NAME:	Shubham Vishwakarma
UID No.	2021700071
BRANCH:	S.Y CSE-DS
BATCH:	D
SUBJECT	Design and Analysis of Algorithms
EXPERIMENT No.	9
Date of Performance	18/04/2023
Date of Submission	23/04/2023

AIM:	**Approximation algorithms (Travelling Salesman Problem)**
Program 1	
PROBLEM STATEMENT :	Implement the Travelling Salesman algorithm for the given scenario.

ALGORITHM / THEORY:	**Algorithm 1:** Dynamic Approach for TSP
	Data: s: starting point; N: a subset of input cities; $dist()$: distance among the cities
	Result: $Cost$: TSP result
	$Visited[N] = 0$;
	$Cost = 0$;
	Procedure TSP(N, s)
	$Visited[s] = 1$;
	if $\|N\| = 2$ *and* $k \neq s$ **then**
	$Cost(N, k) = dist(s, k)$;
	Return Cost;
	else
	for $j \in N$ **do**
	for $i \in N$ *and* $visited[i] = 0$ **do**
	if $j \neq i$ *and* $j \neq s$ **then**
	$Cost(N, j) = \min (TSP(N - \{i\}, j) + dist(j, i))$
	$Visited[j] = 1$;
	end
	end
	end
	end
	Return $Cost$;
	end

PROGRAM:

```
#include<stdio.h>

int ary[10][10],completed[10],n,cost=0;

void takeInput()
{
int i,j;

printf("Enter the number of villages: ");
scanf("%d",&n);

printf("\nEnter the Cost Matrix\n");

for(i=0;i < n;i++)
{
printf("\nEnter Elements of Row: %d\n",i+1);

for( j=0;j < n;j++)
scanf("%d",&ary[i][j]);

completed[i]=0;
}

printf("\n\nThe cost list is:");

for( i=0;i < n;i++)
{
printf("\n");

for(j=0;j < n;j++)
printf("\t%d",ary[i][j]);
}
}

void mincost(int city)
{
int i,ncity;

completed[city]=1;

printf("%d--->",city+1);
ncity=least(city);

if(ncity==999)
{
ncity=0;
printf("%d",ncity+1);
```

```
cost+=ary[city][ncity];

return;
}

mincost(ncity);
}

int least(int c)
{
int i,nc=999;
int min=999,kmin;

for(i=0;i < n;i++)
{
if((ary[c][i]!=0)&&(completed[i]==0))
if(ary[c][i]+ary[i][c] < min)
{
min=ary[i][0]+ary[c][i];
kmin=ary[c][i];
nc=i;
}
}

if(min!=999)
cost+=kmin;

return nc;
}

int main()
{
takeInput();

printf("\n\nThe Path is:\n");
mincost(0); //passing 0 because starting vertex

printf("\n\nMinimum cost is %d\n ",cost);

return 0;
}
```

RESULT:

```
PS C:\Users\smsha\Desktop\SEM 4\DAA\Practicals\Exp9\output> & .\'tsp.exe'
Enter the number of villages: 4

Enter the Cost Matrix

Enter Elements of Row: 1
0 4 1 3

Enter Elements of Row: 2
4 0 2 1

Enter Elements of Row: 3
1 2 0 5

Enter Elements of Row: 4
3 1 5 0


The cost list is:
        0       4       1       3
        4       0       2       1
        1       2       0       5
        3       1       5       0

The Path is:
1--->3--->2--->4--->1

Minimum cost is 7
```

CONCLUSION :	The traveling salesman problem is a permutation problem in which the goal is to find the shortest path between N different cities that the salesman takes is called the TOUR. In other words, **the problem deals with finding a route covering all cities so that the total distance traveled is minimal**.