5.3 Two Implementations for Priority Queues

LEARNING OBJECTIVES

1. To understand how it is possible to replace the underlying representation for an abstract data type without changing the operations it presents to its external users through its interface.
2. To learn how to use information-hiding features of Java to hide the implementation details of an abstract data type's operations.
3. To learn how to change underlying data representations implementing an ADT without changing any of the code that uses the ADT.

Implementing Priority Queues Using Sorted Linked Lists

```
class ListNode {
    ComparisonKey   item;
    ListNode        link;
}
```

Program 5.4 The ListNode Class Definition

```
class PriorityQueue {

    private int         count;          // the number of items in the priority queue

    private ListNode    itemList;       // the linked list of items

    /*-----------------------------------------*/

        // Note that Java automatically defines the no-arg constructor
        // PriorityQueue( ) that creates an initially empty PriorityQueue
        // object having a count of zero and an empty itemList

```

Program 5.5 The Sorted Linked-List Representation of the PriorityQueue Class *(continued)*