```
class Qcell
{
    Object value;
    Qcell next;

    // implement cons
    Qcell(Object value, Qcell next)
    {
        this.value = value;
        this.next = next;
    }
};

class Queue
{
    private Qcell tail = null;

    public boolean
    empty()
    {
        return tail == null;
    }

    public void
    enq(Object x)
    {
        if(tail == null)
            {
                tail = new Qcell(x, null);
                // make it circular
                tail.next = tail;
            }
        else
            {
                tail.next = new Qcell(x, tail.next);
                tail = tail.next;
            }
    }

    public Object
    deq(Queue q)
    {
        Object temp = tail.next.value;
        if(tail.next == tail)
            tail = null;
        else
            tail.next = tail.next.next;
        return temp;
    }
};
```



```
Interface:      2p
   Types et nombre des paramètres
   Vraiment impératif

Implémentation: 2p
   Utilisation d'une liste circulaire
   Respect de l'interface
   Pas de sentinelle

Génie Logiciel: 1p
   Code Bref
   Nom des variables
   etc
```