

问求算法设计与实践02

图结构

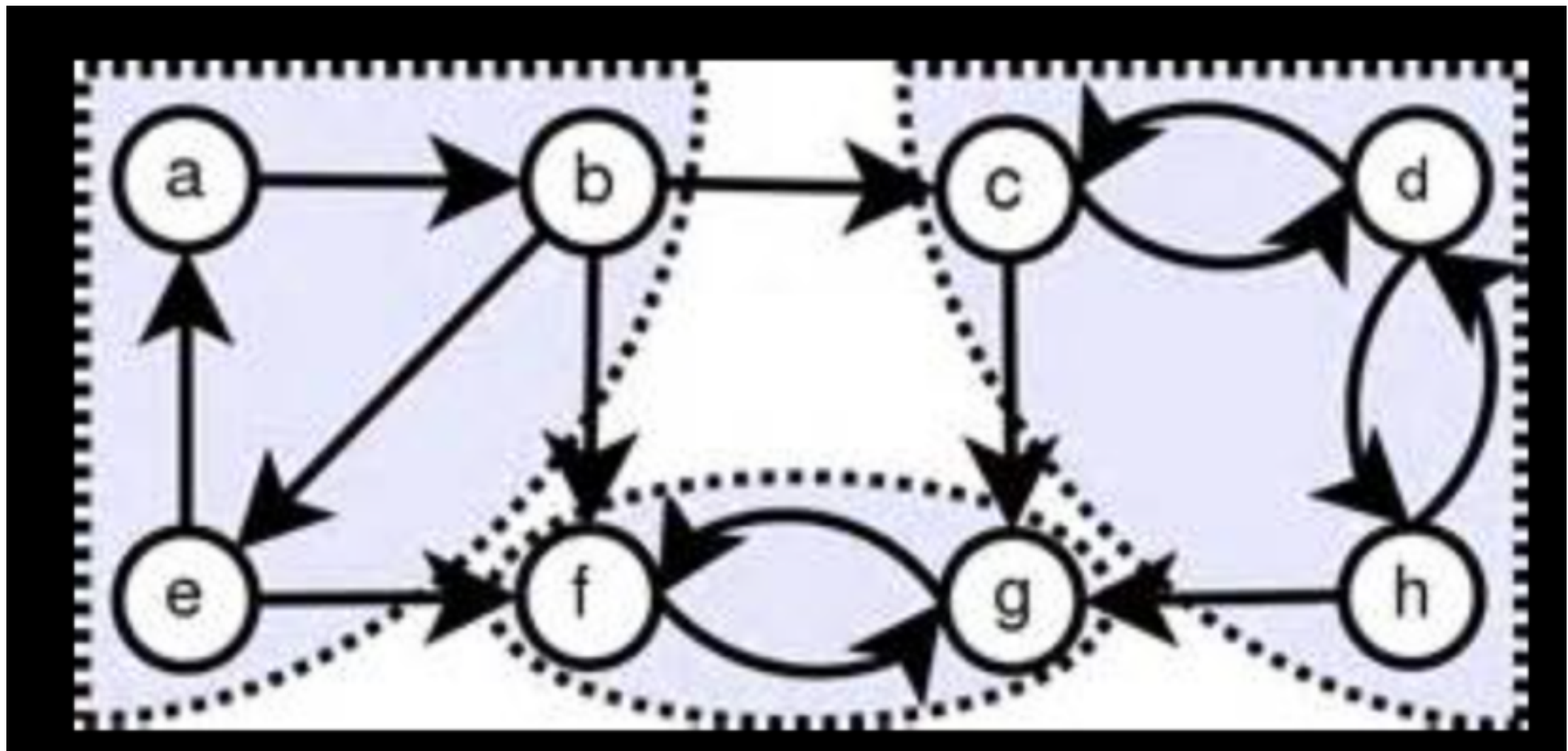
拓扑排序

- 用队列维护当前入度为0的点
- 不断把与队首相连的边删除，将另一端点的入度减1
- 如果有新的点入度变为0，则加入队尾
- 拓扑序不唯一
- DAG

强连通分量(SCC)

- 块(Block), 割点(cut vertex), 桥/割边(cut edge)
- 连通度, 弱连通分量
- Kosaraju algorithm
- Tarjan algorithm
- <http://www.cs.cmu.edu/afs/cs/academic/class/15451-s14/LectureNotes/lecture12.pdf>

强连通分量(SCC)



Tarjan's Algorithm

$LOW(u) = \text{Min}$

{

DFS(u),

$LOW(v)$, (u,v)为树枝边, u为v的父节点

DFS(v), (u,v)为指向栈中节点的后向边(非横叉边)

}

维护一个栈, 记录到过的节点编号

当 $DFS(u) == LOW(u)$ 即找到一个SCC, 依次出栈直到u, 出栈的点即为一个SCC

```
1  stack<int> S;
2  bool B[MN];
3  int D[MN], L[MN], C[MN], SZ[MN];
4  int nowc, nowd;
5  void Dfs( int k)
6  {
7      D[k]=L[k]=++nowd;
8      B[k]=1; S.push(k);
9      for (int p=R[k]; p; p=E[p].ne)
10         if (B[E[p].v])
11             L[k]=min(L[k], D[E[p].v]);
12         else if (!D[E[p].v])
13             Dfs(E[p].v), L[k]=min(L[k], L[E[p].v]);
14         if (D[k]==L[k])
15         {
16             int v;
17             nowc++;
18             do
19             {
20                 v=S.top();
21                 B[v]=0;
22                 SZ[nowc]++;
23                 S.pop();
24                 C[v]=nowc;
25             }while(v!=k);
26         }
27 }
```

problem 2

Problem 2. 有 N 个存钱罐，打开第 i 个罐子的钥匙在第 X_i 个罐子里. 你可以将罐子打碎拿到钥匙. 求一种取出所有钱的方法并使得打碎的罐子最少.

$$N \leq 10^6$$

problem 2 cont'd

- A的钥匙在B中，连接B->A
- 这是一个每个点入度为1的有向图
- 这个离散结构有以下几种形式
 - 环（不会再接环）
 - 环外接多叉树
- 以上形式均打开一个罐子即可
- 用BFS或者并查集找出连通分量个数, $O(N)$

problem 3

Problem 3. 给定 N 个点, M 条带权边, 你要选出 $N - 1$ 条边, 然后从一个点开始, 访问每个点至少 1 次, 然后回到起点. 你每到一个点 i 将支付 D_i 的费用, 你每经过一条边 j , 将支付 F_j 的费用. 要求找出一种费用最小的方案.

$$N \leq 10^5, M \leq 10^6$$

problem 3 cont'd

- N 个点选出 $N-1$ 条边并连通—>树
- 假设选好了一棵树，遍历的路径？
- 每条边经过两次
- 每个点经历 度数 次

problem 3 cont'd

- 最小生成树!
- 每条边的边权为 $2 \times \text{边权} + \text{点权1} + \text{点权2}$
- Kruskal or Prim algorithm
- $O(M \log M)$ or $O(M \log N)$

problem 4

Problem 4. 一个 N 个点 M 条边的无向图，将 N 个点分成尽量多的组，满足任意两个不在同一组的点之间都有边。

$N \leq 100000, M \leq 2000000$

problem 4 cont'd

- 结论：答案等于补图中的连通块的个数。
- 证明：
 - 可行性：按补图的连通块分组是一种方案。
 - 最优性：若分出了更多的组，补图的一个连通块中会有至少两个组的点，从而之间有边，矛盾。

problem 4 cont'd

- 补图中有 $O(N^2)$ 条边
- BFS的优化
- 将未访问点挂链，扩展一个点时，标记原图中该点的相邻点，遍历链表，将所有未标记点（即该点在补图中的邻居）入队并从链表中删除。
- 时间复杂度 $O(N+M)$

SAT(Boolean Satisfiability Problem)

- 给定关于N个布尔变量的若干约束,由and,or,not组成运算符
- 析取范式 合取范式
- 3-SAT, NP, NPC, SAT solver(DPLL)
- 2-SAT (Tarjan 1979)
$$(x_0 \vee x_2) \wedge (x_0 \vee \neg x_3) \wedge (x_1 \vee \neg x_3) \wedge (x_1 \vee \neg x_4) \wedge (x_2 \vee \neg x_4) \wedge (x_0 \vee \neg x_5) \wedge (x_1 \vee \neg x_5) \wedge (x_2 \vee \neg x_5) \wedge (x_3 \vee x_6) \wedge (x_4 \vee x_6) \wedge (x_5 \vee x_6).$$
- Application: <https://en.wikipedia.org/wiki/2-satisfiability#Applications>

problem 6

Problem 6. 有一个大小为 N 的集合 $\{x_1, x_2 \dots x_n\}$, $x_i = \{0, 1\}$, 现在给出它们之间的 M 个逻辑运算的结果 (比如 $x_1 \text{ and } x_2 = 1$), 逻辑运算有 AND, OR, XOR 三种, 问是否存在一种满足所有条件的取值方案。

$$N, M \leq 10^6$$

problem 6 cont'd

- 构建一个 $2N$ 个点有向图：包含 X_{i0} 和 X_{i1}
- 语义是 $X_i=0$ 和 $X_i=1$ (false and true)
- 如果选择了 X_{i0} 就要选择 Y_{j1} 的话，连接(X_{i0}, Y_{j1})
- 跑一遍强连通分量算法
- 如果任意 X_{i0} 和 X_{i1} 不再同一连通分量中，即有解