

Tutorial 5 图遍历相关

2022/4/19

无向图的DFS: TE、BE 有BE就有环

有向图的BFS: TE、BE、CE

无向图的BFS: TE、CE

Q1:

undirected graph G with n nodes, determine whether G is a forest in time $O(n+m)$

一个连通分量只要没有环就是树，判断是否有环，只要有BE就有环。

判断BE:

```
w.color == GRAY && v.parent!=w
```

Q2:

带权[price]顶点, $\text{cost}[u]$ 表示从 u 出发能到达的点中price最小的一个

The Cost of a Graph

- You are given a directed **acyclic** graph (with n vertexes and m edges) in which each node $u \in V$ has an associated **price** p_u which is a positive integer. Define the array **cost** as follows: for each $u \in V$, **cost** $[u]$ = price of the cheapest node reachable from u (including u itself). For instance, in the graph (with prices shown for each vertex), the **cost** values of the nodes A, B, C, D, E, F are 2, 1, 4, 1, 4, 5, respectively.

Give a $O(m+n)$ algorithm and explain the time complexity.

```
graph LR; A((A)) --> C((C)); B((B)) --> D((D)); C --> E((E)); C --> F((F)); D --> F
```

7

与邻居的cost比较，更新一个小的cost在邻居dfs过后，无论是什么颜色都比。

Cycle Containing e

- Q3:
- Given an undirected graph G and an edge e of G . Design an algorithm to determine if there is a cycle in G containing e in time $O(n+m)$.

是否有一个环包含 $e(uv)$

去掉这个边，看看还能不能从 u 到 v 。

Q4:

Sorting ill-Behaved Children

- Arrange n ill-behaved children in a straight line, facing front. The input is a list of m statements of the form “ i hates j ”. If i hates j , then i cannot be put somewhere behind j , because then i is capable of throwing something at j .
- (a) Give an algorithm that orders the line, or say that it is not possible, in $O(m+n)$ time.
- (b) We want to arrange the children in rows such that if i hates j , then i must be in a lower numbered row than j . Give an algorithm to find the minimum number of rows needed, if it is possible.

捣蛋的小孩：

① i 讨厌 $j \rightarrow i$ 必须站在 j 前面 拓扑排序

②

方法一：关键路径问题，找到最大的 eft 即可

方法二：采用队列找拓扑排序的方式（ds讲的方法真的是太通俗易懂了）

Q5:

A Vertex that reaching all the others

Given a directed graph G with n vertices and m edges.

- Design an algorithm to test whether a given vertex v could reach all other vertices in G .
- Design an $O(n+m)$ algorithm to test whether there is a vertex in G which can reach all other vertices in G .

①遍历判断即可。

②线性时间判断是否存在一个点能到所有顶点

没有限制时间的话比较简单。

I 执行强连通分支算法，找出所有的强连通分支

收缩图是无环的，所以说存在拓扑排序，一定会有一个入度为0的顶点（这个顶点说的是收缩图的顶点，不是 G 上的顶点）。

II

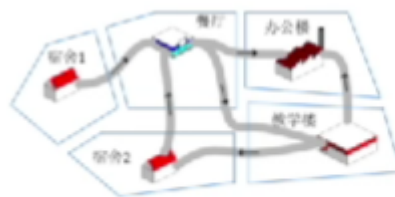
入度为0的点超过一个，说明不存在，只有一个的话，做一次dfs就知道可不可以了

（老师在讲这个习题时很奇怪，他在一开始提供的算法中并没有说需要先执行强连通算法，直接去找了入度为0的点有几个？如果多于1个的话确实是没有一个满足要求的点，但是只有1个或者是没有的时候的判断感觉存在着很大的问题。

Q6

供水问题

- 某校区的供水管网建设规划，管网由若干蓄水池（每个区域有1个）和单向管道构成，箭头表示水流方向。现需要在供水管网规划中选择一个蓄水池作为全校供水源点，以满足全校用水需求。
- 判断从某个特定蓄水池开始供水，是否能将水送到所有蓄水池。
- 设计高效算法解决供水问题。



①遍历

②找到一个可以到达所有地方的点，跟Q5是一样的？是的

Q7

往年考试题

In a directed graph, a vertex is called an “end-point” if it can be reached by every vertex but it can not reach to any other vertex.

(1) “A directed graph has an end-point if and only if it is a DAG”, is it right? If so, prove it; if not, give a counterexample.

(2) Design an algorithm to decide whether a directed graph has an end-point.

(3) Analysis the complexity of your algorithm.

在有向图中，如果存在这样的顶点：所有其他顶点都可以到达它，但它不能到达任何顶点，这样的顶点称为“端点”。

(1) “一个有向图存在端点当且仅当它是一个有向无环图”，这个说法是否正确？若正确，给出证明，否则，举出反例。（5分）

(2) 设计一个算法，判定一个有向图是否存在端点。（10分）

(3) 分析你算法的复杂度。（5分）

①不是的，有环和存在这样的点不冲突

②转置之后调用强连通分量算法，找到入度为0的那一个，然后遍历

难点在于考场上能不能把强连通分量算法给写得比较明白（☹️）

（其实这里做复杂了

端点还有比较明确的定义就是它不到达其它顶点，所以直接看出度为0的点是否唯一且该点到其它点不可达就可以了。

③ $O(n+m)$

往年考试题

在有向图中，如果存在这样的顶点：所有其他顶点都可以到达它，但它不能到达任何顶点，这样的顶点称为“端点”。

(1) “一个有向图存在端点当且仅当它是一个有向无环图”，这个说法是否正确？若正确，给出证明，否则，举出反例。（5分）

(2) 设计一个算法，判定一个有向图是否存在端点。（10分）

(3) 分析你算法的复杂度。（5分）

解答

(1) 不正确，反例很多。

(2) 端点的出度必然为0，而且是唯一的，因为如果有两个出度为0的点，则相互之间不能到达，与端点定义矛盾。假设图的顶点数为 n ，边数为 m 。算法为：

a) 扫描全图，找出出度为0的点 N 。如果这样的点不唯一，则输出“不存在端点”。复杂度为 $O(n)$ 。

b) 构造该图的反向图（即让所有边的方向都反转）。复杂度为 $O(n+m)$ 。

c) 从点 N 开始，用DFS或BFS遍历，如果可以遍历所有节点，则 N 为端点，否则，输出“不存在端点”。复杂度为 $O(n+m)$ 。

(3) 综上，复杂度为 $O(n+m)$ 。

2016备选考题

The police department in the city of Computopia has made all streets one-way. The mayor contends that there is still a way to drive legally from any intersection in the city to any other intersection, but the opposition is not convinced. A computer program is needed to determine whether the mayor is right. However, the city elections are coming up soon, and there is just enough time to run a *linear-time* algorithm.

- (a) Formulate this problem graph-theoretically, and explain why it can indeed be solved in linear time.
- (b) Suppose it now turns out that the mayor's original claim is false. She next claims something weaker: if you start driving from town hall, navigating one-way streets, then no matter where you reach, there is always a way to drive legally back to the town hall. Formulate this weaker property as a graph-theoretic problem, and carefully show how it too can be checked in linear time.

有向图，线性时间算法看是否所有城市都是可达的

①用图论的语言来描述这个图：

顶点表示城市，城市和城市之间有单向道路则它们在有向图中有一条边。

②互相可达说明是一个强连通图，做强连通片算法，看是不是成为了一个强连通片，也就是说从sink SCC开始能不能遍历到所有的顶点。

第二个问题是说town hall是否存在于一个sink SCC中，存在的话就说明是正确的（在一个连通分量里面转，转不到别的连通分量里去，这个也比较好判断，只要找出sink SCC，看town hall是否在连通分量内即可。