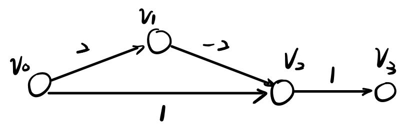
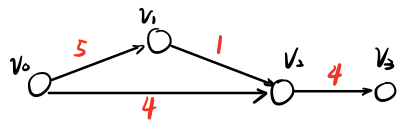
1. 解：a) 不可行，反例如下：





每一条边加3后 $v_0 \to v_3$ 的最短路径仍为

$v_0 \to v_2 \to v_3$，而非正确的 $v_0 \to v_1 \to v_2 \to v_3$.

b) 证：由s与h的定义知：

$\forall u \in V$,

若在可达u的所有路径中存在权值（之和）为负的路径，则其中权值和最小的路径对应的权值和即为 $h(s,u)$ 的值；否则 $h(s,u)=0$.

对 $\forall (u,v) \in E$，讨论以下四种情况.

① $h(s,u)=h(s,v)=0$.

说明从源到u, v的所有路径权值和均非负.

$\Rightarrow w(u,v) \geq 0$.

$\Rightarrow w'(u,v)=w(u,v) \geq 0$.

② $h(s,u)=0$, $h(s,v)<0$.

$\Rightarrow w(u,v)<0$, $h(s,v)=|w(u,v)|$

$\Rightarrow w'(u,v)=0$

③ $h(s,u)<0$, $h(s,v)=0$.

说明所有可达u的路径里权值和最小为负，记为min.

$h(s,v)=0 \Rightarrow |min| \leq w(u,v)$ 即 $w(u,v)+min \geq 0$.

$\Rightarrow w'(u,v)=w(u,v)+h(s,u)-h(s,v)=w(u,v)+min \geq 0$.

④ $h(s,u)$, $h(s,v)<0$.

说明 $w(u,v) \leq 0$ 或 $w(u,v)>0$ 且 $w(u,v) <$ |可达u的所有路径中最小权值和|

$\Rightarrow w'(u,v)=w(u,v)+h(s,u)-h(s,v)$

a. 当 $w(u,v) \leq 0$ 时, $h(s,v)-h(s,u)=w(u,v)$

$\Rightarrow w'(u,v)=0$.

b. 当 $w(u,v)>0$ 时, 由 $h(s,v)<0 \Rightarrow$

$w(u,v) <$ |可达u的所有路径中最小权值和| $=|h(s,u)|$

此时有 $h(s,v)-h(s,u)=w(u,v)$

$\Rightarrow w'(u,v)=0$.

综上, $w'(u,v) \geq 0$ 恒成立

c) 是；设 $\forall u,v \in V$, u, v两点间的最短路径长度为 $l(u,v)$

更新后为 $l'(u,v)$.

假设更新前 u到v的最短路径为 $\langle u, t_1, t_2, \cdots, t_n, v \rangle$

更新后 u到v的最短路径为 $\langle u, t_1', t_2', \cdots, t_k', v \rangle$

$\Rightarrow l(u,v)=w(u,t_1)+w(t_1,t_2)+\cdots+w(t_{n-1},t_n)+w(t_n,v)$

$l'(u,v)=w'(u,t_1')+w'(t_1',t_2')+\cdots+w'(t_{k-1}',t_k')+w'(t_k',v)$

$= w(u,t_1')+h(s,u)-h(s,t_1')+w(t_1',t_2')+h(s,t_1')-h(s,t_2')$
$+\cdots+w(t_{k-1}',t_k')+h(s,t_{k-1}')-h(s,t_k')+w(t_k',v)+h(s,t_k')-h(s,v)$

$= w(u,t_1')+w(t_1',t_2')+\cdots+w(t_{k-1}',t_k')+w(t_k',v)$
$+h(s,u)-h(s,v)$

$\Rightarrow \langle u, t_1', t_2', \cdots, t_k', v \rangle$ 是一条u到v的路径

故必有 $k=n$ 且 $t_i=t_i'$ $(i=1,2,\cdots,n)$

$\Rightarrow l'(u,v)=l(u,v)+h(s,u)-h(s,v)$

该式只与 s, u, v 有关

故用Dijkstra算法求得的最短路径在原图中仍为最短路径.

两个最短路径的关系为 $l'(u,v)=l(u,v)+h(s,u)-h(s,v)$.

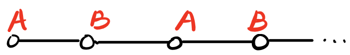
2. 解: 算法描述:

将每一个罪犯看作一个顶点，若两个罪犯可能产生冲突则在对应的两个顶点之间连边，最终产生一个无向图。问题转化为在图上用A、B给所有顶点标记，要求相邻顶点的标记不同。

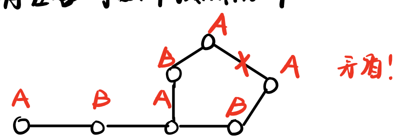
利用深度优先周游遍历所有顶点，并对顶点交替标记。每到一个顶点，则对与其相邻且尚未被标记过的顶点进行标记，若出现矛盾则说明没有合适的分配方案。

分配过程可分为如下三种：

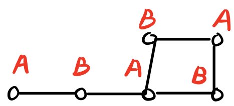
a. 无环



b. 存在含奇数个顶点的环



c. 存在含偶数个顶点的环



复杂度分析:

建图要读入n条边，时间为$O(n)$，

深搜作标记遍历每个顶点一次，时间也为$O(n)$，故时间复杂度为$O(n)$。

3. 证: 假设G存在两棵不同的最小生成树T_1, T_2.

$E(T_1)=\{e_1, e_2, \cdots, e_n\}$, $E(T_2)=\{e_1', e_2', \cdots, e_n'\}$.

(E中边按权重递增排列)

由于$T_1 \neq T_2$ 故$\exists k$, $e_k \neq e_k'$且k最小

不妨设 $weight(e_k) < weight(e_k')$

将e_k加入T_2. 设$e_k=(u,v)$，则

T_2中有环且环中包含e_k.

$\because$ k最小

$\therefore e_i = e_i'$ $(i=1,2,\cdots,k-1)$

$\therefore e_i'$不在环中 $(i=1,2,\cdots,k-1)$ 否则

T_1中有$e_1, e_2, \cdots, e_k$组成的环.

$\therefore$ 此时环中权重最小的边即为e_k.

$\therefore$ 删去环中除e_k外任意一条边即可得到一个权值总

小于T_2的生成树，这与T_2是MST矛盾!

$\therefore$ G具有唯一的最小生成树.

4. 解:

算法描述: 用拓扑排序完成算法。每条关系<i,j>对应图中的一条边根据所有先后关系可以得到一个无环有向图。对图中所有顶点利用队列进行拓扑排序并记录队列中元素情况，若某个顶点只在队列中单独出现了一次(取出该顶点后队列即空)，则说明排名是确定的。

算法复杂度: 以拓扑排序为框架，复杂度为$O(n+e)$

5. 解: 算法描述.

用Krustal算法求解.

n个城市对应n个顶点，除此之外再向图中加一个点作为打井的"水源地"，该点到其它n个顶点的边的权重即为该城市打井的费用，城市与城市之间边的权重即为管道费用。从图中任意一点出发建MST，MST的权重之和即为最小花费。

复杂度分析: Krustal算法建MST复杂度为$O(e\lg e)$，对于稀疏图，$O(e)=O(n)$，故算法复杂度为$O(n\lg n)$