

**Aim:**

Write a C program to implement **Travelling Sales Person** problem using **Dynamic programming**.

**Source Code:**

ISP.c

```
#include<stdio.h>
int ary[10][10], completed[10], n, cost = 0;
void takeInput()
{
    int i, j;
    printf("Number of villages: ");
    scanf("%d", & n);
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
            scanf("%d", & ary[i][j]);
        completed[i] = 0;
    }
    printf("The 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("\nThe Path is:\n");
    mincost(0);
    printf("\nMinimum cost is %d", cost);
    return 0;
}

```

### Execution Results - All test cases have succeeded!

| Test Case - 1         |    |    |  |
|-----------------------|----|----|--|
| User Output           |    |    |  |
| Number of villages: 3 |    |    |  |
| 0 10 15               |    |    |  |
| 10 0 35               |    |    |  |
| 15 35 0               |    |    |  |
| The cost list is:     |    |    |  |
| 0                     | 10 | 15 |  |
| 10                    | 0  | 35 |  |
| 15                    | 35 | 0  |  |
| The Path is:          |    |    |  |
| 1-->2-->3-->1         |    |    |  |
| Minimum cost is 60    |    |    |  |