

Tsp

```
import java.util.Arrays;
import java.util.Scanner;

class tspp
{
    static int[][] distance=new int[50][50];
    static int[][] memo = new int[50][50];
    static int n;

    static int tsp(int mask,int pos)
    {
        if(mask == (1 << n) -1)
        {
            return distance[pos][0];
        }
        if(memo[mask][pos]!= -1)
        {
            return memo[mask][pos];
        }
        int mincost=Integer.MAX_VALUE;
        for (int city=0;city<n;city++)
        {
            if((mask &(1<< city))==0)
            {
                int newcost=distance[pos][city]+tsp(mask |(1<<city),city);
                mincost=Math.min(mincost,newcost);
            }
        }
        return memo[mask][pos]=mincost;
    }

    public static void main(String[] args) {
```

```

Scanner sc=new Scanner(System.in);

System.out.print("Enter the number of cities: ");

    n=sc.nextInt();

    distance = new int[n][n] ;

System.out.print("Enter the distance between cities: \n");
for(int i=0;i<n;i++){
for(int j=0;j<n;j++){
distance[i][j]=sc.nextInt();
}
}

    memo = new int[1 << n][n];
    for (int[] row : memo) {
        Arrays.fill(row, -1);
    }

    int minCost = tsp(1, 0);

    System.out.println("Minimum cost to visit all cities: " + minCost);

    }
}

```