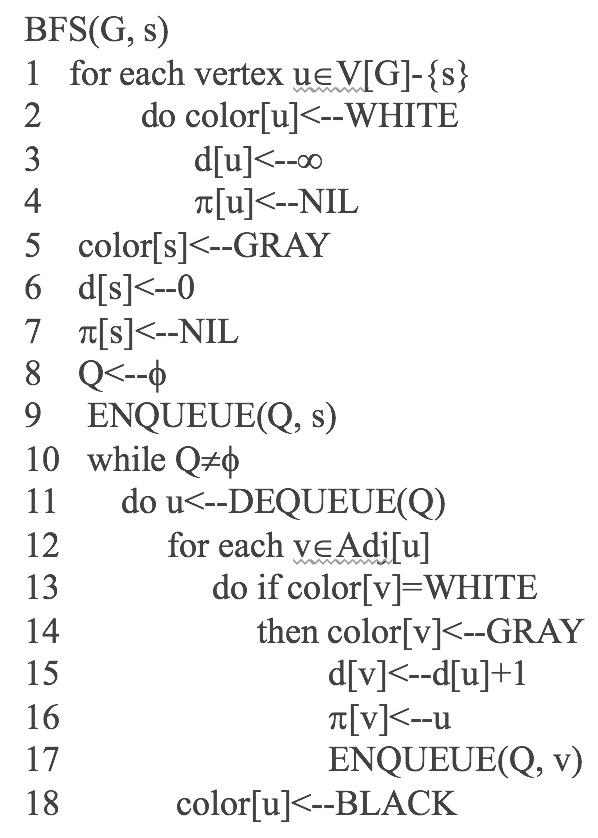
Chpt.22 基本的图算法

22.1 两种表示方式

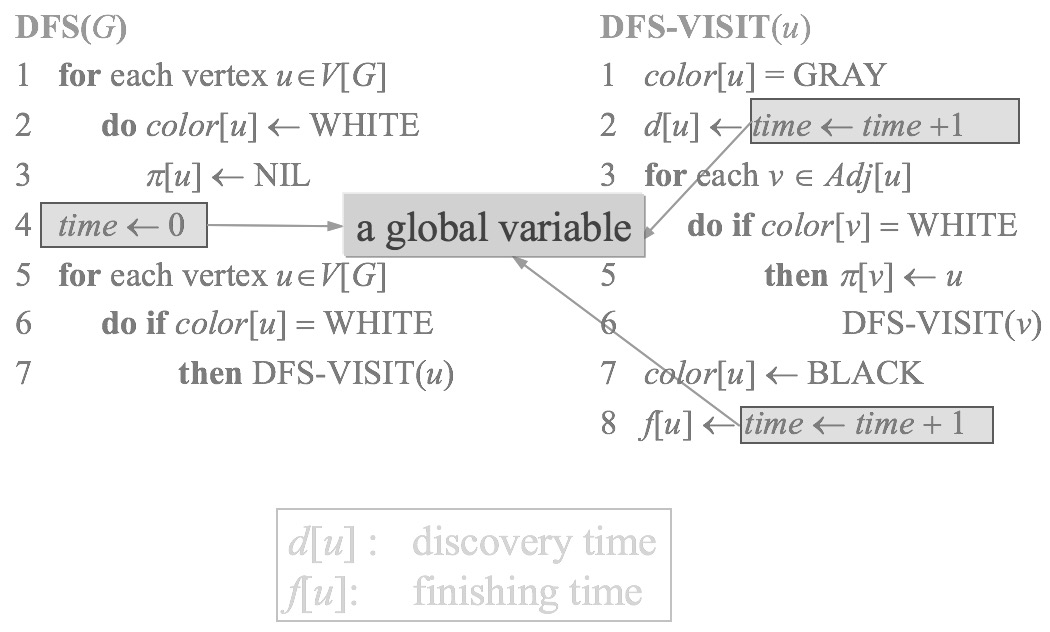
22.2 广度优先搜索

算法：先进先出树 证明：反证法 时间复杂度：O(V+E)



22.3 深度优先搜索

算法：每个节点两个值 时间复杂度：Θ(V+E)



性质：

括号定理：完全包含为后代 相互分离不是后代

后代区间的嵌套

白色路径定理：在有向或无向图G=（V,E）的深度优先森林中，结点v是结点u的后代当且仅当在发现结点u的时间u.d存在一条从结点u到结点v的全部由白色节点所构成的路径。

边的分类：

树边：第一次探索（u，v）时，v为白色说明是一条树边

后向边：u连接到一个祖先v的边，v为灰色说明是一条后向边

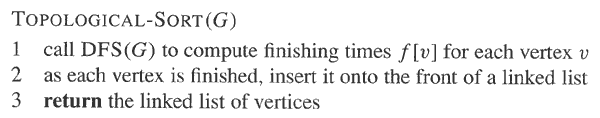
前向边：u连接到一个后代的边 v为黑色

横边：其他所有的边 同树非后代或者非同树 v为黑色

22.4 拓扑排序

算法：深度优先，后被涂黑的放在前面

时间复杂度：O(V+E)



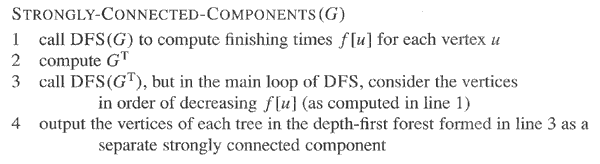
22.5 强连通分量

算法：转置前后各DFS

时间复杂度：Θ(*V* + *E*)

正确性证明：

归纳假设算法第三行所生成的前面k棵树都是强连通分量，现在需要考虑第（k+1）棵树。对于没访问过的强连通分量C’来说u.f=f(C)>f(C’)，C中所有节点都是白色的，所以都是u的后代。二在GT中所有从C发出的边只能指向已经访问过的强连通分量。



Chpt.23 最小生成树

23.1 基本算法

通用算法：

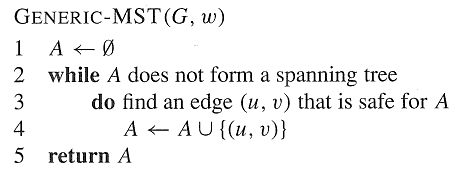
每变循环之前A是某棵最小生成树的子集。每一步添加一条安全边。

最小生成树的三个性质：

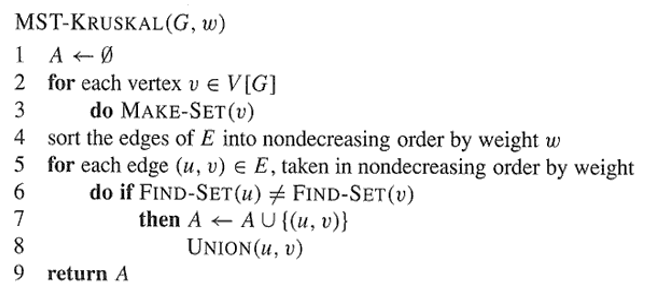
图中权重最小的边一定在最小生成树中

如果每个权重都不同则最小生成树是唯一的

将普通的生成树的边从小到大排列,每个边都大于等于对应最小生成树的边。

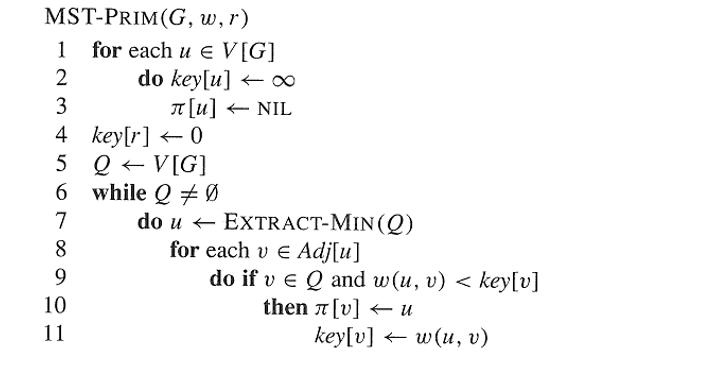


23.2 Kruskal算法 贪心算法



复杂度：O（E lgV）

23.2 Prim算法 贪心算法



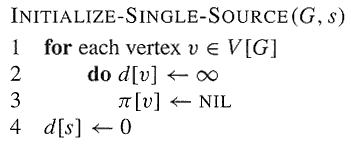
复杂度：O（E lgV）

Chpt.24 单源最短路径

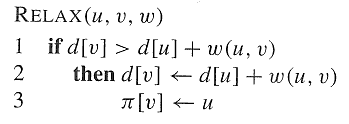
24.0

最短路径变体：+单目的地 单节点对 所有节点对最短路径问题

初始化算法：



松弛算法：



性质：

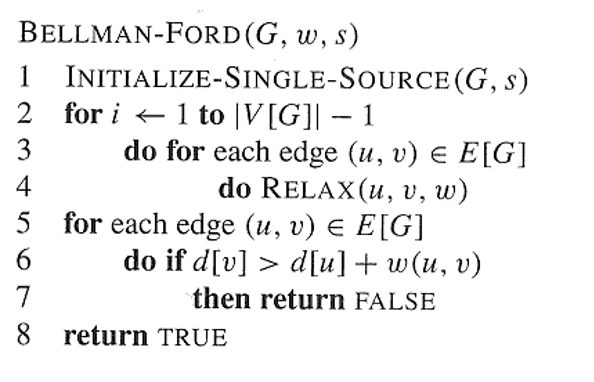
三角不等式 对于任何边 δ(s, v) ≤ δ(s, u) + w(u, v)

上界性质 d[v] ≥ δ(s, v)

非路径性质 s到v之间不存在路径 d[v] = δ(s, v) = ∞.

收敛性质 路径松弛定理 前驱子图性质

24.1 Bellman-Ford算法



正确性：

可以通过路径松弛定理来证明,v0到vk的每一条边都被松弛一遍，k<=|V|-1。

如果没有负圈，根据三角不等式

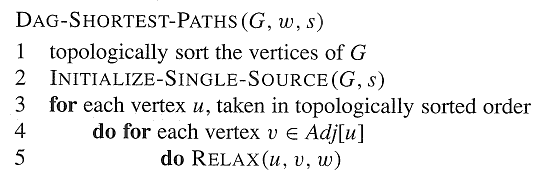


所以不可能返回false

复杂度：O(VE)

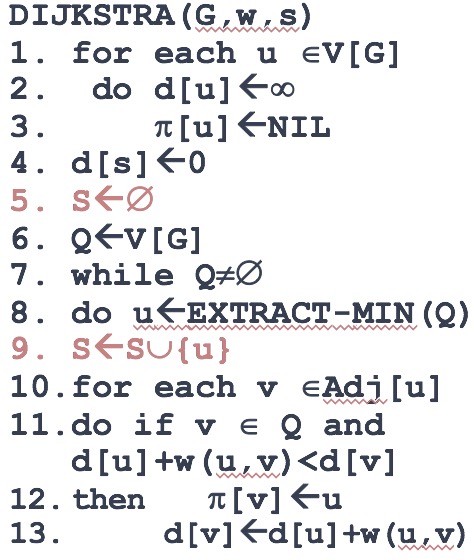
24.2 有向无环图中的单源最短路径问题

算法：先拓扑再依次松弛



复杂度：O(V+E)

24.3 Dijkstra算法



正确性：

U每次被加入集合S时有u.d=最短路径。设u是第一个在加入到集合S时使得不变式不成立的结点。在将u加入到S中时，有S不为空，一定存在某条从结点s到结点u的路径，否则u.d=无穷大。也就只是存在一条从s到u的最短路径p。

复杂度：If using arrays：O(V^2)

Using special data structure O(VlgV+E) or less

24.4 差分约束

线性规划

小于等于，减数指向被减数

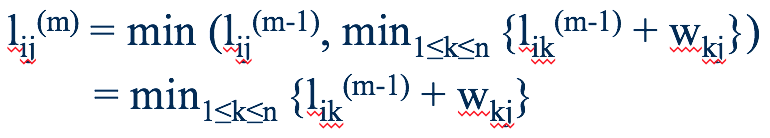
增加节点v0，找最短路径权重

Chpt.25 所有节点对最短路径问题

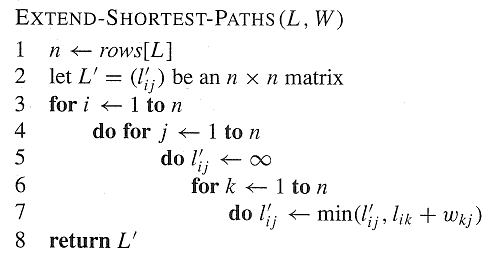
25.1 最短路径和矩阵乘法

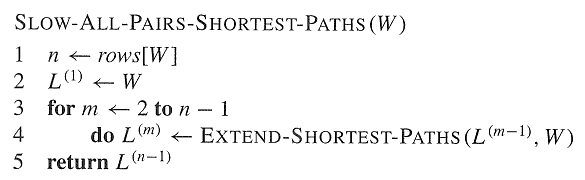
最简单的算法：每个点使用Djikstra,复杂度*:* O(V2 logV+VE)

递归公式



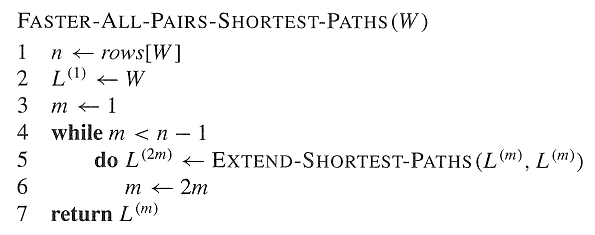
算法





复杂度 Θ(V^4)

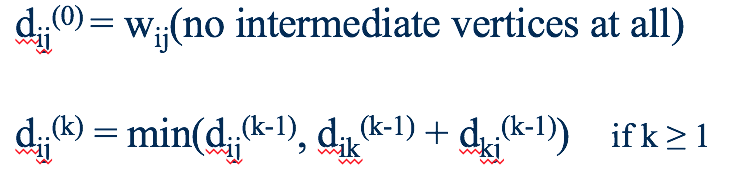
改进算法

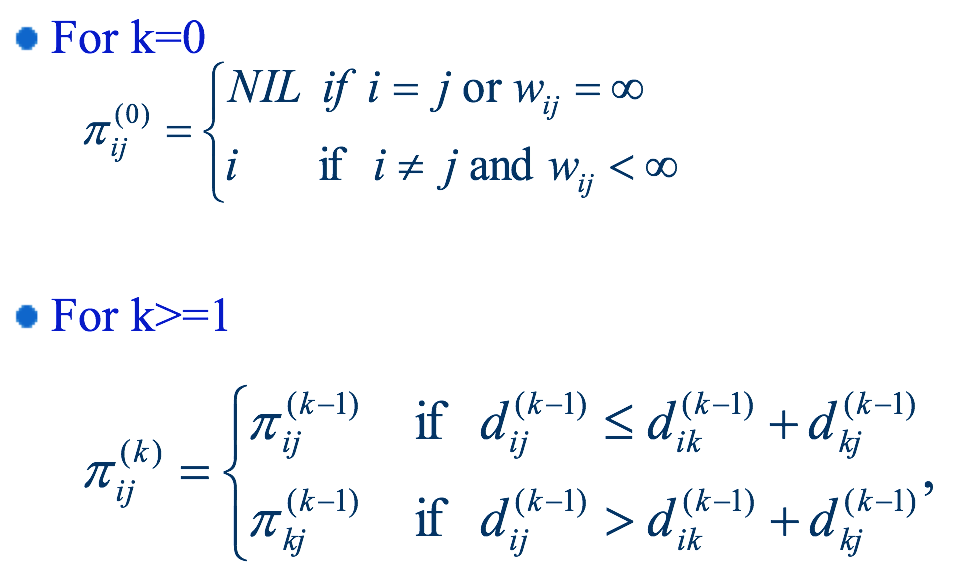


复杂度 Θ(V^3lgn)

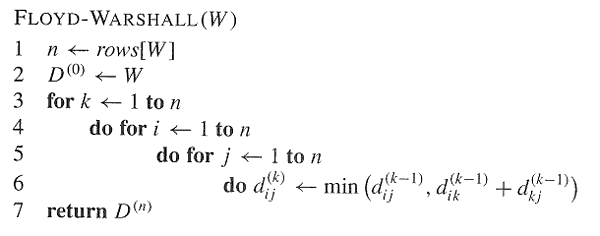
25.2 Floyd-Warshall算法

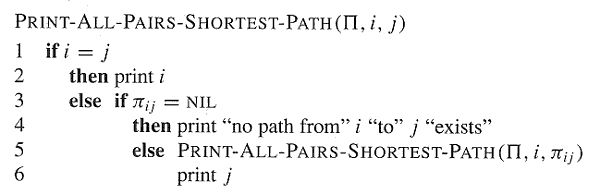
递归公式





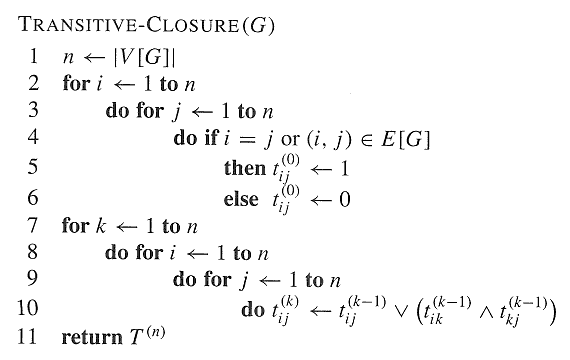
算法





时间复杂度Θ(V^3)

有向图的传递闭包



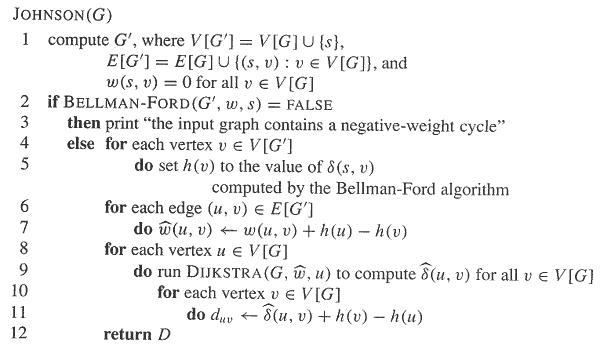
25.3 用于稀疏图的Johnson算法

对于负值重新赋权



再用Dijkstra

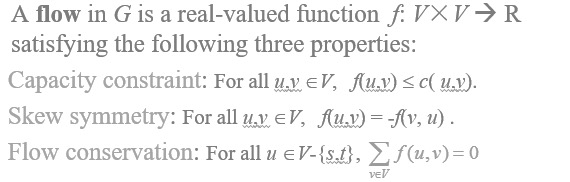
算法 复杂度O(V^2lgV+VE) 二叉最小堆O(VElgV)



Chpt.26 最大流

26.1 流网络

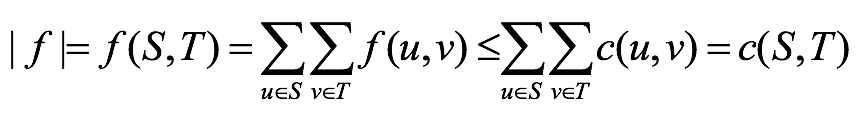
容量限制 流量守恒



26.2 Ford-Fulkerson方法

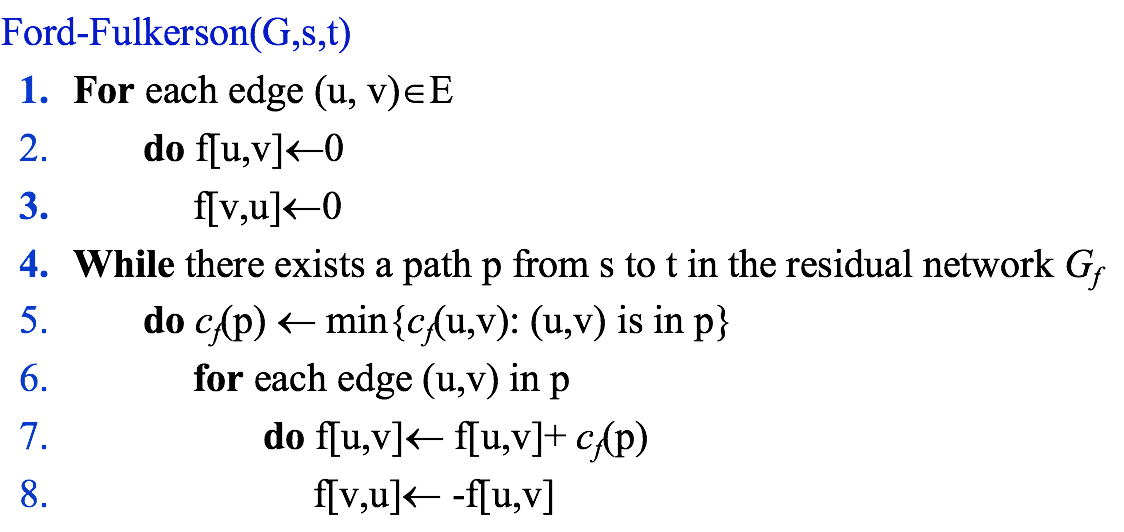
给出残存网络，依次添加增广路径

切割：



f-左到右减右到左

算法



复杂度 O(E|f\*|)

改良 – Edmonds-Karp

每次找增广路径都找最短路径

复杂度 O(VE^2)