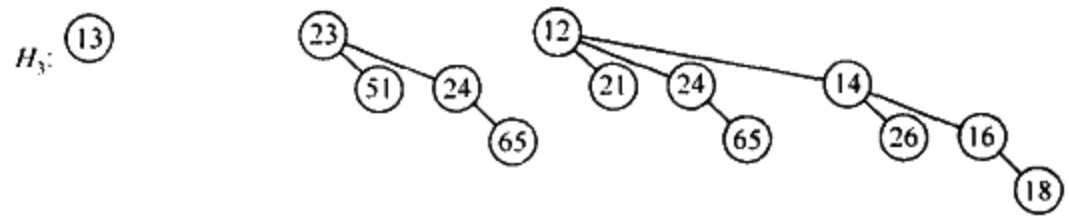

图 6-50 画成森林的二项队列 H_3

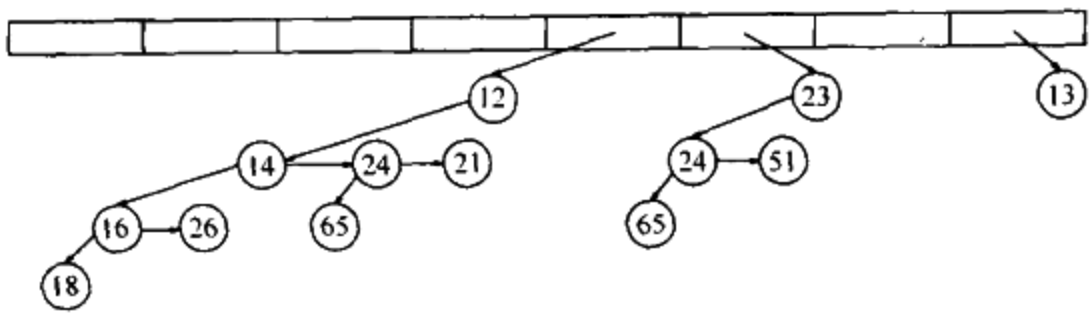

图 6-51 二项队列 H_3 的表示方式

```
1   public class BinomialQueue<AnyType extends Comparable<? super AnyType>>
2   {
3       public BinomialQueue( )
4         { /* See online code */ }
5       public BinomialQueue( AnyType item )
6         { /* See online code */ }
7
8       public void merge( BinomialQueue<AnyType> rhs )
9         { /* Figure 6.55 */ }
10      public void insert( AnyType x )
11        { merge( new BinomialQueue<AnyType>( x ) ); }
12      public AnyType findMin( )
13        { /* See online code */ }
14      public AnyType deleteMin( )
15        { /* Figure 6.56 */ }
16
17      public boolean isEmpty( )
18        { return currentSize == 0; }
19      public void makeEmpty( )
20        { /* See online code */ }
21
22      private static class Node<AnyType>
23      {
24              // Constructors
25          Node( AnyType theElement )
26            { this( theElement, null, null ); }
27
28          Node( AnyType theElement, Node<AnyType> lt, Node<AnyType> nt )
29            { element = theElement; leftChild = lt;  nextSibling = nt; }
30
31          AnyType       element;     // The data in the node
32          Node<AnyType> leftChild;   // Left child
33          Node<AnyType> nextSibling; // Right child
34      }
35
```

图 6-52 二项队列类架构及节点定义