

33.

```
#include <stdio.h>

#define INF 9999

#define MAX 10

void dijkstra(int graph[MAX][MAX], int n, int start) {
    int dist[MAX], visited[MAX] = {0}, i, j, u, min;

    for (i = 0; i < n; i++)
        dist[i] = INF;

    dist[start] = 0;

    for (i = 0; i < n - 1; i++) {
        min = INF;

        for (j = 0; j < n; j++)
            if (!visited[j] && dist[j] < min)
                min = dist[u = j];

        visited[u] = 1;

        for (j = 0; j < n; j++)
            if (graph[u][j] && !visited[j] && dist[u] + graph[u][j] < dist[j])
                dist[j] = dist[u] + graph[u][j];
    }

    printf("Shortest distances from node %d:\n", start);

    for (i = 0; i < n; i++)
        printf("To %d: %d\n", i, dist[i]);
}

int main() {
    int graph[MAX][MAX], n;

    printf("Enter number of vertices: ");

    scanf("%d", &n);

    printf("Enter adjacency matrix (0 for no edge):\n");

    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
```

```
Enter number of vertices: 3
Enter adjacency matrix (0 for no edge):
0 5 0 10
0 0 3 0
0 0 0 1
0 0 0 0
Shortest distances from node 0:
To 0: 0
To 1: 5
To 2: 9999
```

```
        scanf("%d", &graph[i][j]);  
dijkstra(graph, n, 0);  
return 0;  
}
```