现有一个带权的无向图 G，表示一个网络，顶点表示路由器，边权为链路带宽，已知图为连通图，定义一条路径 $p: u_0 u_1 \cdots u_n$ 的带宽为

$w(p) = \min\{w(u_i u_{i+1})\ i=0,1,\cdots,n-1\}$　$w(u_i u_{i+1})$ 为边 $u_i u_{i+1}$ 带宽

求出节点 a 到节点 b 的最大带宽路径

算法一：动态规划　设节点数为 $|V|$　节点 i, j 之间最大带宽为 $w(i,j)$

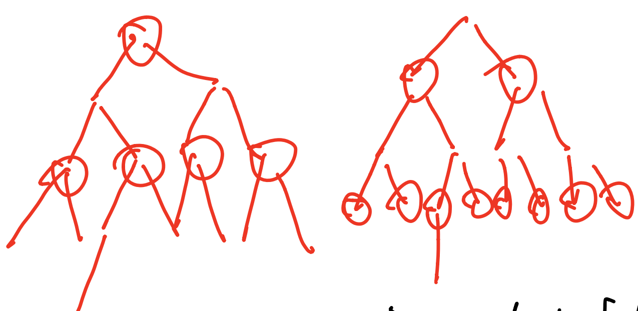
$$w(i,j) = \min_{k=1,2,\cdots,|V| \text{且} k\neq i,j}\{w(i,k), w(k,j)\} \quad O(V^3)$$

算法二：修改 Dijkstra 算法

算法三：修改 Kruskal　Thm 若 T 为图的最大生成树，则 for $s, t \in G$
T 中 s 到 t 的唯一路径 P_u 就是 G 中 s 到 t 的最大带宽路径

16-1 找硬币 d 问　$O(nk)$ 如何得到

树上 DP：一棵树，选择尽量多的顶点使之不相邻（两两）



给节点编号 $1, 2, \cdots, n$
$d(i)$ 代表以 i 为根的子树的独立集大小　求 $d(1)$，1 为根

$d(i, \text{true})$ 代表选择 i；$d(i, \text{false})$ 表示不选；

$$d(i) = \max(d(i,\text{true}), d(i,\text{false}))$$

$$d(i,\text{true}) = 1 + \sum_k d(k,\text{false})$$　k 为 i 在树上的儿子

$$d(i,\text{false}) = \sum d(k)$$

单个机器上的任务调度

一台机器，在零时刻到达了n个任务，每个任务完成所需时间为 t_i

第i个任务等待t会花费 $f_i(t)$ 的代价 $f_i(t)=c_i t$

任务调度是非抢占的

求最小的总等待代价

策略：贪心，但关键在于如何贪心以及贪心算法的正确性

按照 c_i/t_i 贪心

证明贪心选择的正确性

设任务调度的顺序是 $t_{s_1}, t_{s_2}, \dots, t_{s_i}, t_{s_{i+1}}, \dots, t_{s_n}$ ⇒ 策略1 ⇒ 开销 F_1

现在交换第i个任务与i+1个任务 $t_{s_1}, t_{s_2}, \dots, t_{s_{i+1}}, t_{s_i}, \dots, t_{s_n}$ ⇒ 策略2 ⇒ 开销 F_2

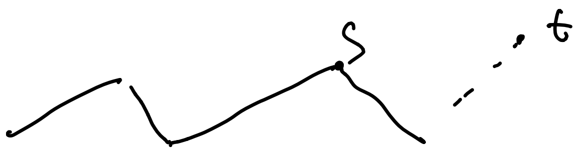
策略1与策略2相比前i-1个任务和第i+2个任务到第n个任务开销不变

$$F_1 - F_2 = c_{s_{i+1}} t_{s_i} - c_{s_i} t_{s_{i+1}} > 0$$

$$\Rightarrow c_{s_{i+1}}/t_{s_{i+1}} > c_{s_i}/t_{s_i}$$

补充：最大生成树上最大带宽路径的证明

现在有一棵最大生成树T，上面有两个节点 s, t



现在s到t有一条最大带宽路径 $P(s,t)$，树上s到t的路径为 $T(s,t)$

$w(P)$ 表示路径P的带宽，我们来证明 $w(T(s,t)) \geq w(P(s,t))$ 即可

⇒ 参考剪切粘贴技巧，反证