```
package queue;

class DQCell
{
  Object value;
  DQCell prev;
  DQCell next;

  DQCell(Object value)
  {
    this.value = value;
    prev = null;
    next = null;
  }

  DQCell(Object value, DQCell prev, DQCell next)
  {
    this.value = value;
    this.prev = prev;
    this.next = next;
  }
};
```

```
public class DQueue implements Queue, ReverseQueue
{
  private DQCell first = new DQCell(null);
  private DQCell last = new DQCell(null);

  public DQueue()
  {
    first.next = last;
    last.prev = first;
  }

  public boolean
  empty()
  {
    return first.next == last;
  }

  public void
  enqStart(Object x)
  {
    DQCell cell = new DQCell(x, first, first.next);
    first.next.prev = cell;
    first.next = cell;
  }
```

```
  public Object
  deq()
  {
    Object temp = first.next.value;
    first.next.next.prev = first;
    first.next = first.next.next;
    return temp;
  }

  public void
  enq(Object x)
  {
    DQCell cell = new DQCell(x, last.prev, last);
    last.prev.next = cell;
    last.prev = cell;
  }

  public Object
  deqEnd()
  {
    Object temp = last.prev.value;
    last.prev.prev.next = last;
    last.prev = last.prev.prev;
    return temp;
  }
};
```