**单源最短路径算法说明书**

**1. 算法功能**

给定带权有向图G={V,E}，其中每条边的权是非负实数。另外，给定Ｖ中一个点，称为源。计算从源到所有其他顶点的最短路长度。

**2. 接口参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 是否必须 | 示例值 | 描述 |
| 1 | graph | double [][] | 是 | [  [0, 10, inf],  [inf, 0, 50],  [inf, inf, 0]  ] | 二维数组，以邻接矩阵表示的有向带权图 |
| 2 | v | int | 否 | 0 | 源点的索引，默认为0，即第一个点 |

1. **接口返回值**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 是否必须 | 示例值 | 描述 |
| 1 | dist | double[] | 是 | [0, 10.0, 50.0] | 源点到其余各点的最短距离 |
| 2 | prev | int[] | 是 | [0, 0, 3] | 之前一个点的索引，可根据此数组输出最短路径 |

1. **算法实现**

（1）输入的有向带权图G={V,E}，V={1,2,...,n}。顶点v是源点。a是一个二维数组，以邻接矩阵的形式表示图G。dist[i]表示当前从源到顶点i的最短特殊路径长度。

（2）在所有不在S中的顶点中选择其dist最小的顶点x。

（3）将x加入S，对于所有从x出发的边(x,y)更新dist[y]=min{d[y], d[y]+a[x][y]}。

**5. 注意事项**

若要输出最短路径可递归地遍历数组prev