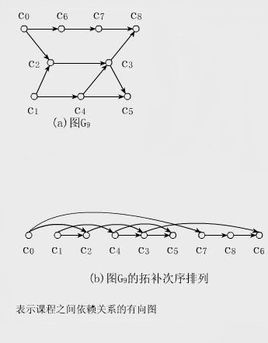
# 拓扑排序：

<https://blog.csdn.net/qq_35644234/article/details/60578189>

给你一个DAG(有向无环图) 进行拓扑排序，是将G中所有顶点排成一个线性序列，使得图中任意一对顶点u和v，若边(u,v)∈E(G)，则u在线性序列中出现在v之前。通常，这样的线性序列称为满足拓扑次序(Topological Order)的序列，简称拓扑序列。简单的说，由某个集合上的一个[偏序](https://baike.baidu.com/item/%E5%81%8F%E5%BA%8F/2439087)得到该集合上的一个[全序](https://baike.baidu.com/item/%E5%85%A8%E5%BA%8F/10577699)，这个操作称之为拓扑排序。 拓扑排序是对点来说的，输出是一组拓扑序的点。



## Kahn算法:

显然拓扑排序的顺序是每次取入度是0的点，然后删除与其相邻的边。

Kahn的算法的思路其实就是，手动模拟的拓扑排序，我们先使用一个栈保存入度为0 的顶点，然后输出栈顶元素并且将和栈顶元素有关的边删除(代码中不必真的删除边，而是改变其他点入度)，减少和栈顶元素有关的顶点的入度数量并且把入度减少到0的顶点也入栈。

## Dfs搜索法:

由于图是有向无环的，显然可以以每个入度是0的点开始dfs,搜到一个点如果他入度是0，和 Kahn很像就删除边(代码中不必真的删除边，而是改变其他点入度) ，dfs如果判断环路要比Kahn麻烦。建议用Kahn代码短

* 拓扑排序一定要考虑重边自环。因为这会影响度数。

# 模型

## 权值传播：

Hihocoder1175:给定图，图每个点有权值代表病毒，病毒会沿着有向图传播问传播完成后病毒总数，用拓扑序把每访问到点的权值向前加和。

## 有向无环图路径计数：

就是找入度为0的点按照Kahn算法的步骤，并且像权值传播那样，往前传播路径数量。起点权值初始化是1其余点都是0。