

有向无环图的拓扑排序

Dezeming Family

2022 年 2 月 3 日

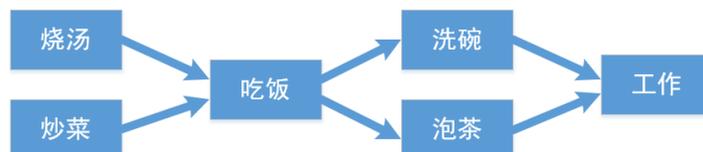
DezemingFamily 系列书和小册子因为是电子书，所以可以很方便地进行修改和重新发布。如果您获得了 DezemingFamily 的系列书，可以从我们的网站 [<https://dezeming.top/>] 找到最新版。对书的内容建议和出现的错误欢迎在网站留言。

目录

一 介绍	1
二 排序算法一：深度优先搜索算法	1
三 总结	2
参考文献	2

一 介绍

我们希望借助计算机来安排一天工作，比如什么时候做饭，什么时候泡茶，什么时候洗碗之类的，我们一次只能干同一件事，而且做事情之前需要有一个先后顺序，比如你肯定是吃饭以后再刷碗，而不能做完饭以后直接刷碗。



这种结构一般被称为一个“有向无环图 (Directed Acyclic Graph, DAG)”。我们如果想要“工作”，就必须把“洗碗”和“泡茶”这两件事完成，也就是说，“工作”依赖于“洗碗”和“泡茶”；而“洗碗”和“泡茶”都依赖于吃饭。在计算机程序中，例如编译器和 makefile 就需要考虑编译和链接的顺序。

拓扑排序就是，给这种图进行一个顺序排序，使得任何一个节点都要在依赖项访问到以后再进行访问。当且仅当这个有向图没有环时，这种拓扑排序才是可能的。

二 排序算法一：深度优先搜索算法

基于深度优先搜索的算法并不是最基本的算法，但是较为简单，掌握了“树”数据结构即可实现。

该算法以任意顺序循环遍历图中的每个节点，启动深度优先搜索，当搜索到自拓扑排序开始以来已经访问过的任何节点或叶节点时终止。调用搜索的程序如下：

Algorithm 1 调用拓扑排序的程序

L 初始化为空 list: 用来装已经排好序的节点
while 存在节点没有 permanent 标记 **do**
 选择一个没有标记的节点 n
 调用 visit(n)
end while

visit(node n) 伪代码如下:

Algorithm 2 visit(node n): 基于深度优先搜索的拓扑排序

if (n 有一个 permanent 标记) return
if (n 有一个 temporary 标记) stop (这不是一个 DAG)
把 n 标记为 temporary
for 对于所有 $n \rightarrow m$ 的边 **do**
 visit(m)
end for
把 temporary 标记从 n 移除
把 n 标记为 permanent 标记

三 总结

暂时我们只先考虑深度优先搜索排序算法, 其他算法, 例如 Kahn 算法将在后面再陆续补充。

参考文献

[1] https://en.wikipedia.org/wiki/Topological_sorting