Unit 16: Iterators

Anthony Estey

CSC 115: Fundamentals of Programming II

University of Victoria

Unit 16 Overview

- Learning Objectives: (You should be able to…)
  - use an Iterator object to loop through all elements in a collection

Java's built in list classes

- Java's built in List and Linked List classes:
  - https://docs.oracle.com/javase/8/docs/api/java/util/List.html
  - https://docs.oracle.com/javase/8/docs/api/java/util/LinkedList.html

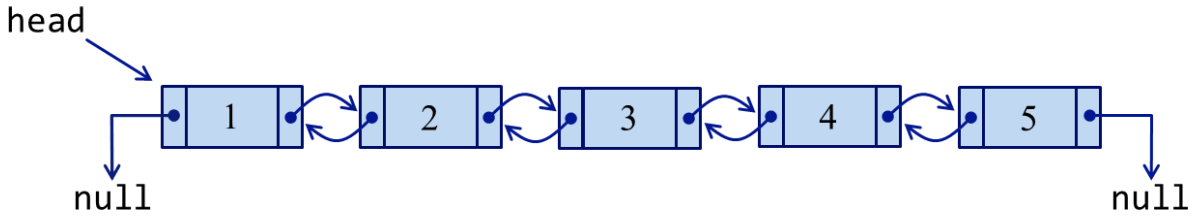
- Why use these?

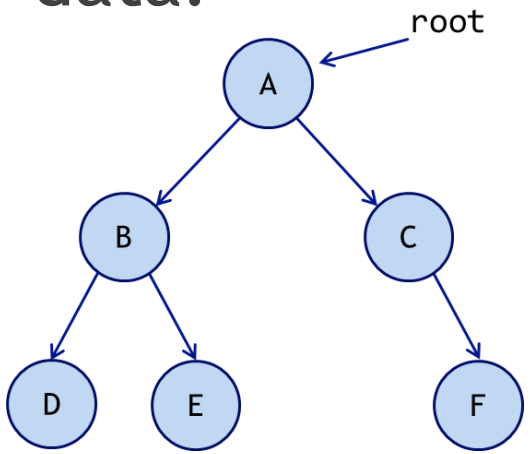
1. They has all of the list operations implemented for us
2. They are generic, so we can use them with any types of data we wish

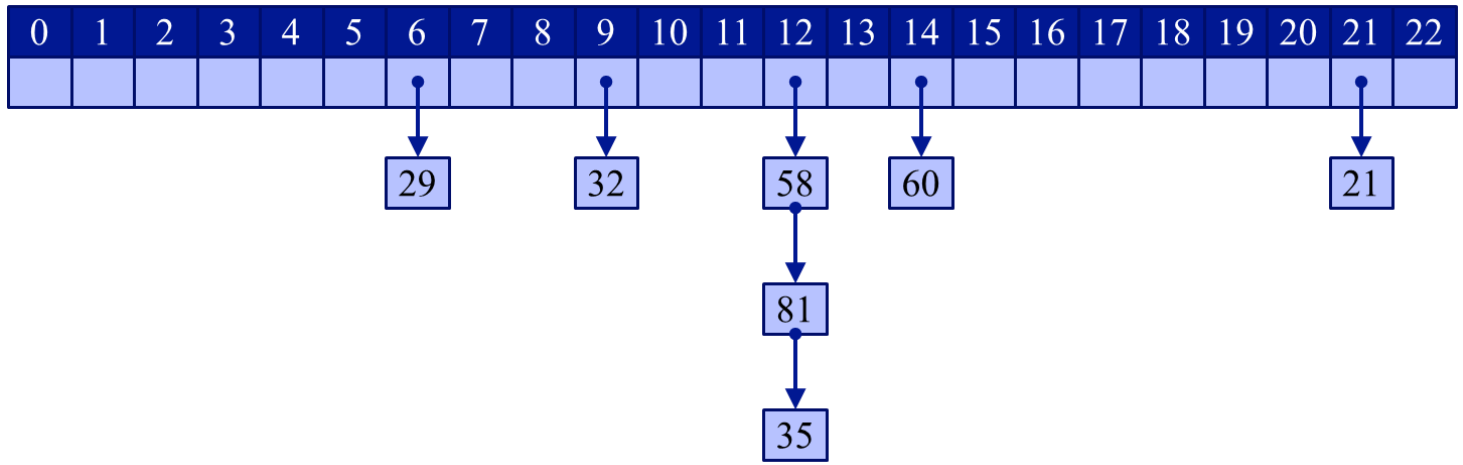
- This means:
  - We don't need to implement a list for every type of object we create
  - We can assume that it has been thoroughly tested and works correctly

Different ways to manage and organize data

- Over the past few months we have explored a number of different ways we can store and operate on a collection of data:







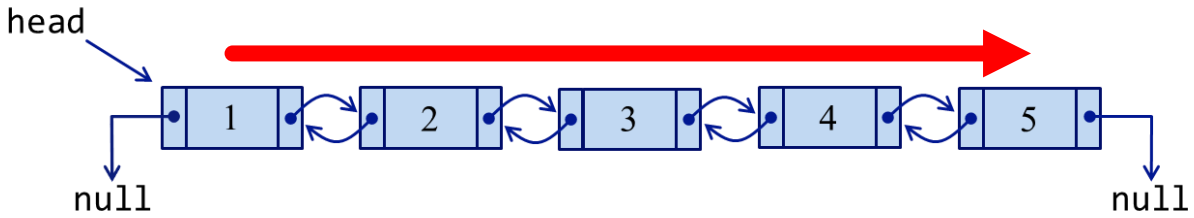
Different ways to manage and organize data

- We've talked a little bit about the pros and cons of each implementation, mostly in Big-Oh terms
- Worst-case time complexity:

	Insert	Remove	Find
Unordered array	$O(1)$	$O(n)$	$O(n)$
Ordered array	$O(n)$	$O(n)$	$O(\log n)$
Unordered list	$O(1)$	$O(n)$	$O(n)$
Ordered list	$O(n)$	$O(n)$	$O(n)$
BST	$O(n)$	$O(n)$	$O(n)$
BST (balanced)	$O(\log n)$	$O(\log n)$	$O(\log n)$

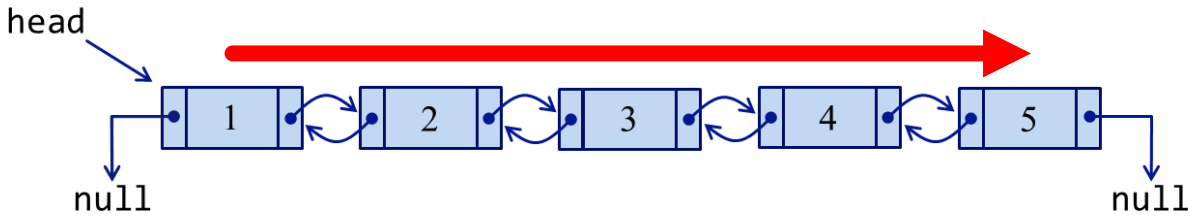
Different ways to manage and organize data

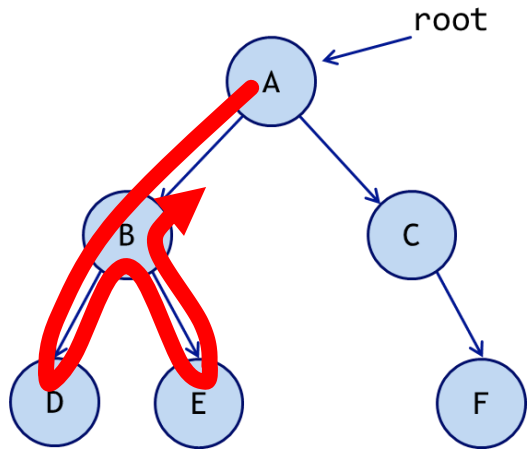
- There are different ways we can output the contents of our collection:



Different ways to manage and organize data

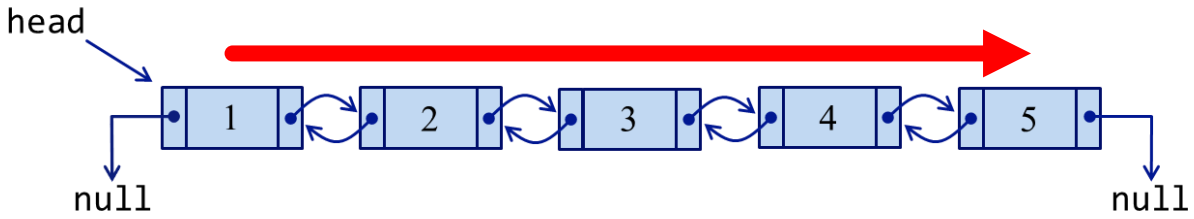
- There are different ways we can output the contents of our collection:

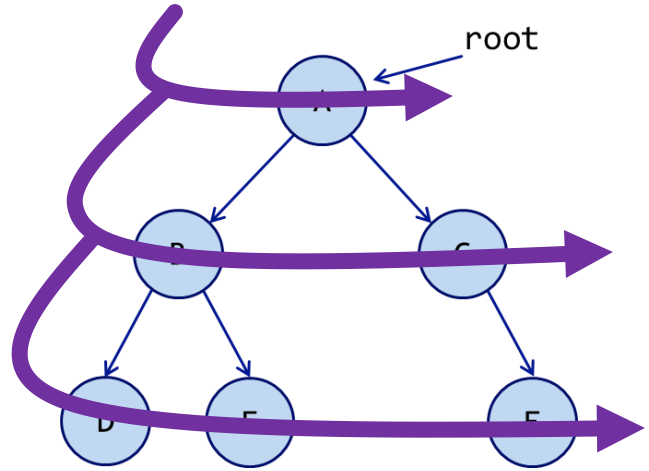




Different ways to manage and organize data

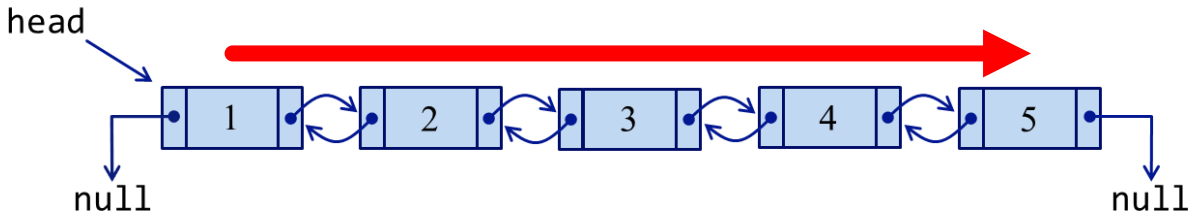
- There are different ways we can output the contents of our collection:

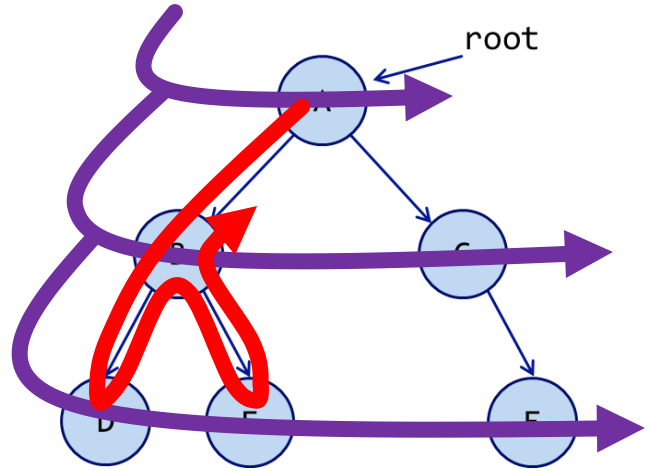


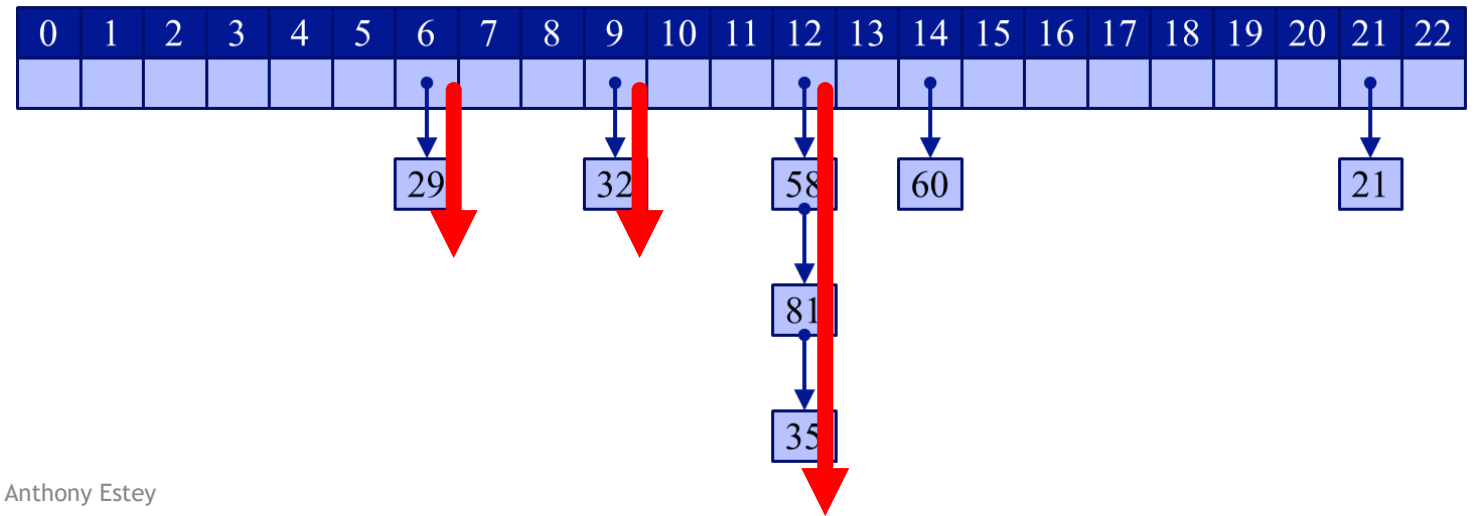


Different ways to manage and organize data

- There are different ways we can output the contents of our collection:







Iterators

- Iterators allow us to loop through all of the elements in a collection

- Java has a built-in Iterator class (in the java.util package)
  - https://docs.oracle.com/javase/8/docs/api/java/util/Iterator.html

- The class has a few methods:
  - forEachRemaining
  - hasNext
  - next
  - remove

Iterator example

```
LinkedList<Integer> entries = new LinkedList<Integer>();
entries.add(7);
entries.add(12);
entries.add(2);
entries.add(5);

Iterator<Integer> it = entries.iterator();
while (it.hasNext()) {
    Integer cur = it.next();
    System.out.println(cur);
}
```

Make a linked list of integers

Add some values to the list

Connect an iterator to the list

While there are still more elements to visit…

Output the element and move to the next one