int G[N][N]=

{

{0,1,9,0,0,0},

{0,0,7,2,0,0},

{0,0,0,0,1,0},

{0,0,4,0,8,2},

{0,0,0,0,0,5},

{0,0,0,0,4,0}

};

void WriteGraph()

{

int i,j;

for( i = 0; i < N; i++)

{

for( j = 0; j < N; j++)

printf("%i ", G[i][j]);

printf("\n");

}

printf("\n");

}

void WritePath(int\* T, int Start, int V)

{

if(V!=Start)

WritePath(T,Start,T[V]);

printf("%i ",V+1);

}

void Dijkstra(int start,int \*T,int \*D)

{

int minn, index,i;

int\* V;

V = (int\*)malloc(N\*sizeof(int));

for(i = 0; i < N; i++)

{

V[i] = 0;

D[i] = INT\_MAX;

T[i] = -1;

}

T[start] = 0;

D[start] = 0;

index=start;

int f=1;

while (f)

{

f=0;

minn = INT\_MAX;

for(i = 0; i < N; i++)

if((V[i] == 0) && (D[i] <= minn))

{

minn = D[i];

index = i;

}

V[index] = 1;

for(i = 0; i < N; i++)

if((V[i] == 0) && (G[index][i] != 0) && (D[index] != INT\_MAX) && ((D[index] + G[index][i]) < D[i]))

{

D[i] = D[index] + G[index][i];

T[i] = index;

f=1;

}

for(i=0; i<N; i++)

if (!V[i]) f=1;

}

}

int main()

{

setlocale(LC\_ALL, "Russian");

int i,T1[N],T2[N],D1[N],D2[N];

int v1,v2;

printf("Введите вершину v1:");

scanf("%d",&v1);

printf("Введите вершину v2:");

scanf("%d",&v2);

v1--;

v2--;

WriteGraph();

Dijkstra(v1,T1,D1);

Dijkstra(v2,T2,D2);

for(i=0;i<N;i++){

if(D1[i]>D2[i]&&D1[i]!=INT\_MAX&&D2[i]!=INT\_MAX&&i!=v1&&i!=v2){

printf("Вершина: %d\n",i+1);

printf("Путь от v1:");

WritePath(T1,v1,i);

printf("Длина от v1: %d\n",D1[i]);

printf("Путь от v2:");

WritePath(T2,v2,i);

printf("Длина от v2: %d\n\n",D2[i]);

}

}

return 0;

}