Code::Blocks | Search results | Cccc | Build log | Build messages

Start here | FLOYD'S ALGORITHM.c

```c
#include <stdio.h>
int a[10][10],d[10][10],n;

int min(int a,int b)
{
    return (a<b?a:b);
}

void floyd()
{
    int i,j,k;
    for (i=0;i<n;i++)
    {
        for (j=0;j<n;j++)
        {
            d[i][j]=a[i][j];
        }
    }
    for (k=0;k<n;k++)
    {
        for (i=0;i<n;i++)
        {
            for (j=0;j<n;j++)
            {
                d[i][j]=min(d[i][j],d[i][k]+d[k][j]);
            }
        }
    }
}

int main()
{
    int i,j;
    printf("Enter the number of vertices:\n");
    scanf("%d",&n);
    printf("Enter the adjacency matrix:\n");
    for (i=0;i<n;i++)
    {
        for (j=0;j<n;j++)
        {
            scanf("%d",&a[i][j]);
```

Logs & others

Code::Blocks | Search results | Cccc | Build log | Build messages

Start here | FLOYD'S ALGORITHM.c

```
16              d[i][j]=a[i][j];
17          }
18      }
19      for (k=0;k<n;k++)
20      {
21          for (i=0;i<n;i++)
22          {
23              for (j=0;j<n;j++)
24              {
25                  d[i][j]=min(d[i][j],d[i][k]+d[k][j]);
26              }
27          }
28      }
29  }
30
31  int main()
32  {
33      int i,j;
34      printf("Enter the number of vertices:\n");
35      scanf("%d",&n);
36      printf("Enter the adjacency matrix:\n");
37      for (i=0;i<n;i++)
38      {
39          for (j=0;j<n;j++)
40          {
41              scanf("%d",&a[i][j]);
42          }
43      }
44      floyd();
45      printf("The distance matrix:\n");
46      for (i=0;i<n;i++)
47      {
48          for (j=0;j<n;j++)
49          {
50              printf("%d ",d[i][j]);
51          }
52          printf("\n");
53      }
54      return 0;
55  }
56
```