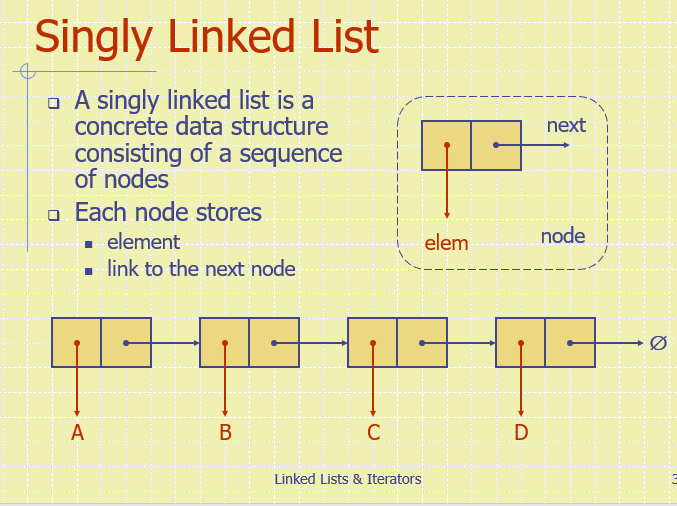
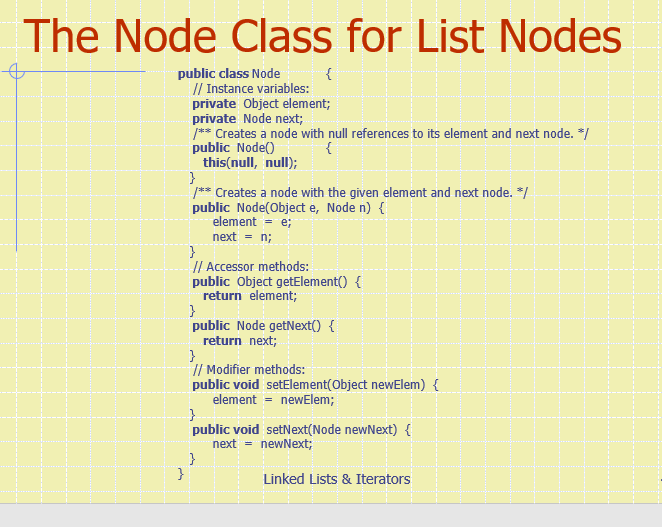
Singly linked list

一个singly linked list 是一个具体的data structure，由一系列nodes组成

每一个node存储了，信息，下一个node的link



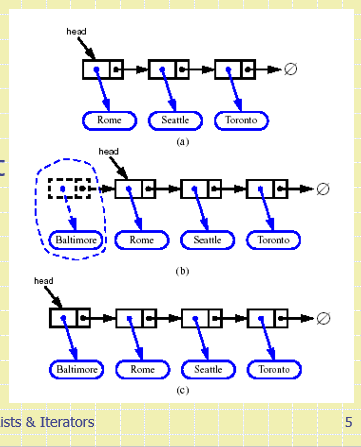


怎么在head insert

创建一个新node，

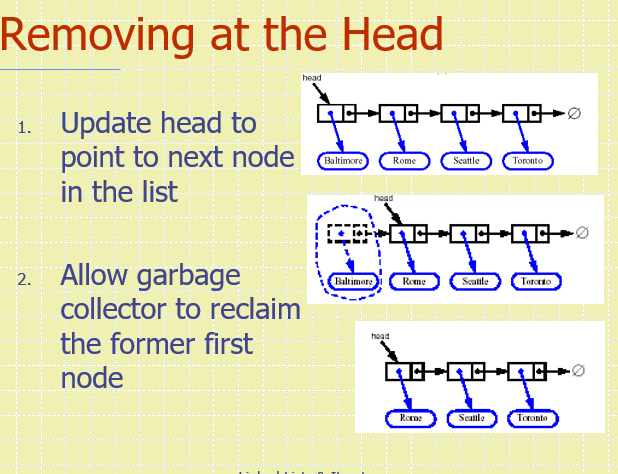
在node中加入要的元素，

让新node指向原来的head，更新head，让head指向新node



从head remove

1. 让head指向下一个node
2. 让garbage collector 回收第一个node



INSERT AT TAIL

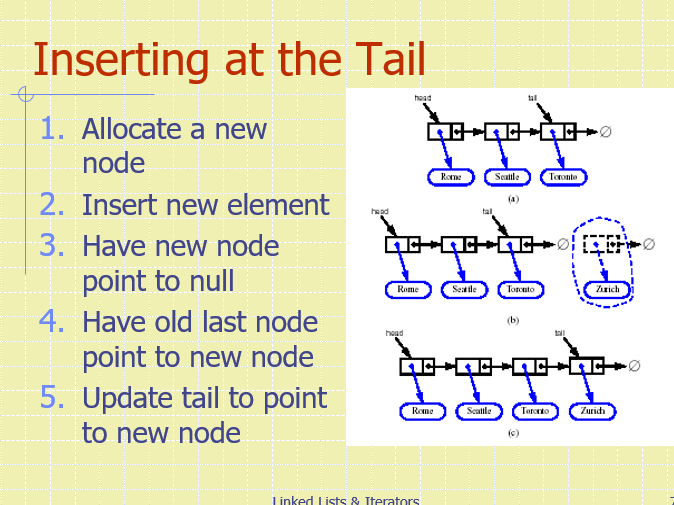
创造一个新node

加入新element

让新node指向Null

让old tail node指向新node

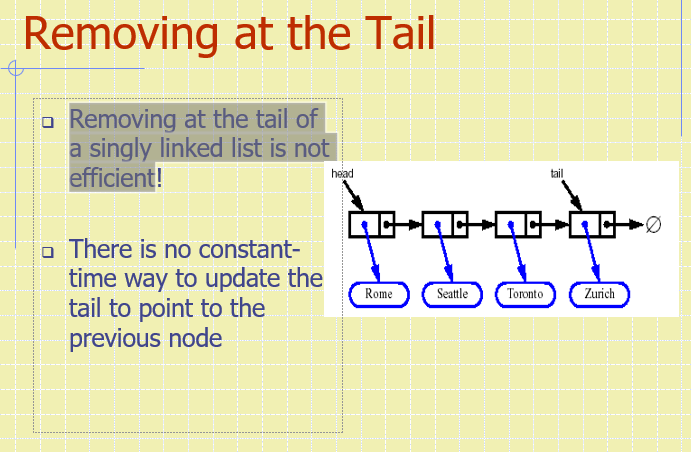
update tail让他指向新node



remove tail

删除singliy linked list效率很低

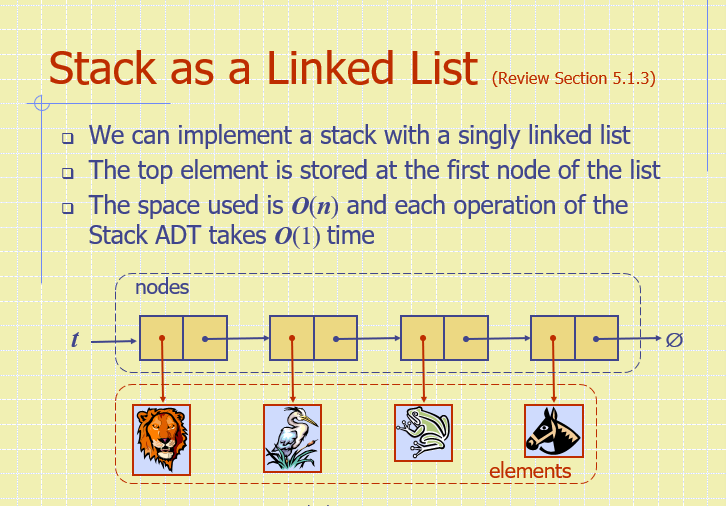
没有一个固定时间complexity的方法来update tail让他指向之前一个node



我们可以用singly linked list来构建一个stack

top就是first node

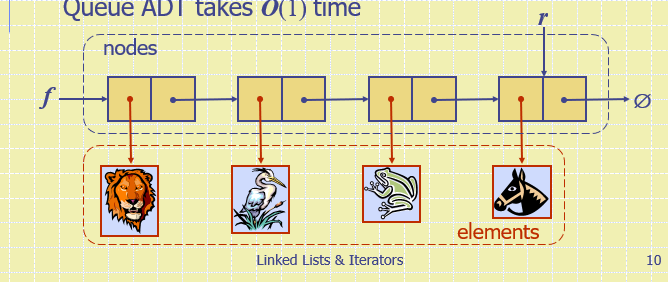
需要的space是O（n），每一个对STACK ADT的操作使用O（1）时间



QUEUE AS A LINKED LIST

front是first node，rear是last node

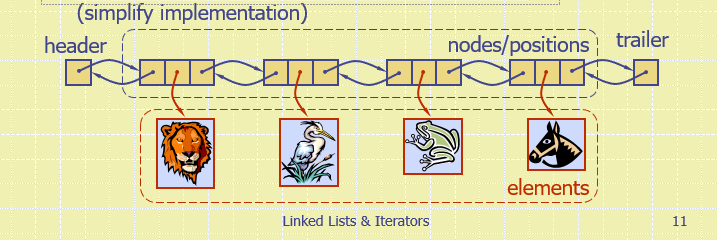
所用空间是O（n）,每个操作是O（1）



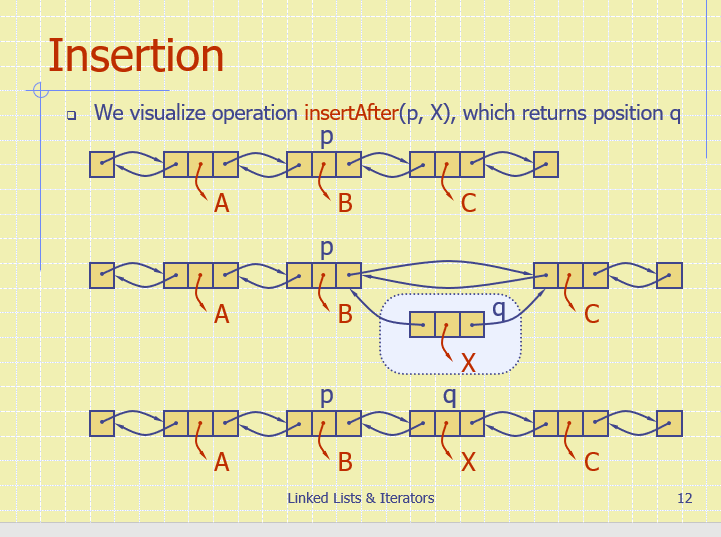
Doubly Linked List 是一种Node List ADT

每一个node包含，element，前一个node的link，后一个node的link

header和trailer有一个特殊节点，单向有element，双向head 和trailer没element



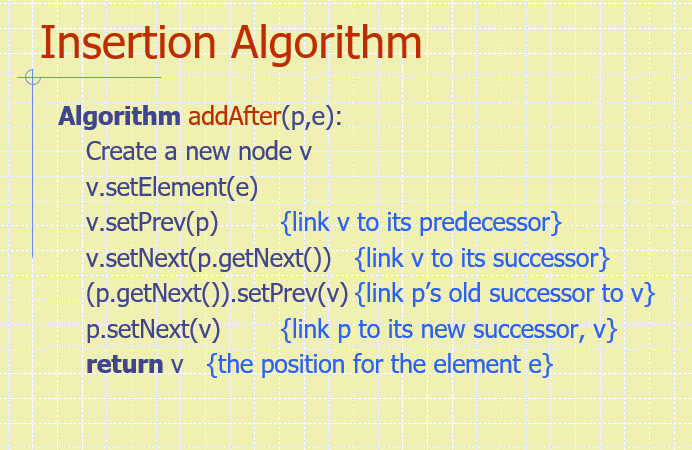
Insertion



我们实现用insertAfter(p,X)的方法，会return position q的位置

先创建一个新node,值是X，位置是q，左边指向B，右边指向C

然后在改变BC的指向



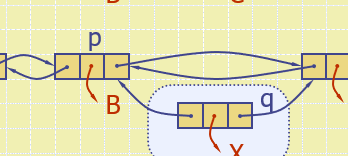
创造新node

设置element

设置Prev让他连上前一个

设置next让他连上后一个//后一个就是原来P的后一个

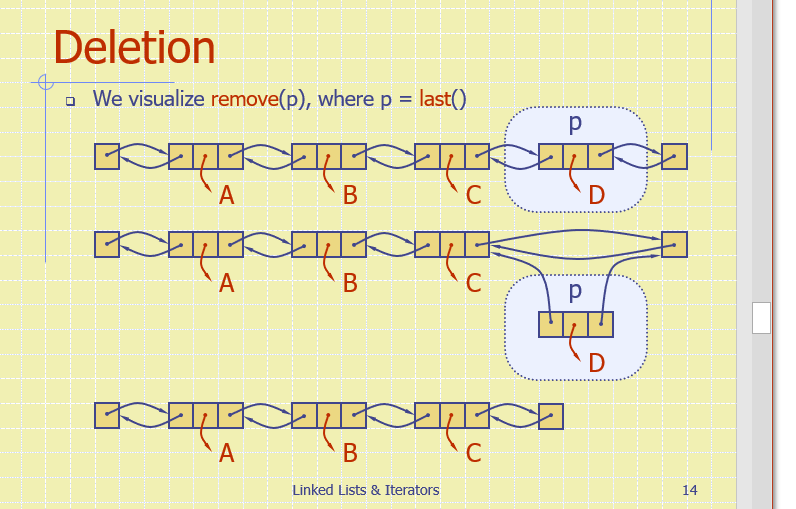
这时V的后一个是P的后一个，P的后一个还是原来的后一个

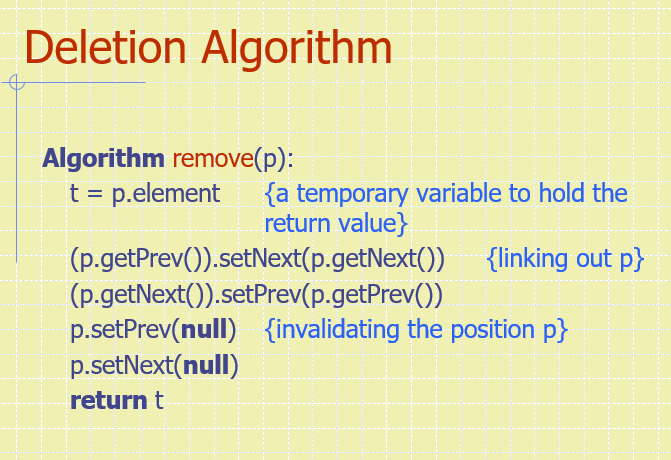


让P原来的后一个把前一个设置成V

让P的后一个设置成V，

DELETION





创造一个暂时变量来从存储P这个位置的element

让P之前的一个node指向P的后一个node

让P的后一个node指向之前的Node

让P不再指向这两个Node

return t//P的值

performance

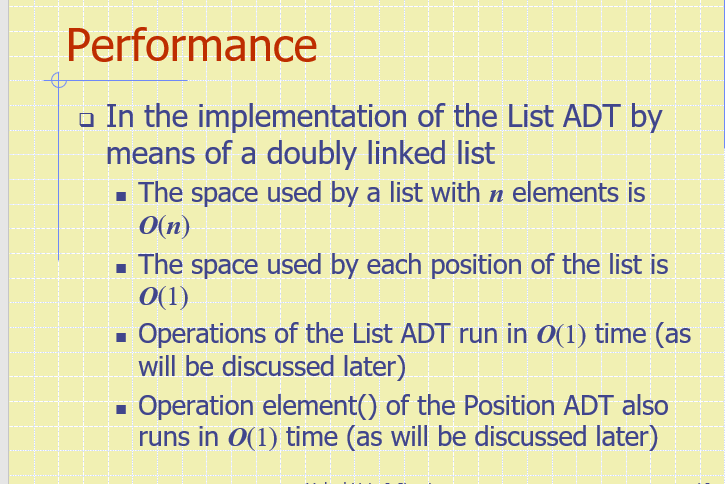
对于List ADT中的doubly linked list

用的space是O（n）

每一个位置所用的space是O（1）

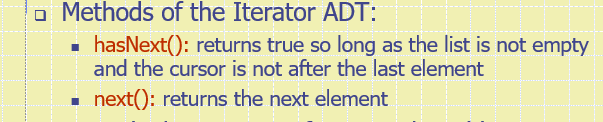
每一个关于List ADT操作是O（1）时间

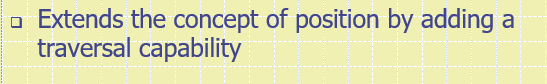
每一个关于P哦实体欧尼ADT的操作仍是O（1）



Iterator展示了扫描元素集合的过程

他在list 的element之间放了一个指针，或者在head之前或是last之后



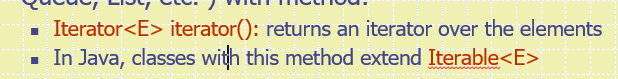


iterator通过增加了一个traversal capability遍历功能扩展了位置的概念

Iterable Class

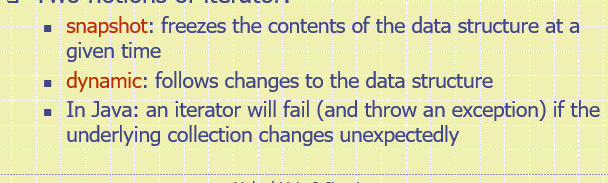
iterator 通常与另一个数据结构相连，这个数据结构implement Iterable ADT

我们可以吧不同ADT作为参数（STACK,QUEUE,LIST等等）



iterator()；return一个iterator

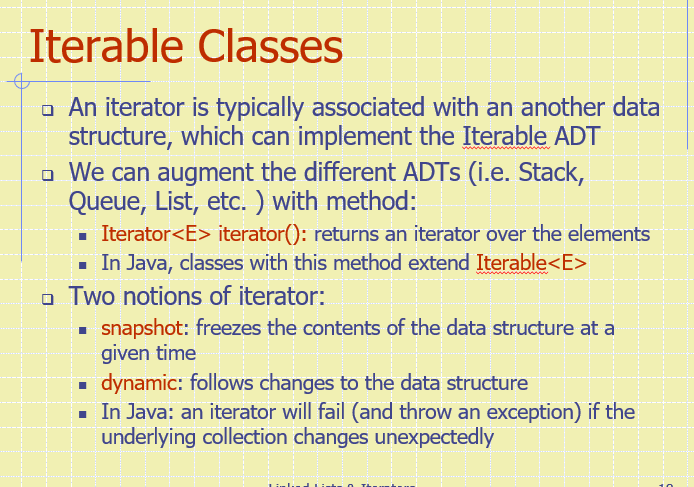
在JAVA中，这样的class extend Iterable<E>



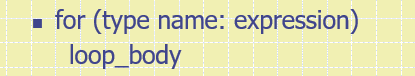
snapshot 快照:冻结数据结构在给定瞬间的内容

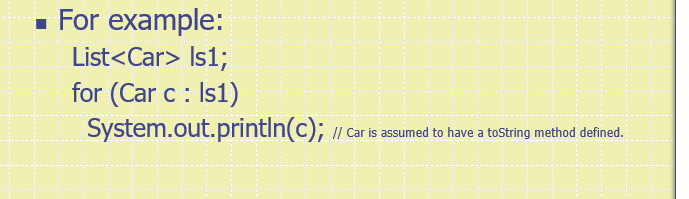
dynamic：动态，允许数据结构的改变

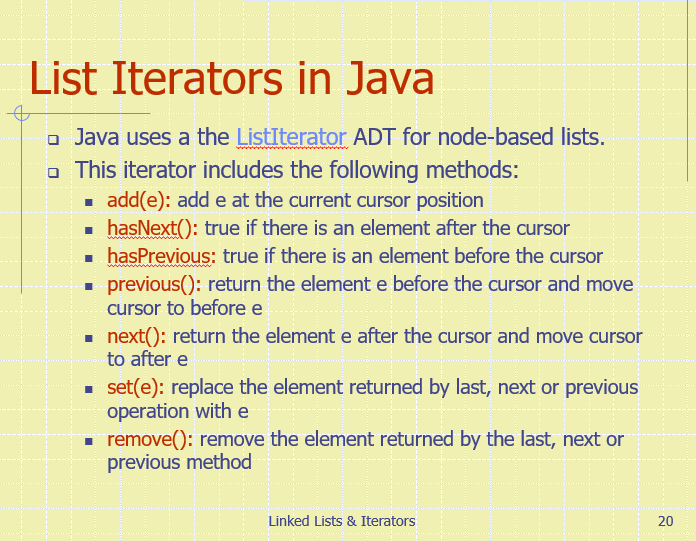
在Java中:如果底层集合意外改变，迭代器将失败(并抛出异常)



JVA提供了一个简单方法来loop iterable class中的每一个元素







对一个node-based list来说， JAVA使用的是ListIterator ADT

有以下method

