```c
#include <stdio.h>

#define INF 9999
#define MAX 100

void dijkstra(int graph[MAX][MAX], int n, int start) {
    int distance[MAX], visited[MAX], i, j, min, next;

    for (i = 0; i < n; i++) {
        distance[i] = INF;
        visited[i] = 0;
    }

    distance[start] = 0;

    for (i = 0; i < n - 1; i++) {
        min = INF;
        for (j = 0; j < n; j++) {
            if (!visited[j] && distance[j] < min) {
                min = distance[j];
                next = j;
            }
        }

        visited[next] = 1;

        for (j = 0; j < n; j++) {
            if (!visited[j] && graph[next][j] && distance[next] + graph[next][j] < distance[j]) {
                distance[j] = distance[next] + graph[next][j];
            }
        }
    }

    printf("Vertex\tDistance from Source\n");
    for (i = 0; i < n; i++) {
        printf("%d\t%d\n", i, distance[i]);
    }
}

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

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

    printf("Enter adjacency matrix (use 0 for no edge):\n");
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
```

```c
        scanf("%d", &graph[i][j]);

    printf("Enter starting vertex: ");
    scanf("%d", &start);

    dijkstra(graph, n, start);

    return 0;
}
```

Enter number of vertices: 5
Enter adjacency matrix (use 0 for no edge):
0 10 0 30 100
0 0 50 0 0
0 0 0 0 10
0 0 20 0 60
0 0 0 0 0
Enter starting vertex: 0
Vertex  Distance from Source
0       0
1       10
2       50
3       30
4       60

--------------------------------
Process exited after 21.47 seconds with return value 0
Press any key to continue . . .