10. Write Java programs to
(a) Implement All-Pairs Shortest Paths problem using **Floyd's algorithm**.

a)

```
import java.util.Scanner;
public class FloydsDemo
{
  public static void main(String [] args)
  {
    int d [][] = new int[20][20]; // Temporary 2-D array for calculation
    int w [][] = new int[20][20]; // 2-D array to store weight/cost matrix
    int i,j,n;

    Scanner in = new Scanner(System.in);

    System.out.println("Enter the number of nodes/vertices");
    n = in.nextInt();

    System.out.println("Enter the Weight/cost matrix");
    for(i=1;i<=n;i++)
    {
     for(j=1;j<=n;j++)
     {
      w[i][j] = in.nextInt();
     }
    }

    floyd(n,d,w);
  }

public static void floyd(int n ,int d[][],int w[][])
{
    int i,j,k;

    for(i=1;i<=n;i++)
    {
     for(j=1;j<=n;j++)
     {
      d[i][j] = w[i][j];
     }
  }
// Calculation of All-pair shortest paths
    for(k=1;k<=n;k++)
    {
     for(i=1;i<=n;i++)
     {
      for(j=1;j<=n;j++)
      {
       d[i][j] = min(d[i][j],d[i][k]+d[k][j]);
     }
    }
  }

   System.out.println("All pair shortest path matrix is");
   for(i=1;i<=n;i++)
   {
    System.out.print("\t"+i );
   }

   System.out.println("\n \t------------------------------------------");
   for(i=1;i<=n;i++)
   {
    System.out.print(i + "|"+ "\t");
     for(j=1;j<=n;j++)
     {
      System.out.print(d[i][j] + "\t");
     }
    System.out.print("\n");
   }
  }
  // Method to find the minimum of two numbers
  public static int min(int a,int b)
  {
   if(a<b)
       return a;

   else
       return b;
  }
}
```