



数据结构与算法 (Python版)

图的应用：强连通分支

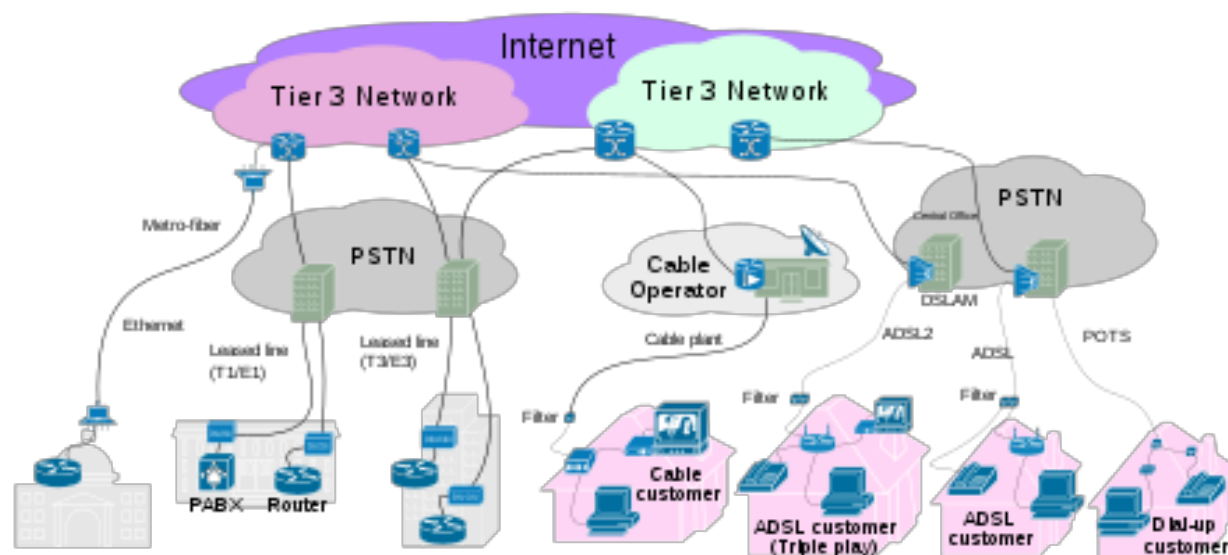
陈斌 北京大学 gischen@pku.edu.cn

强连通分支

❖ 我们关注一下互联网相关的**非常巨大图**：

由主机通过网线（或无线）连接而形成的图；

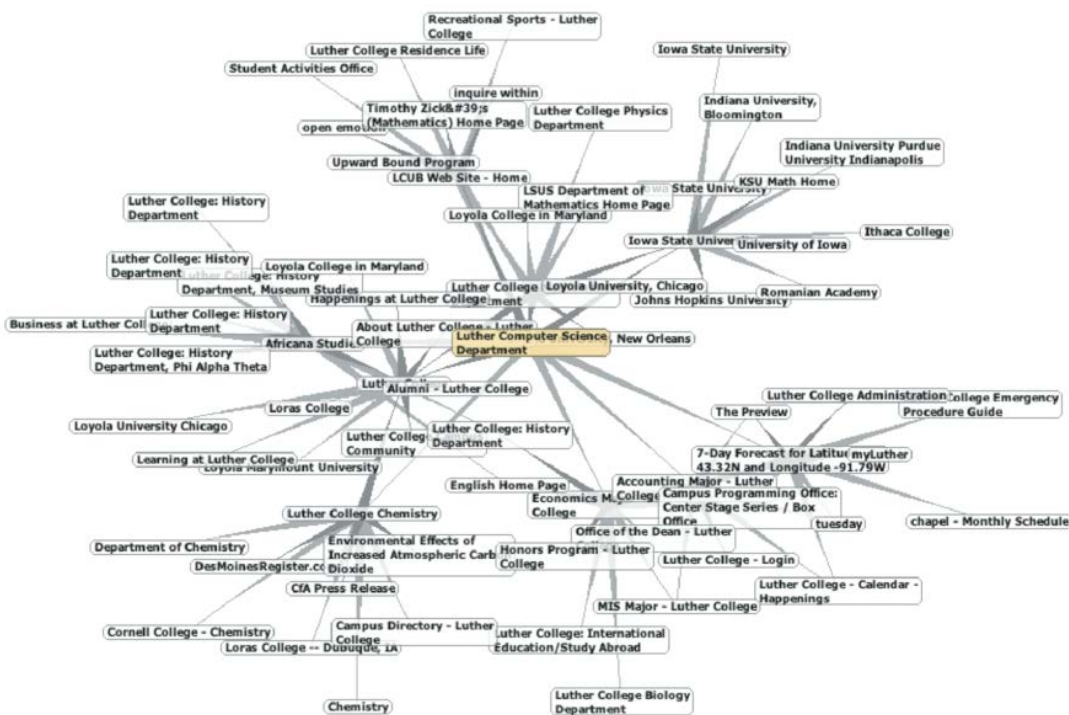
以及由网页通过超链接连接而形成的图。



强连通分支

❖ 先看网页形成的图

以网页（URI作为id）为顶点，网页内包含的超链接作为边，可以转换为一个有向图。



图中包含了许多路德学院其它系的网站
包含了一些爱荷华其它大学学院的网站
还包含了一些人文学院的网站

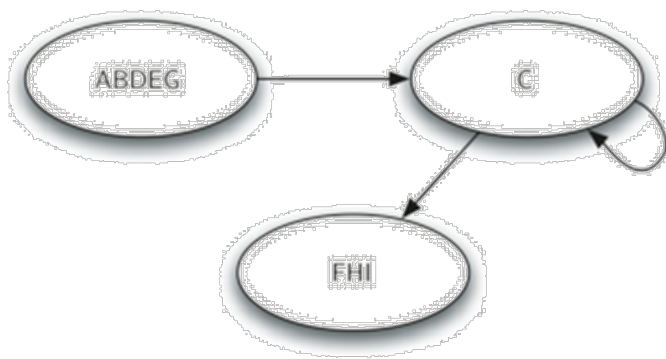
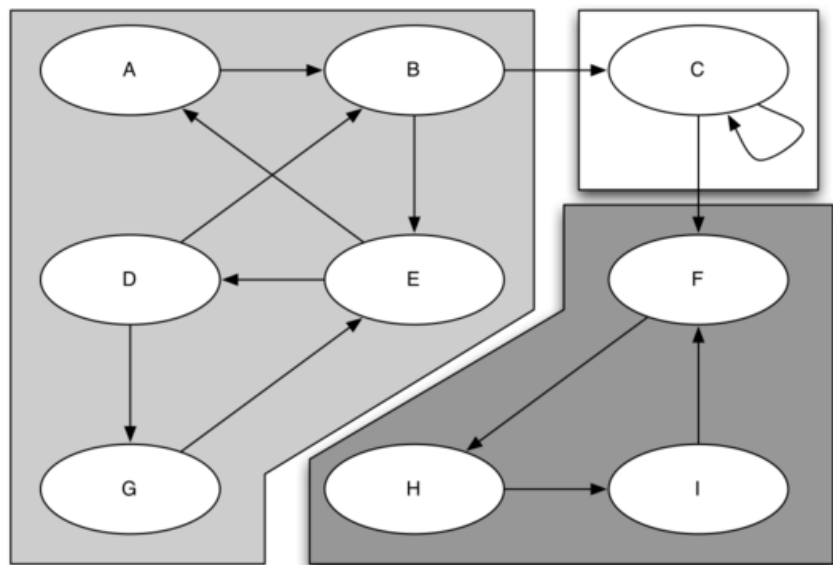


强连通分支

- ❖ 我们可以猜想，Web的底层结构可能存在某些同类网站的聚集
- ❖ 在图中发现高度聚集节点群的算法，即寻找“强连通分支 Strongly Connected Components” 算法
- ❖ 强连通分支，定义为图G的一个子集C
C中的任意两个顶点 v, w 之间都有路径来回，即 $(v, w)(w, v)$ 都是C的路径，
而且C是具有这样性质的最大子集

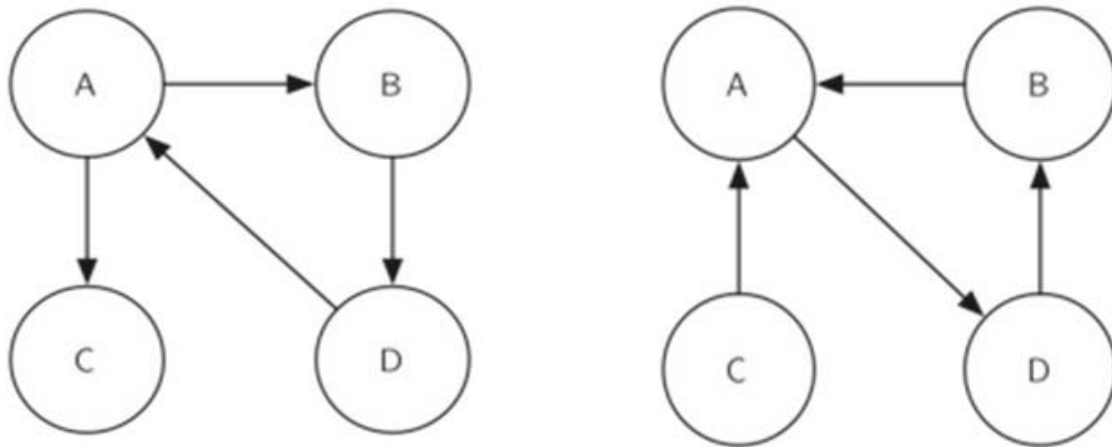
强连通分支例子

- ❖ 下图是具有3个强连通分支的9顶点有向图
- ❖ 一旦找到强连通分支，可以据此对图的顶点进行分类，并对图进行化简。



强连通分支算法：转置概念

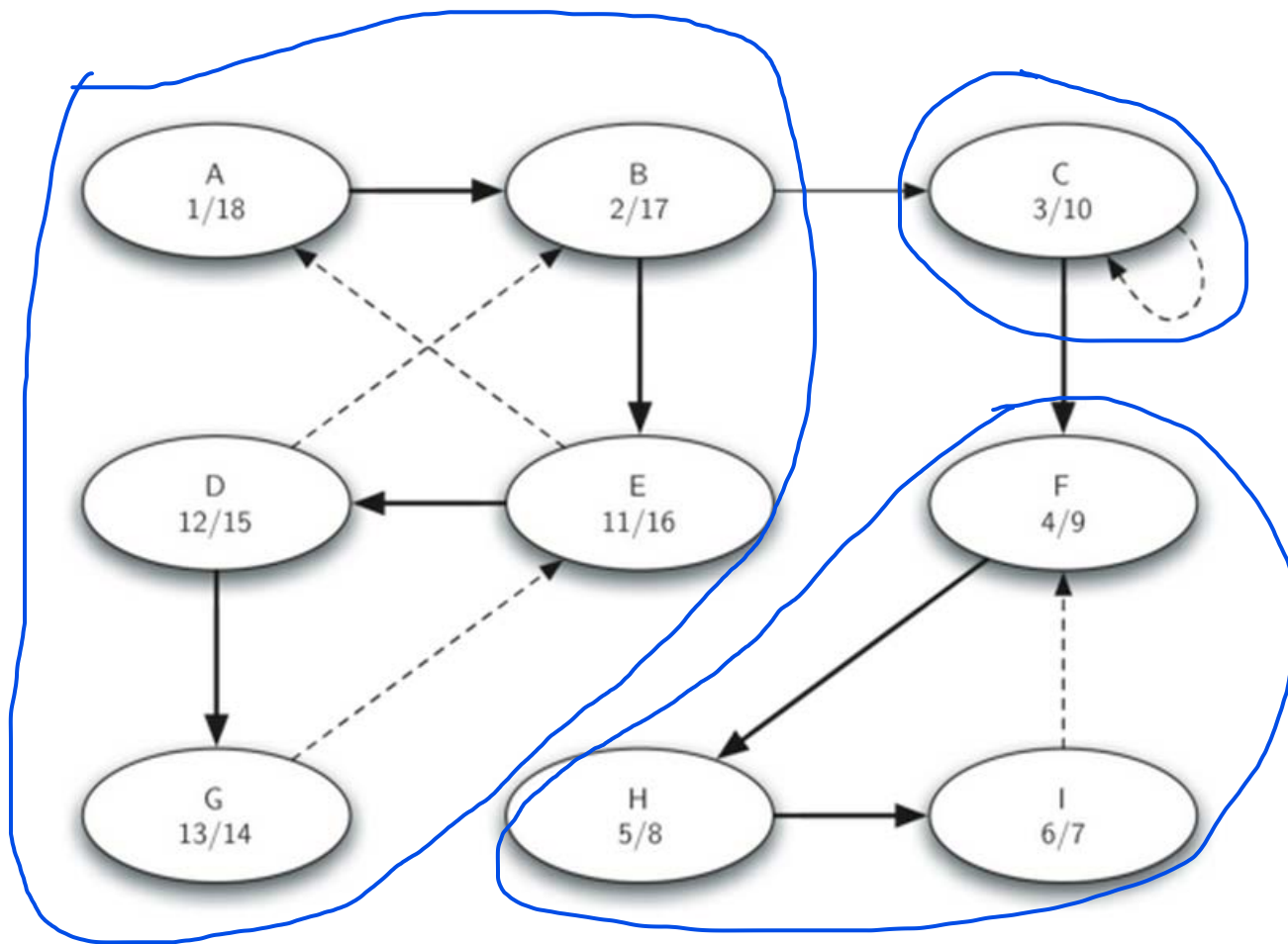
- ❖ 在用深度优先搜索来发现强连通分支之前，先熟悉一个概念：Transposition转置
一个有向图 G 的转置 G^T ，定义为将图 G 的所有边的顶点交换次序，如将 (v, w) 转换为 (w, v)
可以观察到图和转置图在强连通分支的数量和划分上，是相同的。



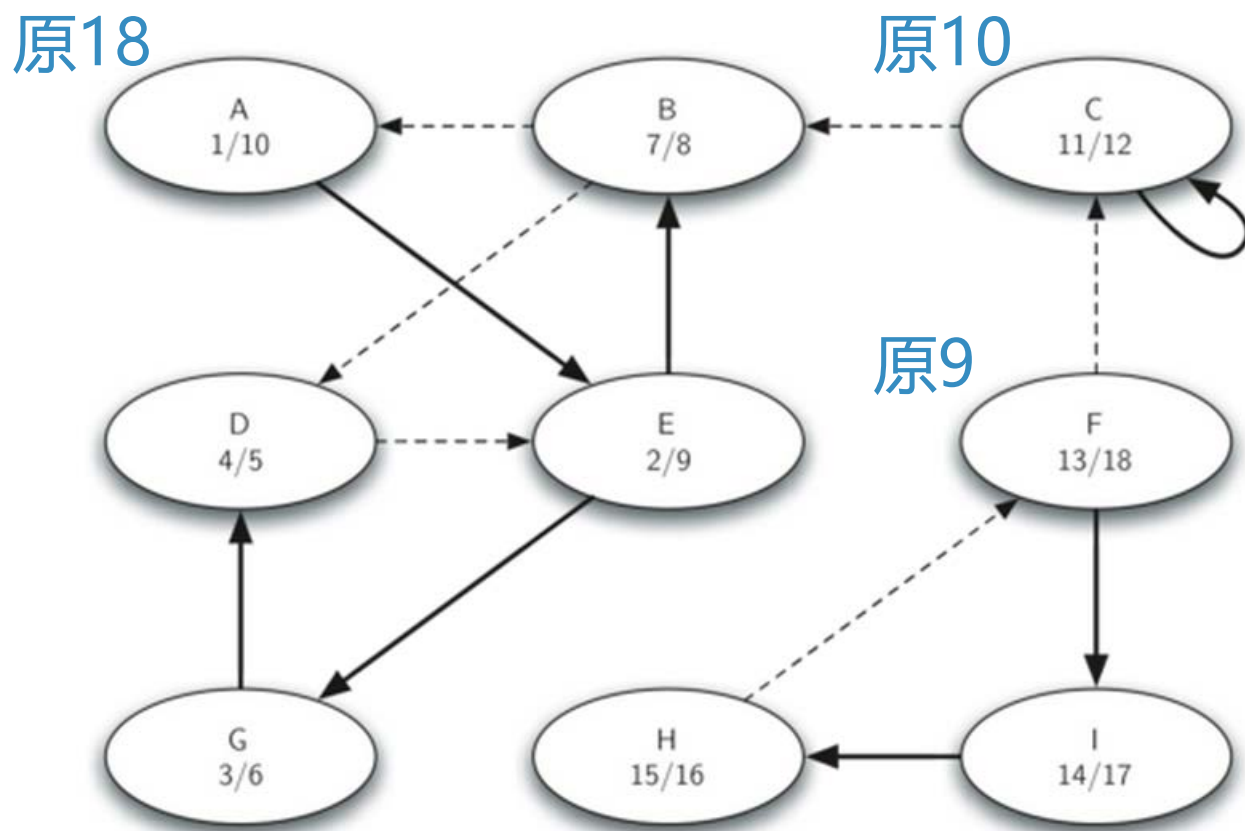
强连通分支算法: Kosaraju算法思路

- ❖ 首先, 对图G调用DFS算法, 为每个顶点计算“结束时间”;
- ❖ 然后, 将图G进行转置, 得到 G^T ;
- ❖ 再对 G^T 调用DFS算法, 但在dfs函数中, 对每个顶点的搜索循环里, 要以顶点的“结束时间”倒序的顺序来搜索
- ❖ 最后, 深度优先森林中的每一棵树就是一个强连通分支

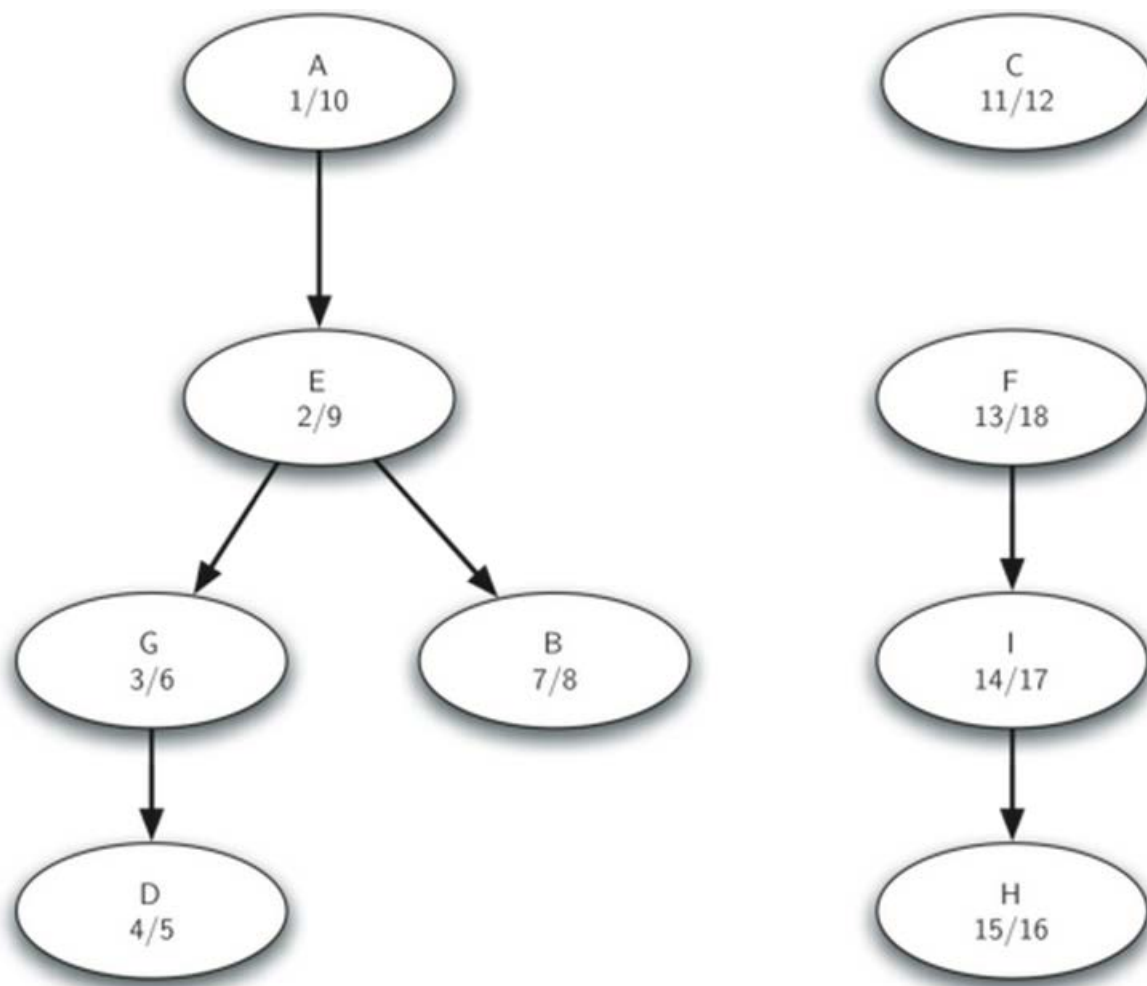
Kosaraju算法实例：第一趟DFS



Kosaraju算法实例：转置后第二趟DFS



Kosaraju算法实例：结果



强连通分支算法

❖ 另外的常用强连通分支算法

Tarjan算法

Gabow算法, 对Tarjan的改进

❖ 参考阅读

<https://baike.baidu.com/item/tarjan算法>

