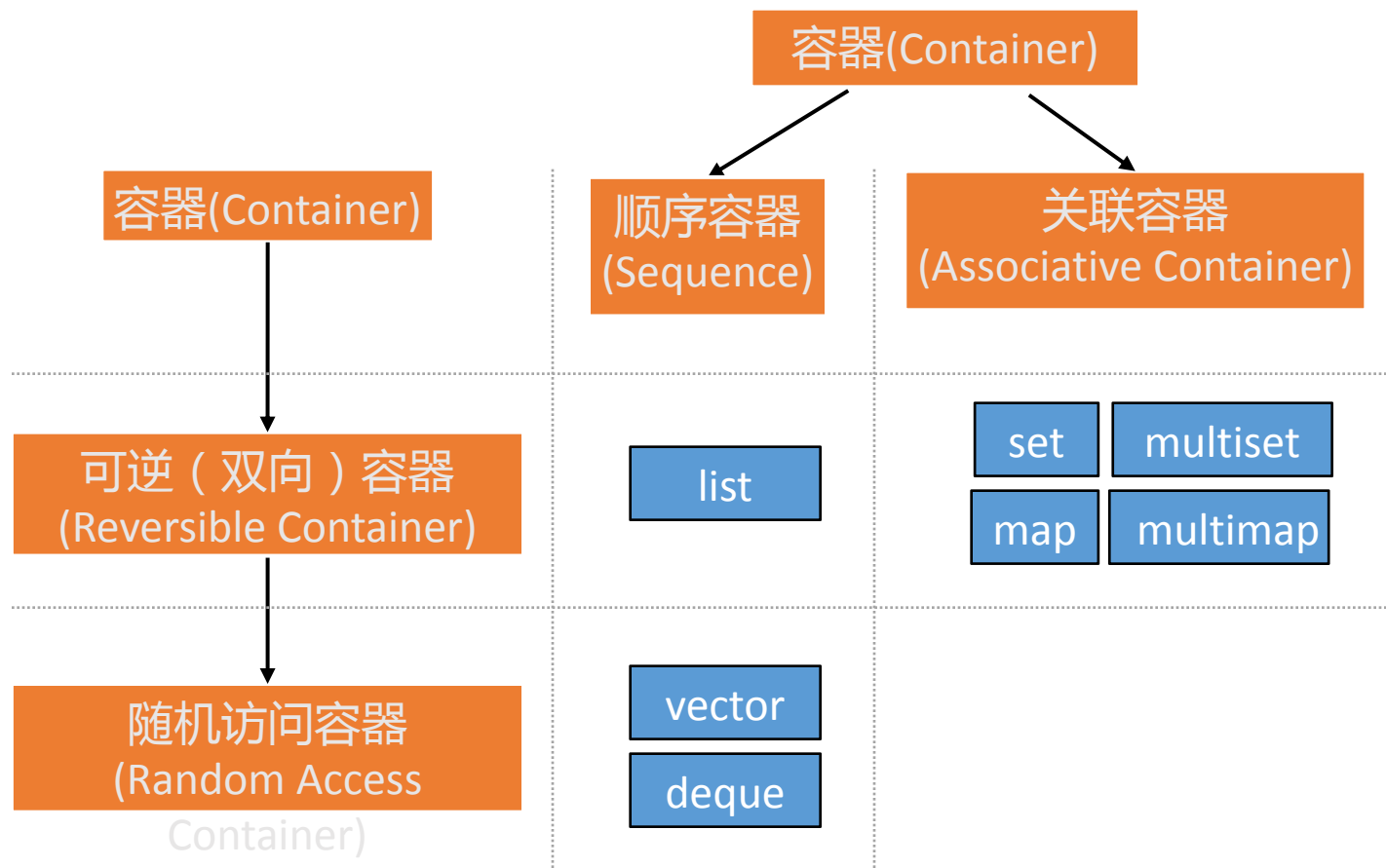
- 向量 ( vector )
- 双端队列 ( deque )
- 列表 ( list )
- 集合 ( set )
- 多重集合 ( multiset )
- 映射 ( map )
- 多重映射 ( multimap )

► 按容器中元素的组织方式

- 顺序容器
- 关联容器

► 按与容器相关的迭代器类型

- 可逆容器
- 随机访问容



# 容器的通用功能

## ► 容器的通用功能

- 用默认构造函数构造空容器
- 支持关系运算符：`==`、`!=`、`<`、`<=`、`>`、`>=`
- `begin()`、`end()`：获得容器首、尾迭代器
- `clear()`：将容器清空
- `empty()`：判断容器是否为空
- `size()`：得到容器元素个数
- `s1.swap(s2)`：将s1和s2两容器内容交换

## ► 随机访问容器

- `s[n]`：获得容器s的第n个元素

## ► 相关迭代器类型（s表示容器类型）

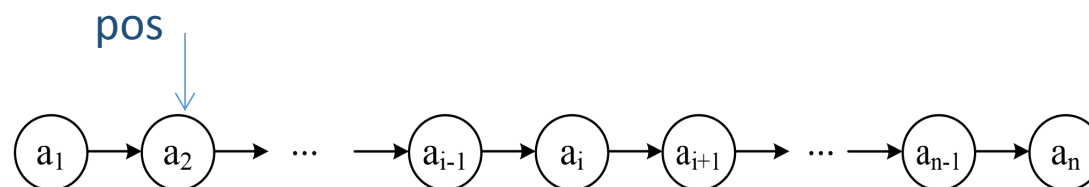
- `S::iterator`：指向容器元素的迭代器类型
- `S::const_iterator`：常迭代器类型

## ► 示例：容器功能

- `vector<Student> s1, s2;`
- `s1==s2`
- `s1.begin()`、`s1.end()`
- `s1.clear()`
- `s1.empty()`
- `s1.size()`
- `s1.swap(s2)`

## ► 示例：相关迭代器

- `vector<Student>::iterator pos;`
- `vector<Student>::const_iterator cpos;`



# 可逆(双向)容器的功能

## ► 可逆容器

- `S::reverse_iterator` : 逆向迭代器类型
- `S::const_reverse_iterator` : 逆向常迭代器类型
- `rbegin()` : 指向容器尾的逆向迭代器
- `rend()` : 指向容器首的逆向迭代器

## ► 例

- `copy(s.begin(), s.end(), ostream_iterator<int>(cout, " "));`
- `copy(s.rbegin(), s.rend(), ostream_iterator<int>(cout, " "));`

