

有向图的连通性 与拓扑排序



主讲人：邓哲也



有向图的连通性

由于有向图的边具有方向性，所以有向图的连通性比较复杂。

有向图的连通性分为三种：

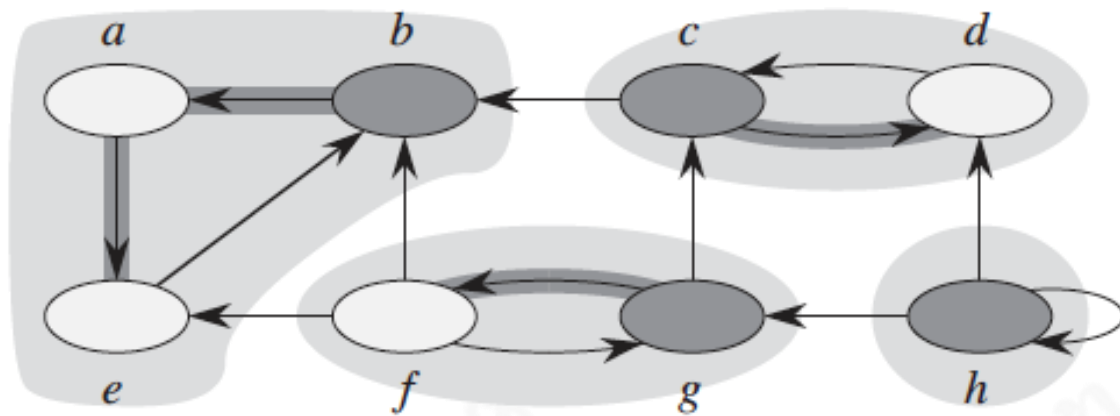
强连通

单连通

弱连通

强连通

若 G 是有向图，如果对图 G 的任意两个顶点 u 和 v ，既存在 u 到 v 的路径，也存在 v 到 u 的路径，则称该有向图为强连通有向图。对于非强连通图，其极大强连通子图称为其强连通分量。



单连通和弱连通

若 G 是有向图，如果对图 G 的任意两个顶点 u 和 v ，存在 u 到 v 的路径或 v 到 u 的路径，则称该有向图为单连通有向图。

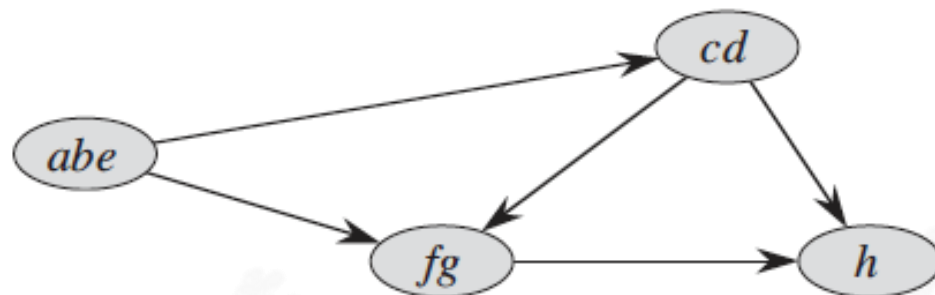
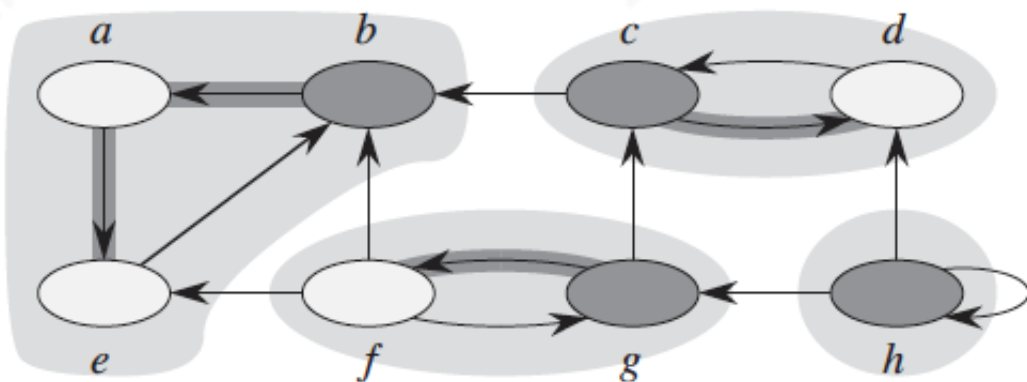
若 G 是有向图，如果忽略图 G 中每条有向边的方向，得到的无向图（即有向图的基图）连通，则称该有向图为弱连通有向图。

强连通图一定是单连通和弱连通图。

单连通图一定也是弱连通图。

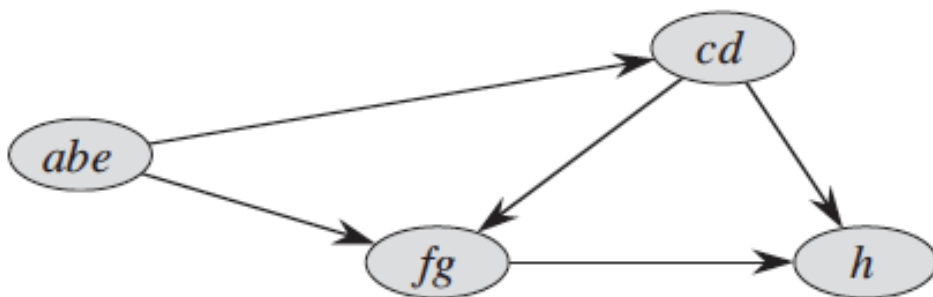
有向无环图 (DAG)

如果把每个强连通分量看成一个点，那么这个图就变成了一个有向无环图 (DAG)。



判断有向无环图的方法

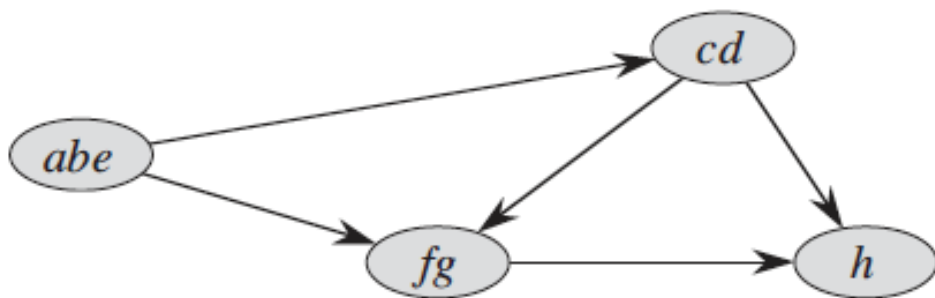
判断有向无环图的方法：可以对图进行一次深度优先搜索，如果没有返向边，那么就是一个有向无环图。



判断有向无环图的方法

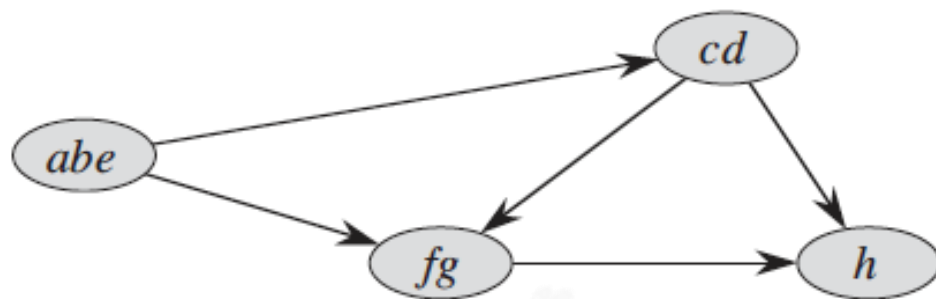
判断有向无环图的方法：可以对图构造一个拓扑序列，使得图中所有的前驱和后继关系能够得到满足。

如下图的一个拓扑序列为：abe, cd, fg, h



拓扑排序实现方法

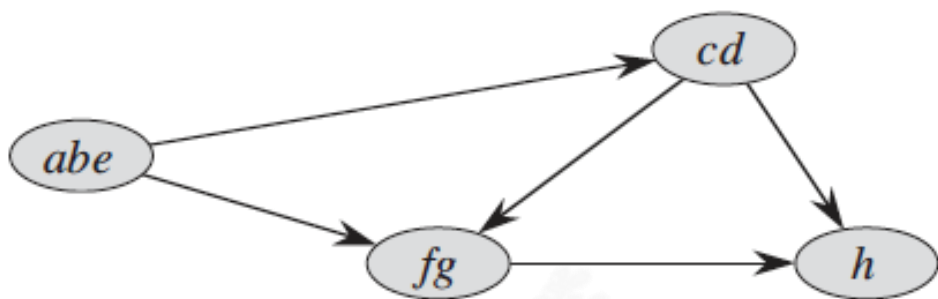
1. 在图中选出一个入度为 0 的点并输出
2. 从图中删去该顶点及该顶点发出的所有边
3. 重复 1 和 2，直到找不到入度为 0 的顶点。



拓扑排序实现方法

按照上述方法进行拓扑排序，结果有两种：

1. 所有顶点都被输出，说明这是个有向无环图
2. 还有顶点没有输出，说明这个图有环



拓扑排序实现方法

拓扑排序在实现时需要建立一个 `count` 数组，记录各个顶点的入度。

入度为 0 的点就是无前驱的点。

还需要建立一个栈，用于存放入度为 0 的点。只要出现入度为 0 的点就将它压入栈中。

拓扑排序伪代码

1. 建立入度为 0 的顶点栈，初始时将所有入度为 0 的点入栈
2. 栈非空时，弹出栈顶顶点并输出，删去从该顶点发出的边，边的终点入度减 1，若减完后为 0 则入栈。
3. 如果输出的点小于顶点总数，则说明有环。

拓扑排序时间复杂度

每个点都入栈一次，每条边都被访问了一次，
故时间复杂度为 $O(n + m)$ 。

下节课再见