

```
1  #include "const.h"
2  #include "graph.h"
3  #include "routage.h"
4
5  #include <stdlib.h> // rand ()
6
7  #include <stdio.h> // printf ()
8
9  #define inf 9999
10
11
12  routage* init(graphe* G, int taille) {
13      int i, j;
14      routage* R = malloc(sizeof(routage));
15
16
17      for(i = 0; i < taille; ++i)
18      {
19          for(j = 0; j < taille; ++j)
20          {
21              R->poids[i][j] = G->list[i][j];
22
23              if(R->poids[i][j] == -1)
24              {
25                  R->poids[i][j] = inf;
26                  R->succ[i][j] = -1;
27              }
28              else
29              {
30                  R->succ[i][j] = j;
31              }
32          }
33      }
34
35      for(i = 0; i < taille; ++i)
36      {
37          R->poids[i][i] = 0;
38          R->succ[i][i] = i;
39      }
40      return R;
41  }
42
43  void Floyd_Warshall(routage* R, int taille) {
44      int i, j, k;
45
46      for(k = 0; k < taille; k++)
47      {
48          for(i = 0; i < taille; i++)
49          {
50              for(j = 0; j < taille; j++)
51              {
52                  if(R->poids[i][k] != inf && R->poids[k][j] != inf
53                     && (R->poids[i][j] > (R->poids[i][k] + R->poids[k][j])) )
54                  {
55                      R->poids[i][j] = R->poids[i][k] + R->poids[k][j];
56                      R->succ[i][j] = R->succ[i][k];
57                  }
58              }
59          }
60      }
61  }
```

```
58
59     }
60 }
61 }
62 }
63
64 void afficher_chemin(routage* R, int deb, int fin) {
65     int stock_deb = deb;
66     int voisin[TAILLE_GRAPHE] = {-1};
67
68     int i, suiv = R->succ[deb][fin];
69     for(i = 0; suiv != fin && i < TAILLE_GRAPHE; i++)
70     {
71         deb = R->succ[deb][fin];
72         voisin[i] = deb;
73         suiv = R->succ[deb][fin];
74     }
75
76     if(suiv == fin)
77     {
78         voisin[i] = fin;
79
80         printf("\nChemin de %d à %d :\n%d", stock_deb, fin, stock_deb);
81         for(int j = 0 ; j < i+1; j++)
82         {
83             printf(" -> %d", voisin[j]);
84         }
85         printf("\n");
86     }
87     else
88     {
89         printf("error\n");
90     }
91 }
92
93 void libere_routage(routage* R) { free(R); }
94
```