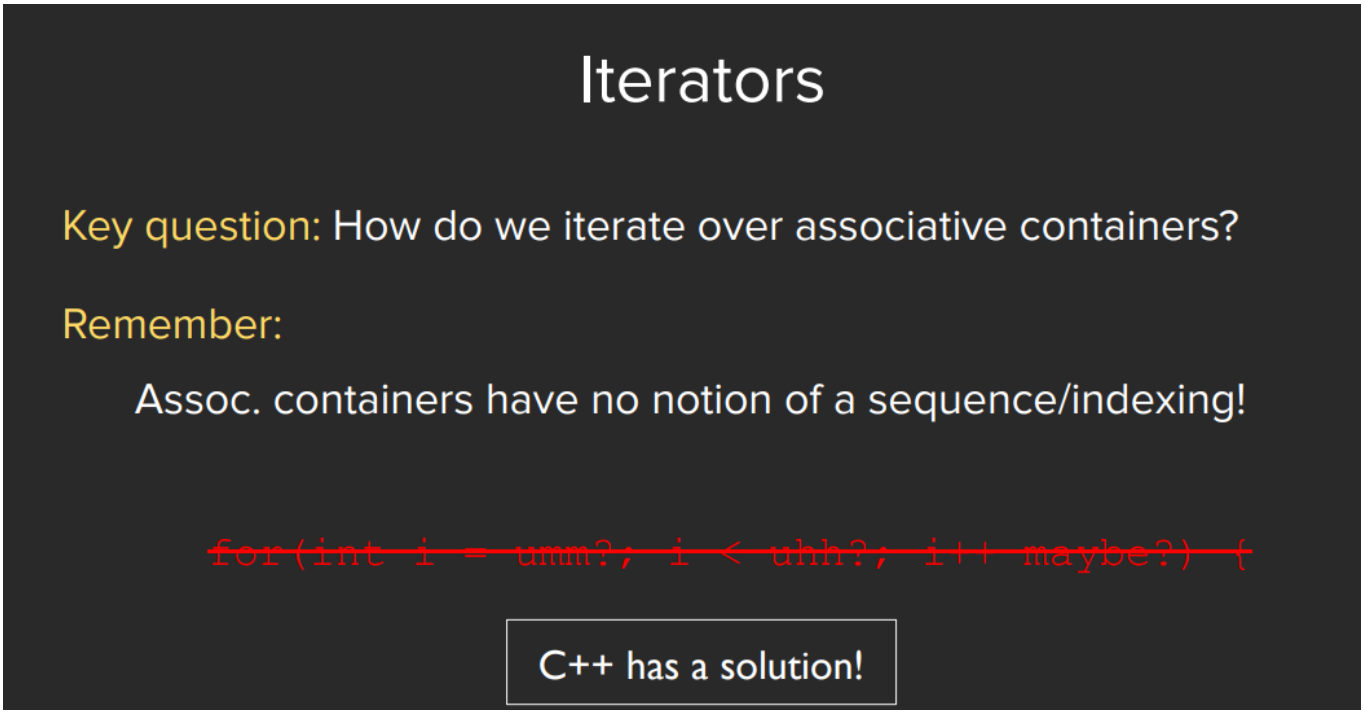
Iterators

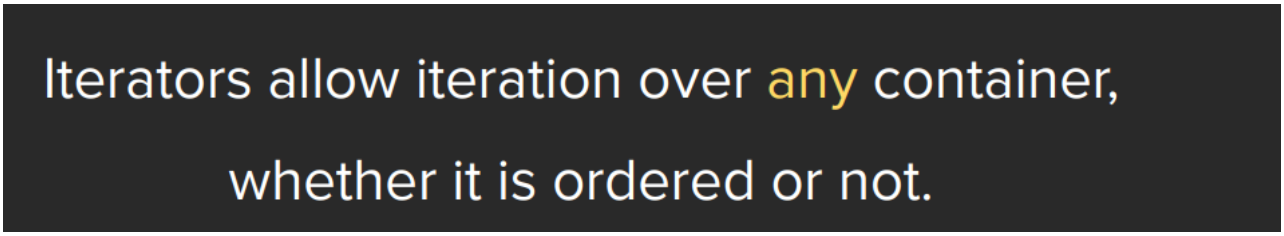
如何访问关联容器里面的数据



iterators

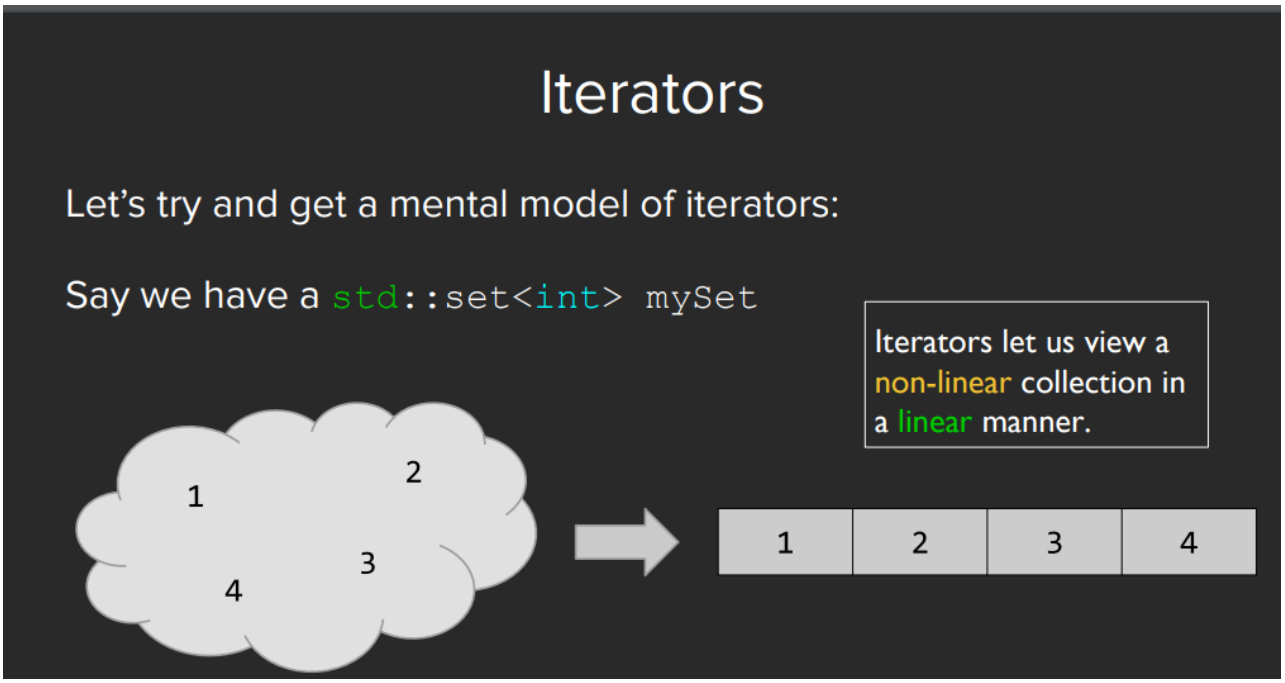
iterators介绍

1. Iterators are something in STL that allow iteration over any container, so all of this sequence and associative containers.



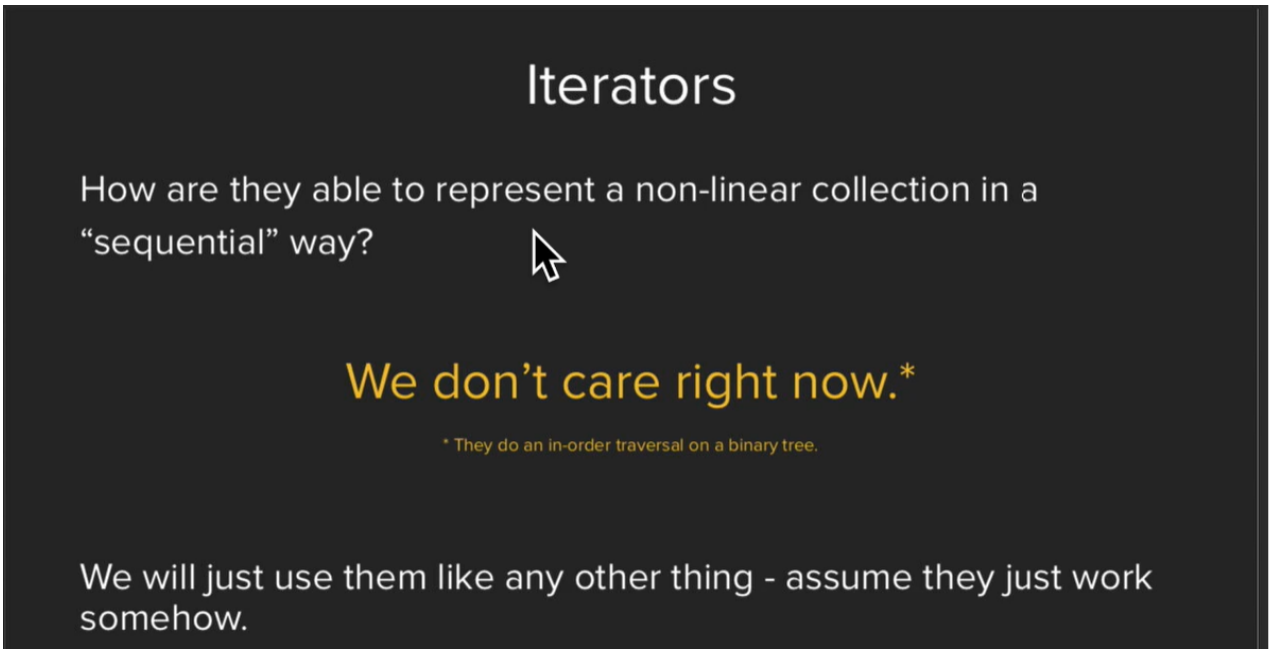
iterator的作用

1. iterator are a device that let us view this nonlinear container in a linnar manner



iterator的机制是如何实现的

1. the key idea is iterators are like a sort of abstraction.
2. You're not actually worrying about like what exactly how its implemented, you just know that it's pointing to some elements and you can say next or previous on that.
   目前我们这个阶段, 还不用掌握`iterator`是如何实现的

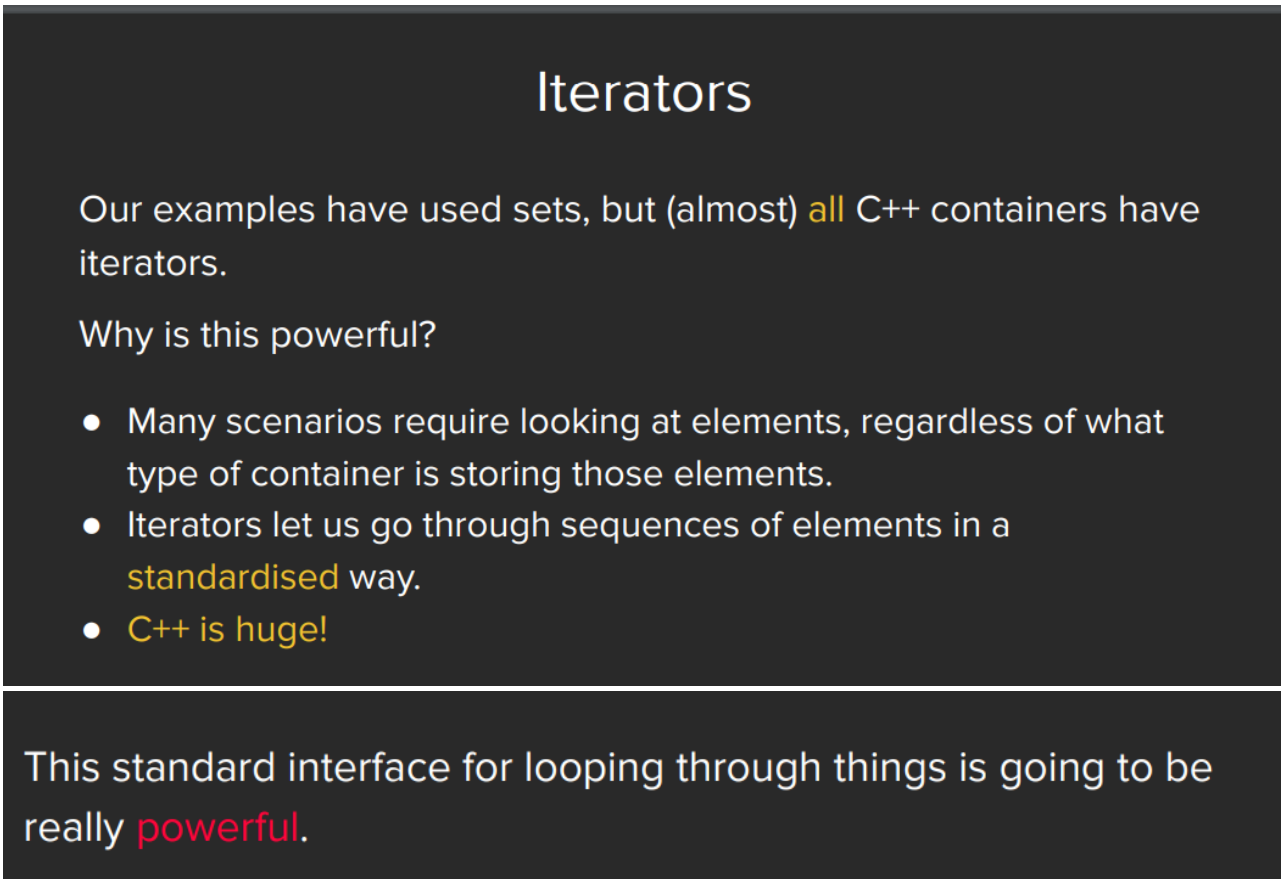


iterator的底层实现

1. iterator的底层实现就是指针, 可以将`iterator`看作是对指针的封装
   即便`iterator`和指针还是有一点不同的, 但是我们仍然可以将`iterator`理解为指针(内存)
2. vector iterators are just pure iterators.
3. sequential containers iterators really are pointers under the hood.

为什么iterator那么有用

1. 为什么iterator那么有用
   任何的容器都以相同的方式使用`iterator`



iterator相关操作(使用iterator)

* iterator(解引用)

1. 对iterator解引用,得到其所指向的对象

`++iterator` vs `iterator++`

前缀后缀递增对比	效果
`++iterator`	先递增`iterator`,然后再使用`iterator`
`iterator++`	先使用`iterator`, 然后再递增`iterator`

begin函数

end函数

示例: 对比vector::iterator和vector: operator[]的效率

1. 当我们使用使用`for`循环来遍历vector中的数据时
   是使用`operator[]`访问数据的效率高?
   还是使用`vector::iterator`访问数据的效率高?
   基本上来说老师觉得两者的效率其实是一样的
   because teacher believe C++ may actually under the hood , be using an iterator to implement that.

操作总结