题目：算法设计与分析作业

作者：数学强基 2301 刘欣楠

关键词：算法、作业

目　录

1 第四次作业

1.1 习题 5-4

解. 以每个点为初始团个进行依次搜索, 每次搜索遍历与最后一个加入结点相邻的所有节点，如果存在加入后仍为团的相邻节点, 则加入, 并从该点继续搜索, 反之回溯. □

1.2 习题 5-6

```
#include<stdio.h>
#define INF -1
int n,length = 0,shortest_length = INF,**graph,*path,*best_path;

void swap(int *array, int i, int j)
{
    int temp;
    temp = array[i];
    array[i] = array[j];
    array[j] = temp;
}

void output()
{
    printf("最短路径长度是%d\n", shortest_length);
    printf("此时路径为: %d", best_path[1]);
    for (int i = 2; i < n + 1; i++)
    printf("-->%d", best_path[i]);
    printf("-->%d\n", best_path[1]);
}

void travel(int t)
{
    if (t == n)
    {
        if (graph[path[t - 1]][path[t]] != INF
        && graph[path[t]][1] != INF
        && (length + graph[path[t - 1]][path[t]] + graph[path[t]][1] < shortest_length ||
            shortest_length == INF))
        {
            for (int i = 0; i < n + 1; i++)
            best_path[i] = path[i];
            shortest_length = length + graph[path[t - 1]][path[t]] + graph[path[t]][1];
        }

        return;
    }

    for (int i = t; i < n + 1; i++)
    {
        if (graph[path[t - 1]][path[i]] != INF
        && (length + graph[path[t - 1]][path[i]] < shortest_length || shortest_length ==
            INF))
        {
            swap(path, i, t);
```

```
            length += graph[path[t - 1]][path[t]];
            travel(t + 1);
            length -= graph[path[t - 1]][path[t]];
            swap(path, i, t);
        }
    }
}

int main()
{
    int start,m;

    cin>>n>>m;
    graph = (int**)malloc(sizeof(int*) * (n + 1));
    for (int i = 0; i <= n; i++)
    graph[i] = (int*)malloc(sizeof(int) * (n + 1));

    path = (int*)malloc(sizeof(int)*(n + 1));

    best_path = (int*)malloc(sizeof(int)*(n + 1));

    for (int i = 0; i < n + 1; i++)
    {
        path[i] = i;

        for (int j = 0; j < n + 1; j++)
        graph[i][j] = INF;
    }
    for(int i=1,u,v,w;i<=m;i++)
    {
        cin>>u>>v>>w;
        graph[u][v]=graph[v][u]=w;
    }
    printf("请输入以第几个结点作为起点：");
    scanf("%d", &start);
    travel(start+1);
    output();

    return 0;
}
```