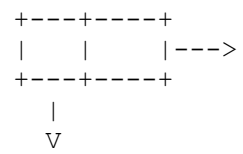


```

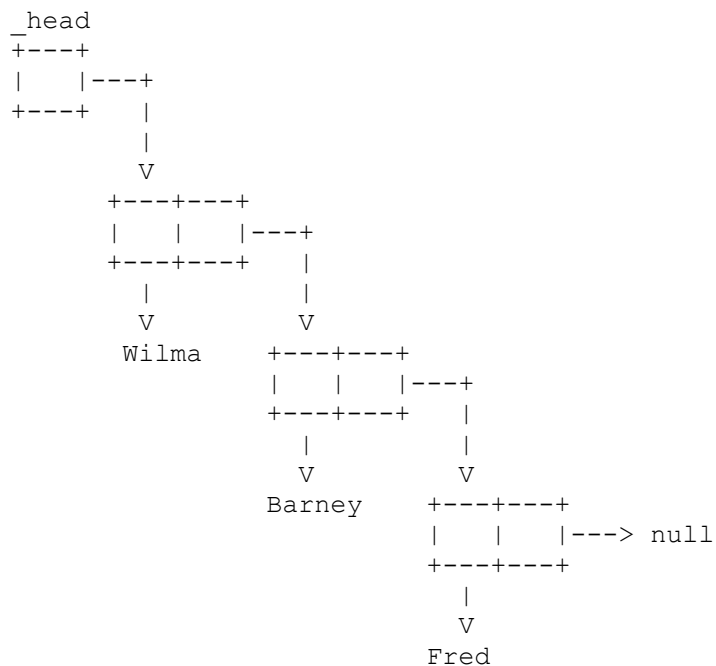
public class testQueueLinkO
{
    public static void main( String[] args )
    {
        Stack test = new Queue();
        test.insert( "Fred" );
        test.insert( "Barney" );
        test.insert( "Wilma" );
        System.out.println( test );
        System.out.println
            ( "Delete..." + test.delete() );
        System.out.println( test );
        if (test.isEmpty())
            System.out.println( "empty..." );
        Object o = test.delete();
        if (o instanceof String)
        {
            String s = (String) o;
            System.out.println
                ( "Pop..." + s +
                  " length=" + s.length() );
        }
        if (test.isEmpty())
            System.out.println( "empty..." );
    }
}

```



```
class Link
{
    public Object data;
    public Link next;
}
```

--- Back to Front ----



```

class Queue
{
    private Link _head;

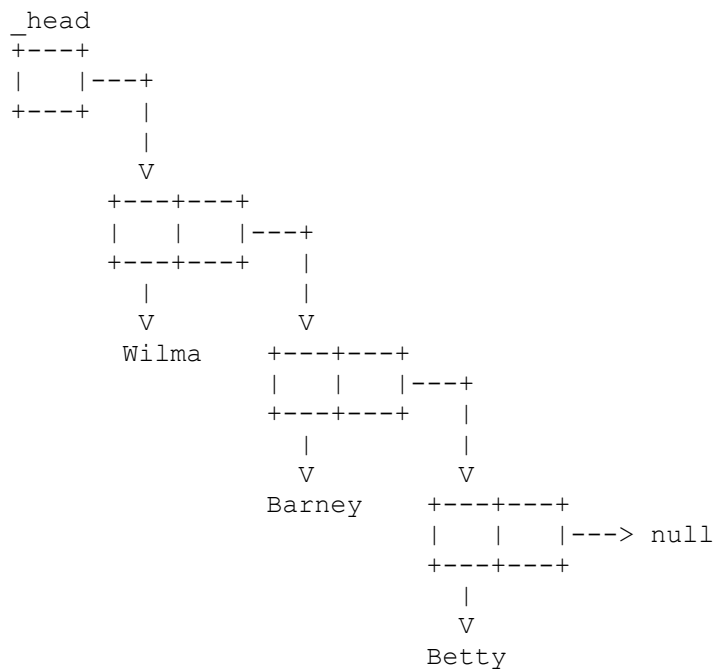
    public Queue()
    {
        _head = null;
    }

    _head
    +----+
    |    |----> null
    +----+

    public boolean isEmpty()
    {
        return (_head == null);
    }
}

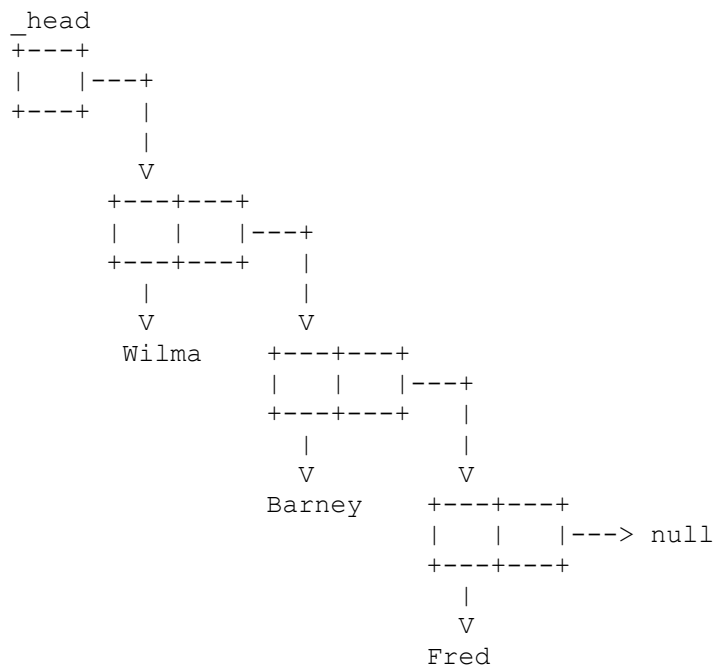
```

--- Back to Front ----



```
public void insert( Object item )
{
    Link add = new Link();
    add.data = item;
    add.next = _head;
    _head = add;
}
```

--- Back to Front ----

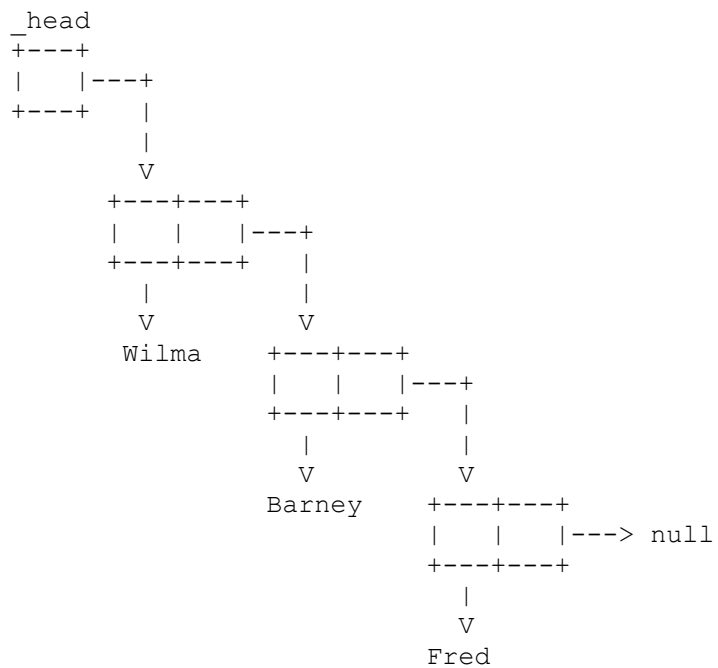


```

public String toString()
{
    String s = "";
    s += "Queue contents\n";
    s += "-- back --\n";
    Link curr = _head;
    while( curr != null )
    {
        s += curr.data + "\n";
        curr = curr.next )
    }
    s += "-- front --";
    return s;
}

```

--- Back to Front ----



```

/*
    Link curr = _head;
    while( curr != null )
    {
        curr = curr.next )
    }
*/

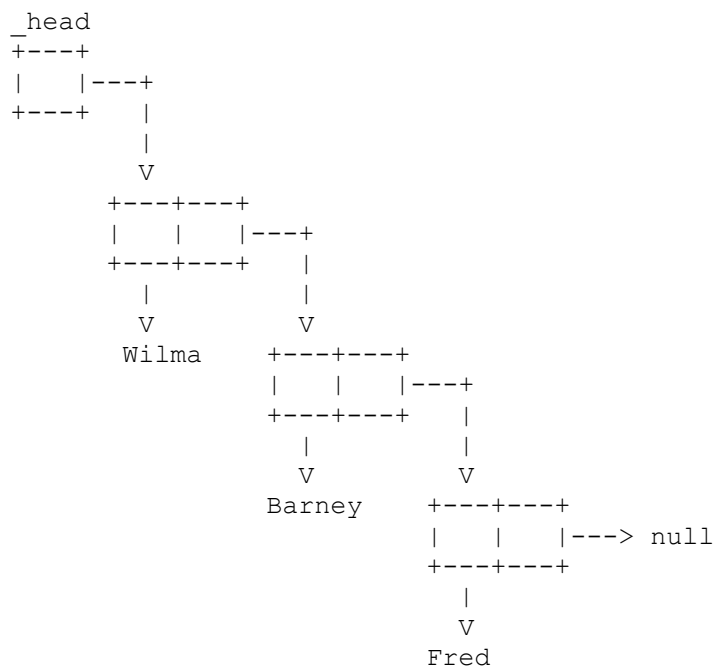
```

```

public Object deleteEnd()

```

--- Back to Front ----



```

public Object deleteEnd()
{
    if ( _head == null ) return null;

    Link prev = null;
    Link curr = _head;
    while ( curr.next != null )
    {
        prev = curr;
        curr = curr.next )
    }
    if ( prev != null )
        prev.next = null;
    else
        _head = null;

    return curr.data;
}

```

--- Front to Back ---

```

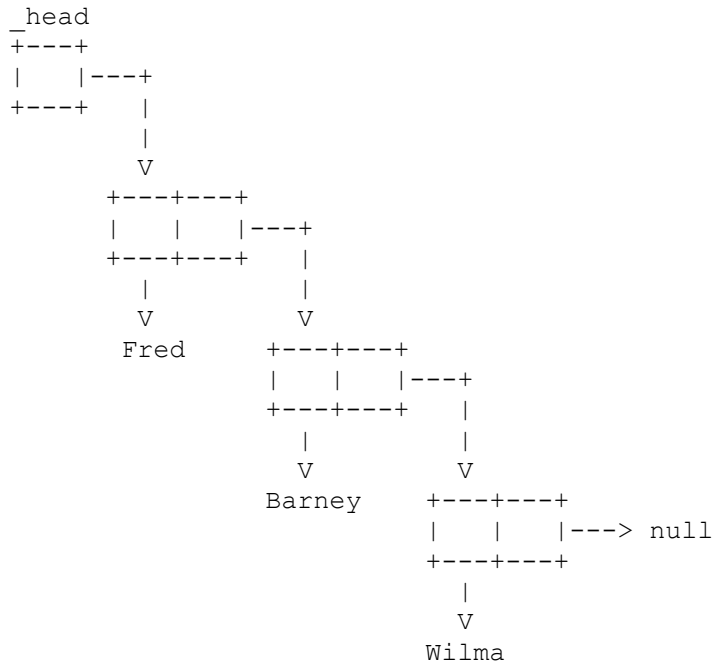
head
+----+
|    |-----+
+----+      |
              |
              V
+---+-----+
|    |     |-----+
+---+-----+   |
                |
                V
                Fred
+---+-----+
|    |     |-----+
+---+-----+   |
                |
                V
                Barney
+---+-----+
|    |     |-----> null
+---+-----+
                |
                V
                Wilma

```

```
public void insertOnEnd( Object item )
{
```



--- Front to Back ----



```

public void insertOnEnd( Object item )
{
    Link add = new Link();
    add.data = item;
    add.next = null;

    if ( _head == null )
        _head = add;
    else
    {
        Link curr = _head;
        while ( curr.next != null )
            curr = curr.next );
        curr.next = add;
    }
}

```

```

// Or, an alternate form...
// for ( Link curr = _head;
//       curr.next != null;
//       curr = curr.next );
// curr.next = add;

```

