Review. $O(nT)$.

$O(nT)$ 是否是多项式时间?

① T: INT. — 占有 w bits.

随机存取机的变形.

$\Rightarrow T \le 2^w$, $\underline{w \ge \lg n}$

※ 根据 Transdichotomous model.

假设机器一个"字"的大小和问题大小一致.

$\Rightarrow$ 问题大小 (problem size): n.

字: w bit $\Rightarrow 2^w \ge n$. $\Rightarrow w \ge \lg n$.

$\Rightarrow$ 无上限. w 可以任意大.

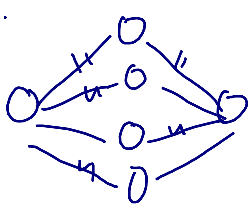
Greedy Algorithm.

① 最小生成树. 树? = connected acylic graph.

生成树 = 图G的边和所有节点构成的无环图

最小生成树 = 生成树 with minimal total weight.

例.

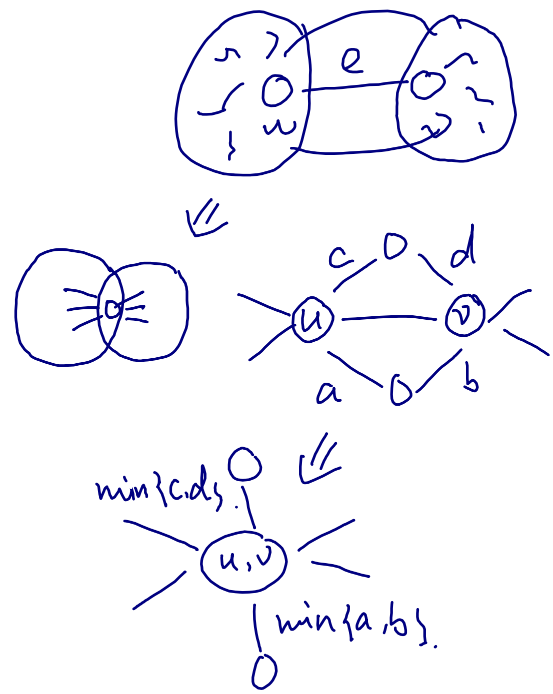


$\rightarrow$ 指数.

※ Greedy Properties: ① optimal substructure.

② greedy choice property: 局部最优.

MST 的最优子结构：$e=\{u,v\}$，属于某些 MST.



→ 将 u,v 合并.

→ T' 是 G/e 的 MST

则 $T' \cup e$ 是 G 的 MST.

Dynamic Programming.

① guess e in a MST.

② 合并 e.

③ 递归.

④ decontract e

⑤ 把 e 加入 MST.

回.

指数时间.

Lemma 1. T' 是 G/e 的 MST，则 $T' \cup \{e\}$ 是 G 的 MST.

证明：假设 $e \in T^* \rightarrow$ MST.

$\Rightarrow T^* - e$ is MST of G/e

$$w(T') \le w(T^* - e)$$

$$w(T' \cup \{e\}) = w(T') + w(e) \le w(T^* - e) + w(e) = w(T^*)$$

$\Rightarrow T' \cup \{e\}$ is MST

$u \in S,\ v \notin S$.

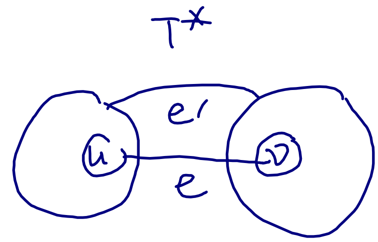
Lemma 2. 对于任意 cut $(S, V-S)$. any 最小权值的 crossing edge $e=\{u,v\}$.

一定在某个 MST 中

证明: cut & paste argument.

- let T^* be a MST of G.
- assume $e \notin T^*$

  则必有 $e' \in T^*$ cross the cut.



- $T^* \setminus \{e'\} \cup \{e\}$ 仍旧是生成树.

$$w(T^* - e' \cup \{e\}) = w(T^*) - w(e') + w(e)$$

$$\le w(T^*)$$

$\Downarrow$

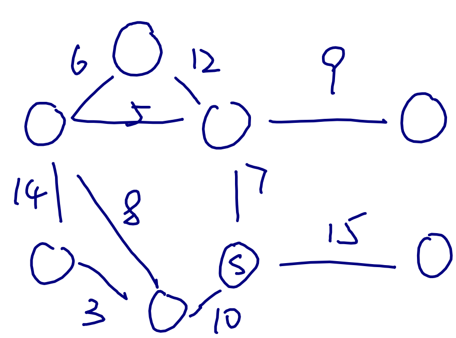
$\Rightarrow$ 仍旧是MST

Prim's 算法.

- 保存一个 priority queue Q on $V \setminus S$. 使得 $v.key = \min\{w(u,v) \mid u \in S\}$.
- init $Q = V$, $\begin{cases} s.key = \emptyset. & \text{for } s \in V \\ v.key = \infty. & \text{for } v \in V \setminus S. \end{cases}$
- until Q empty.

```
u = Extract-Min(Q).
for v ∈ Adj[u]:
    if v ∈ Q & w(u,v) < v.key
        v.key = w(u,v).
        v.parent = u.
return {(v, v.parent) | ∀v ∈ V}.
```

例子.



证明正确性.

树 T_S within $S \subseteq$ MST T^* 在 G.

用数学归纳法： 假设 $T' \in$ MST. T^*.

Time: 同 Dijkstra. $O(V\lg V + E)$.