

Dijkstra算法代码实现



主讲人：邓哲也



P0J2387 Til the Cows Come Home

- 给出一个 n 个点， m 条无向边（带非负边权）的图。
- 求出从 1 走到 n 的最短路。
- $n \leq 1000$, $m \leq 2000$, 有重边。

- 样例输入：

- 5 5
- 1 2 20
- 2 3 30
- 3 4 20
- 4 5 20
- 1 5 100

- 样例输出：

- 90

Dijkstra算法代码实现

- 使用邻接矩阵来存图。
- `edge[i][j]` 表示 `i` 到 `j` 之间的边权。
- 初始化 `edge` 数组:
- `for(int i = 1;i <= n;i ++)`
- `for(int j = 1;j <= n;j ++)`
- `if (i != j) edge[i][j] = 0x3f3f3f3f;`

Dijkstra算法代码实现

- 读入无向图，注意处理重边。
- `int u, v, w;`
- `for(int i = 1; i <= m; i++) {`
- `scanf("%d%d%d", &u, &v, &w);`
- `edge[u][v] = min(edge[u][v], w);`
- `edge[v][u] = min(edge[v][u], w);`
- `}`

Dijkstra算法代码实现

- dijkstra 的初始化部分:

```
void dijkstra(int u) {  
    for(int i = 1; i <= n; i++) {  
        S[i] = 0;  
        dist[i] = edge[u][i];  
    }  
    S[u] = 1;
```

Dijkstra算法代码实现

- dijkstra 的循环部分:

```
for(int i = 1;i < n;i ++){
    int min = 0x3f3f3f3f, v = u;
    for(int j = 1;j <= n;j ++){
        if(!S[j] && dist[j] < min)
            min = dist[j], v = j;
    }
    S[v] = 1;
    for(int j = 1;j <= n;j ++){
        if(!S[j] && dist[v] + edge[v][j] < dist[j])
            dist[j] = dist[v] + edge[v][j];
    }
}
```

Dijkstra算法代码实现

- 调用 `dijkstra(1)`, 输出 `dist[n]` 即为答案。
- 快去动手试一试吧!

下节课再见