********** OUTPUT **********

```
Length of aisle is: 9
This is the original aisle: [0, 1, 1, 0, 1, 0, 0, 0, 1]
A technique of closest neighbouring people from start and end of aisle will determine
whether to move people left or right

****Calculating proximity of two people from left hand side****
this is value of person 0 :0
this is value of person 1 :1
this is value of person 2 :1
this is value of person 3 :0

Distance is:-1

****Calculating proximity of two people from right hand side****
this is value of person 8 :1
this is value of person 7 :0
this is value of person 6 :0
this is value of person 5 :0
this is value of person 4 :1
this is value of person 3 :0

Distance is:3


***VERDICT: People to move left

Total number of moves left: 9
[1, 1, 1, 1, 0, 0, 0, 0, 0]

Total number of moves right: 20
[0, 0, 0, 0, 0, 1, 1, 1, 1]
```

**** CODE ****

```
/*
Online Java - IDE, Code Editor, Compiler


Online Java is a quick and easy tool that helps you to build, compile, test your programs online.
*/
import java.util.Arrays;


public class Main
```

```
{
   public static void main(String[] args) {
      shiftPeople sp = new shiftPeople();

      sp.leftOrRight();

      sp.beginMove();


      }

}


class shiftPeople

{
      int count=0;

      int  distanceLeft=0;

      int distanceRight=0;

      int temp;

      int pos[]=new int[3];


      int[] people = new int[]{0, 1, 1, 0, 1, 0, 0, 0, 1};

      int length=people.length;


   public shiftPeople()

   {


      System.out.println("Length of aisle is:  " + length);
      System.out.println("This is the original aisle:  " + Arrays.toString(people));




       // It is realised that logic is that from LHS and RHS

      // LHS - closest distance between furthest left 1 and its closest one

      // RHS - closest distance between furthest right 1 and its closest one

      //Whichever is smallest, this is direction of movement of people.
```

```
System.out.println("A technique of closest neighbouring people from start and end of aisle will determine");

System.out.println("whether to move people left or right");


//this deals with LHS

System.out.println("\n****Calculating proximity of two people from left hand side****");

for (int i=0;i<length;i++)

{

   System.out.println("this is value of person " + i + " :" + people[i]);

     if (count==2)

      {

         break;


      }


   if (people[i]==1)

   {

      count++;


      //System.out.println("index position of people: " + i);

      pos[count]=i;


   }


}

distanceLeft = pos[2]-pos[1]-1;

System.out.println("\nDistance is:" + distanceLeft + "\n");


count=0;

System.out.println("****Calculating proximity of two people from right hand side****");

  for (int i=length-1;i>-1;i--)

{
```

```
        System.out.println("this is value of person " + i + "  :" + people[i]);
          if (count==2)
           {
              break;


           }


        if (people[i]==1)
        {
           count++;
           pos[count]=i;
           distanceRight = pos[1]-pos[2]-1;


        }
    }
    System.out.println("\nDistance is:" + distanceRight + "\n");


}


public void leftOrRight()
{
    if (distanceLeft>distanceRight)
    {
       System.out.println("\n***VERDICT:  " + "People to move right");
    }



    else if (distanceRight>distanceLeft)
    {
       System.out.println("\n***VERDICT:  " + "People to move left");
    }


    else
```

```
    {
       System.out.println("\n***VERDICT:   " + "no difference moving left or right");
    }

}

public void beginMove()
{
    int moves=0;
    int counter=0;
    int [] original = new int [length];
    original=people;

    do
    {

    for  (int i=length-1;i>0;i--)
    {
       if ((people[i-1]==0) && (people[i]==1))
       {
           people[i]=0;
          people[i-1]=1;
             moves++;
       }
    }
    counter++;
    }while(counter<length-1);

    System.out.println("\nTotal number of moves left: " + moves);
    System.out.println(Arrays.toString(people));

    people=original;
```

```
        moves=0;
        counter=0;



        do
        {

        for (int i=0;i<length-1;i++)
        {
           if ((people[i+1]==0) && (people[i]==1))
           {
              people[i]=0;
              people[i+1]=1;
              moves++;
           }
        }
        counter++;
        }while(counter<length-1);

        System.out.println("\nTotal number of moves right: " + moves);
        System.out.println(Arrays.toString(people));









        }
}
```