#include<stdio.h>

#define LARGE 10000

#define MAX 30

Struct state {

Int len, pre, label;

};

Struct state states[MAX];

Int a[MAX][MAX];

Int main() {

Int i, j, s, d, n, tem, min, mini;

Printf(“Enter no.of vertices: “);

Scanf(“%d”, &n);

Printf(“\nEnter adjacency matrix\n”);

For (i = 0; i < n; i++)

For (j = 0; j < n; j++)

Scanf(“%d”, &a[i][j]);

For (i = 0; i < n; i++) {

States[i].len = LARGE;

States[i].pre = -1;

States[i].label = 0;

}

Printf(“\nEnter source vertex: “);

Scanf(“%d”, &s);

Printf(“\nEnter destination vertex: “);

Scanf(“%d”, &d);

States[d].len = 0;

States[d].label = 1;

Tem = d;

While (tem != s) {

For (i = 0; i < n; i++) {

If (a[tem][i] != 0 && states[tem].len + a[tem][i] < states[i].len && states[i].label == 0) {

States[i].len = states[tem].len + a[tem][i];

States[i].pre = tem;

}

}

Min = LARGE;

For (i = 0; i < n; i++) {

If (states[i].len < LARGE && states[i].label == 0) {

Min = states[i].len;

Mini = i;

}

}

States[mini].label = 1;

Tem = mini;

}

Tem = s;

Printf(“\nPath length: %d”, states[s].len);

Printf(“\nPath\n”);

Printf(“%d”, tem);

Do {

Tem = states[tem].pre;

Printf(“ -> %d”, tem);

} while (tem != d);

Return 0;

}