

6. 图

深度优先搜索 性质

身后有余忘缩手
眼前无路想回头

邓俊辉

deng@tsinghua.edu.cn

DFS树/森林

❖ 从顶点s出发的DFS

- 在**无向图**中将访问与s**连通**的所有顶点 **connected component**
- 在**有向图**中将访问由s**可达**的所有顶点 **reachable component**

❖ 经DFS确定的树边，不会构成回路

❖ 从s出发的DFS，将以s为根生成一棵DFS树；所有DFS树，进而构成DFS森林

❖ DFS树及森林由parent指针描述（只不过所有边取反向）

❖ DFS之后，我们已经知道**森林**乃至**原图**的**全部**信息了吗？

就某种意义而言，是的...

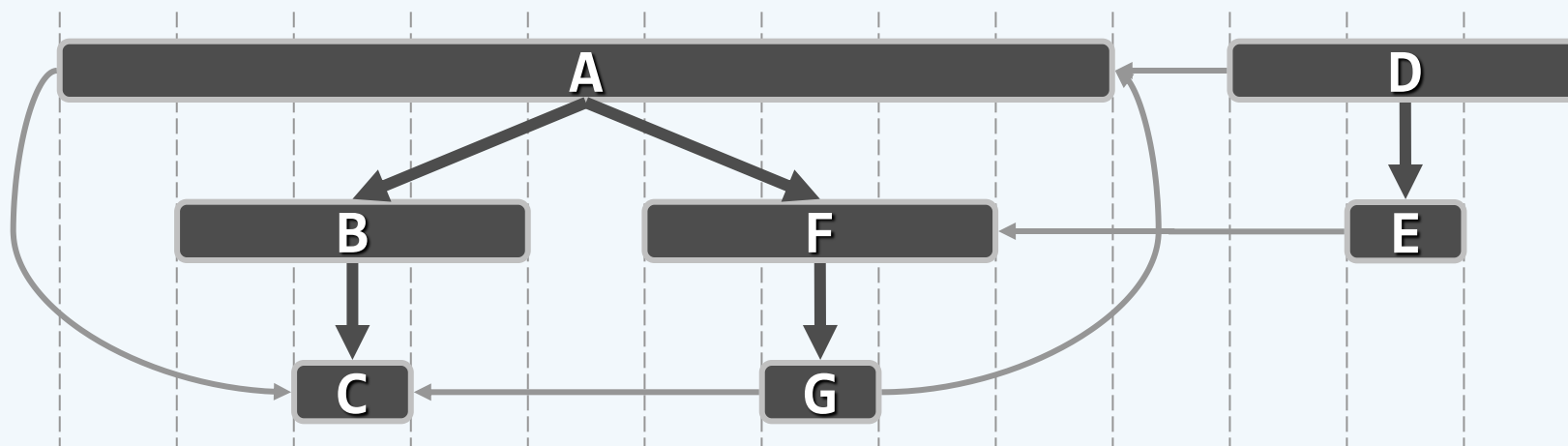
括号引理

❖ **活跃期** : $\text{active}[u] = (\text{dTime}[u], \text{fTime}[u])$

❖ Parenthesis Lemma : 给定有向图 $G = (V, E)$ 及其任一DFS森林, 则

- u 是 v 的后代 iff $\text{active}[u] \subseteq \text{active}[v]$
- u 是 v 的祖先 iff $\text{active}[u] \supseteq \text{active}[v]$
- u 与 v “无关” iff $\text{active}[u] \cap \text{active}[v] = \emptyset$

❖ 仅凭 $\text{status}[]$ 、 $\text{dTime}[]$ 和 $\text{fTime}[]$, 即可对各边分类...



边分类

❖ TREE(v, u):   可从当前 v 进入处于UNDISCOVERED状态的 u

❖ BACKWARD(v, u):   试图从当前 v 进入处于DISCOVERED状态的 u

DFS发现后向边 iff 存在回路

//后向边数 == 回路数?

❖ FORWARD(v, u):  

试图从当前顶点 v 进入处于VISITED状态的 u ，且 v 更早被发现

❖ CROSS(v, u):  

试图从当前顶点 v 进入处于VISITED状态的 u ，且 u 更早被发现

❖ 无向图中，后向边与前向边不予区分，跨边没有

//为什么?

遍历算法的应用

连通图的支撑树 (DFS/BFS Tree)	DFS/BFS
非连通图的支撑森林	DFS/BFS
连通性检测	DFS/BFS
无向图环路检测	DFS/BFS
有向图环路检测	DFS
顶点之间可达性检测/路径求解	DFS/BFS
顶点之间的最短距离	BFS
直径	BFS
Eulerian tour	DFS
拓扑排序	DFS
双连通分量、强连通分量分解	DFS
...	...