

```

#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <stdlib.h>
#define INF 987654321
#define MIN(a, b) (((a) < (b)) ? (a) : (b))
void travel();
void tsp();
int W[10][10], D[10][10], P[10][10], cnt=0, n, sub_cnt, minlength;
int main()
{
    int i, j, k=0;
    printf("n의 개수를 입력해주세요 : ");
    scanf("%d", &n);
    sub_cnt = (1 << (n - 1));
    minlength = -1;
    cnt = sub_cnt;
    for (i = 1; i <= n; i++)
    {
        for (j = 0; j < sub_cnt; j++)
            P[i][j] = 0;
    }
    printf("\n그래프의 인접 행렬식 가중치를 입력해주세요. (무한대는 0를 입력해주세요)\n");
    for (i = 1; i <= n; i++)
    {
        printf("%d행의 데이터 입력 \n", i);
        for (j = 1; j <= n; j++)
        {
            int w;
            scanf("%d", &w);
            if (w == 0)
                w = INF;
            W[i][j] = w;
        }
    }
    printf("인접행렬식 표현 : \n");
    for (i = 1; i <= n; i++)
    {
        printf("\n\n");
        for (j = 1; j <= n; j++)
            if (W[i][j] == INF)
                printf("\tINF");
    }
}

```

```

        else
            printf("\t%d", W[i][j]);
    }

    travel();
    printf("\n최단 경로의 길이 : %d\n", minlength);
    printf("D행렬 값 : \n");
    for (int i = 1; i <= n; i++) {
        for (int j = 0; j < sub_cnt; j++) {
            if (D[i][j] == INF)
                printf("0\t");
            else
                printf("%d\t", D[i][j]);
        }
        printf("\n");
    }
    printf("\nP행렬 값 : \n");
    for (int i = 1; i <= n; i++) {
        for (int j = 0; j < sub_cnt; j++) {
            if (P[i][j] == 0)
                printf("0\t");
            else
                printf("V%d\t", P[i][j]);
        }
        printf("\n");
    }
    printf("\n최적경로 : \n");
    int A = cnt - 1, fin = 0;
    i = 1;
    printf("v1 ");
    while (k < n - 1) {
        fin = P[i][A];
        printf("v%d ", fin);
        i = fin;
        if (fin == 2)
            A = A - 1;
        else
            A = A - (fin - 2 << 1);
        k++;
    }
    printf("v1");

```

```
}
```

```
void travel()
```

```
{
```

```
    int i, j, k, A;
```

```
    int sub_cnt = 1 << (n - 1);
```

```
    for (i = 1; i <= n; i++)
```

```
    {
```

```
        for (j = 0; j < (1 << (n - 1)); j++)
```

```
            D[i][j] = INF;
```

```
    }
```

```
    for (i = 2; i <= n; i++)
```

```
        D[i][0] = W[i][1];
```

```
    for (k = 1; k <= n - 2; k++)
```

```
    {
```

```
        tsp();
```

```
    }
```

```
    int pj = 0;
```

```
    for (j = 2; j <= n; j++)
```

```
    {
```

```
        int _visit = sub_cnt - 1 - (1 << (j - 2));
```

```
        if (W[1][j] + D[j][_visit] < D[1][sub_cnt - 1])
```

```
        {
```

```
            D[1][sub_cnt - 1] = W[1][j] + D[j][_visit];
```

```
            pj = j;
```

```
        }
```

```
    }
```

```
    P[1][sub_cnt - 1] = pj;
```

```
    minlength = D[1][sub_cnt - 1];
```

```
}
```

```
void tsp() {
```

```
    int rep;
```

```
    for (int i = 2; i <= n; i++) {
```

```
        for (int A = 1; A < cnt; A++) {
```

```
            rep = 0;
```

```

        if ((A & (1 << (i - 2))) == 0) {
            for (int j = 2; j <= n; j++) {
                if ((A & (1 << (j - 2))) != 0) {
                    int fin = A - (1 << (j - 2));
                    D[i][A] = MIN(W[i][j] + D[j][fin], D[i][A]);
                    rep = j;
                }
            }
            P[i][A] = rep;
        }
    }
}
}
}

```

Microsoft Visual Studio 디버그 콘솔

n의 개수를 입력해주세요 : 4

그래프의 인접 행렬식 가중치를 입력해주세요. (무한대는 0을 입력해주세요)

1행의 데이터 입력
0 2 9 0

2행의 데이터 입력
1 0 6 4

3행의 데이터 입력
0 7 0 8

4행의 데이터 입력
6 3 0 0

인접행렬식 표현 :

	INF	2	9	INF
1	INF	6	4	
INF	7	INF	8	
6	3	INF	INF	

최단 경로의 길이 : 21

0행렬 값 :

0	0	0	0	0	0	21
1	0	0	0	10	0	0
0	8	0	0	14	12	0
6	4	0	0	0	0	0

P행렬 값 :

0	0	0	0	0	0	V3
0	0	V3	0	V4	0	0
0	V2	0	0	V4	V4	0
0	V2	V3	V3	0	0	0

최적경로 :
v1 v3 v4 v2 v1

C:\Users\com\source\repos\문제_동적계획법\Debug\문제_동적계획법.exe(프로세스 23504개)이(가) 종료되었습니다
(코드: 0개).

디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지되면 자동으로 콘솔 닫기]를 사용
하도록 설정합니다.

이 창을 닫으려면 아무 키나 누르세요...

```
Microsoft Visual Studio 디버그 콘솔
n의 개수를 입력해주세요 : 4
그래프의 인접 행렬식 가중치를 입력해주세요. (무한대는 0을 입력해주세요)
1행의 데이터 입력
0 10 15 20
2행의 데이터 입력
5 0 9 10
3행의 데이터 입력
6 13 0 12
4행의 데이터 입력
8 8 9 0
인접행렬식 표현 :
      INF    10    15    20
      5      INF    9    10
      6      13    INF    12
      8      8     9    INF
최단 경로의 길이 : 35
D행렬 값 :
0      0      0      0      0      0      35
5      0      15     0      18     0      25     0
6      18     0      0      20     25     0      0
8      13     15     23     0      0      0      0
P행렬 값 :
0      0      0      0      0      0      V2
0      0      V3     0      V4     0      V4     0
0      V2     0      0      V4     V4     0      0
0      V2     V3     V3     0      0      0      0
최적 경로 :
v1 v2 v4 v3 v1
C:\Users\com\source\repos\외판원_문제_동적계획법\Debug\외판원_문제_동적계획법.exe(프로세스 4564개)이(가) 종료되었습니다(
코드: 0개).
```