The Algorithm must be written in general

Question 11.

Link is an entity which can hold a maximum of 100 integers. Link enables the user to add elements from the rear end and remove integers from the front end of the entity. Define a class Link with the following details : **[10]**

Class name : ***Link***

Data Members/instant variables :

lnk [ ]	:	*entity to hold the integer elements.*
max	:	*stores the maximum capacity of the entity*
begin	:	*to point to the index of the front end.*
end	:	*to point to the index of the rear end.*

Member functions :

Link (int mm)	:	*constructor to initialize max = mm, begin = 0, end = 0.*

void addlink (int v): *to add an element from the rear index if possi-ble otherwise display the message "OUT OF SIZE..."*

int dellink ( ) : *to remove and return an element from the front index, if possible otherwise display the message "EMPTY..." and return –99*

void display( ) : *displays the elements of the entity.*

(a) *Specify the class Link giving details of the* ***constructor (int), void add-link (int), int dellink ( ) and void display ( ).*** **[9]**

THE MAIN FUNCTION AND ALGORITHM NEED NOT BE WRITTEN.

(b) *What type of data structure is the above entity ?* **[1]**

Answer 11.

(a)
```
class Link
{
  int lnk[]=new int[100];
  int begin,end,max;

  public Link(int mm)
  {
    max=mm;
    begin=end=0;
  }
  void addlink(int v)
  {
    if(end==0)
    {
     end=begin=1;
     lnk[end]=v;
    }
    else if(end==max)
    {
     System.out.println('List is
     Full');
    }
    else
    {
     lnk[++end]=v;
    }
  }
  int dellink()
  {
    int a;
    if(begin==0)
    {
    System.out.println("Empty...");
     return(-99);
    }
    else if(begin==end)
    {
      a=lnk[begin];
      begin=end=0;
      return a;
    }
    else
    {
      a=lnk[begin];
      begin++;
      return(a);
    }
  }
  void display()
  {
    int i;
    for(i=begin;i<=end;i++)
    {
      System.out.println(lnk[i]);
    }
  }
}
```

(b) It is a queue.

Question 12.

A super class Detail has been defined to store the details of a customer. Define a sub-class Bill to compute the monthly telephone charge of the customer as per the chart given below :

Number of Calls	*Rate*
1–100	*Only rental charge*
101–200	*60 paisa per call + rental charge*
201–300	*80 paisa per call + rental charge*
Above 300	*1 rupee per call + rental charge*