1. 分治法

将 X 向量拆分为 x_1, x_2. 则 $H_k x$ 为 $\begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$= \frac{1}{\sqrt{2}} \begin{bmatrix} H_{k-1}x_1 + H_{k-1}x_2 \\ H_{k-1}x_1 - H_{k-1}x_2 \end{bmatrix}$$

记 $H_k x$ 复杂度 ~~$\Theta(\cdot)$~~ $T(n)$, n 为 x 维数. 则

有 $T(n) = 2T(\frac{n}{2}) + \Theta(n)$. 即先计算 $H_{k-1}x_1$ 与 $H_{k-1}x_2$,

再 $\Theta(n)$ 时间进行向量加法。

```
FJLT(H, x):
    res1 = FJLT(H_{k-1}, x1)
    res2 = FJLT(H_{k-1}, x2)
    return 1/√2 [res1 + res2
                 res1 - res2]
```

时间复杂度由主定理 $n^{\log_b a} = n^{\log_2 2} = n = \Theta(f(n))$, 故 $T(n) = \Theta(n\log n)$

2. (Hash 函数运用)

$\forall a, \hat{f}_a = g(a)\, C[h(a)]$. 记 X_p 为 $h(i_p) = h(a)$ 这一事件的示性函数. 则

$$\hat{f}_a = g(a)\left(\sum_{p=1}^{n} X_p\, g(i_p) \cdot c_p\right)$$

$= \sum_{p=1}^{n} X_p c_p \cdot g(a) \cdot g(i_p)$. 这其中 c_p 为常数. X_p 与 $g(a), g(i_p)$ 相互独立 (h, g 随机选取).

$$\text{故 } E(X) = E(\hat{f}_a) = \sum_{p=1}^{n} E(X_p c_p\, g(a)\, g(i_p))$$

$$= \sum_{p=1}^{n} c_p \cdot E(X_p) \cdot E(g(a) \cdot g(i_p))$$

$$= \sum_{i_p \neq a} c_p \cdot E(X_p) \cdot E(g(a) \cdot g(i_p)) + c_a \cdot 1 \cdot 1$$

而 $g(a) \cdot g(i_p)$ 取 $1/-1$ 的概率相同, 故 $E(g(a) \cdot g(i_p)) = 0$. 即前半部分为0. 故有:

$$E(X) = c_a.$$

3、平摊分析

解法一：聚合分析

注意到 在每位翻转之后，再经过 2^i 步才会再翻转，故

A[i] 将会翻转 $\lfloor \frac{n}{2^i} \rfloor$ 次，其所需时间开销

$$T(n) = \sum_{i=0}^{k} \lfloor \frac{n}{2^i} \rfloor \cdot 2^i \le \sum_{i=0}^{k} \frac{n}{2^i} \cdot 2^i = \sum_{i=0}^{k} n = kn$$

故单次的平摊开销 $O(k)$，即 $O(1)$

每一次置位的(均摊)代价 2^{k+1}，复位的均摊代价 0.

解法二：核算法

定义 INCREMENT 的均摊~~开销~~为 2^{i+1} (代价)，则每一次置位 (0->1) 都将存下 2^i 余额，而当该位复位 (1->0) 时，又恰好可以使用这些余额支付，鉴于初始化全为0，可知任一时刻余额都非负

于是该定义是合法的 ~~单次均摊开销 $O(2^{k+1})$~~，即 $O(1)$ n次操作的均摊代价 $O(n \cdot 2^{k+1})$，故单次 $O(1)$

解法三：势函数

定义势函数 $\Phi(S)$ = 计数~~器~~所表示的二进制值

故除了全1后 INCREMENT 清零外，$\Phi(S_i) - \Phi(S_{i-1}) = 1$，而

边界情况 $\Phi(S_i) - \Phi(S_{i-1}) = 1 - 2^k$

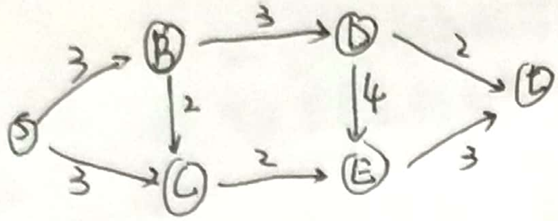
假设每次操作复位了 t_i 位，则除了边界外 $c_i = 2^{t_i+1} - 1$，而

边界情况则为 $2^k - 1$

故 $\hat{c}_i = \begin{cases} 2^{t_i+1} - 1 + 1 = 2^{t_i+1} \le 2^{k+1} \\ 1 - 2^k + 2^k - 1 = 0 \end{cases}$，即均摊操作 $O(1)$.

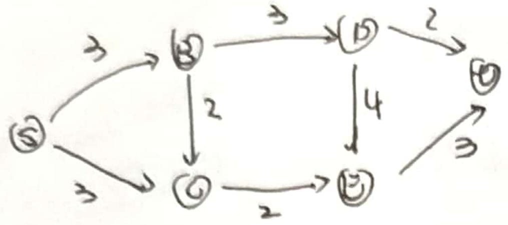
4. 网络流

Dinic算法，画出详细步骤

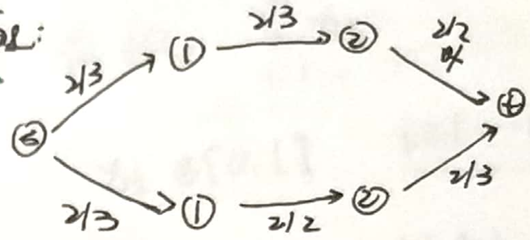


阶段一：

Gf:



GL:



有 blocking flow

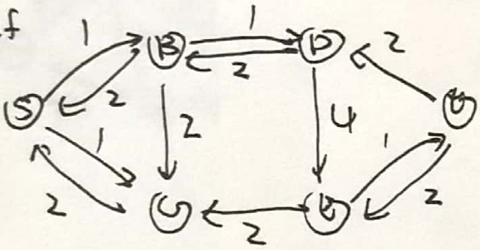
此阶段增广路长度为3

- S→B→D→t. 容量为2.
- S→C→E→t. 容量为2

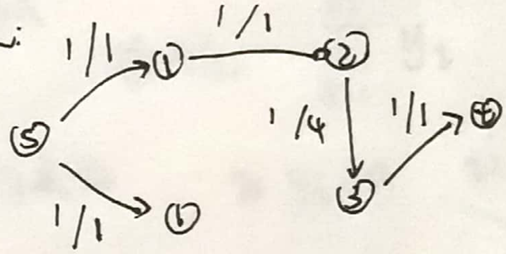
总 blocking flow 为4. 总 flow 值为4

阶段二：

Gf



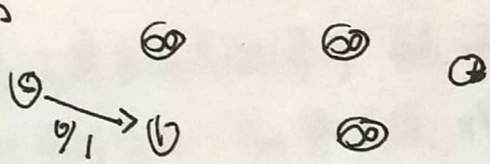
GL:



blocking flow: S→B→D→E→t. 容量为1

此阶段增广路径长度为5. 总 blocking flow 为1. 总 flow 值为5

最后 此时t已无法由S到达. GL为



算法结束

5. 动态规划

a) $P[S,i] = \begin{cases} 0 & |S| = 1 \\ \max\limits_{j} \{ P[S-\{i\}, j] + pref(j,i),\ j \neq i, j \in S \} & |S| > 1 \end{cases}$

b)

```
MAX_PREFERENCE:
    ~~for all i in S{1,...n}:~~                          // 初始化边界条件
        ~~P[{i}, i] = 0~~        for all Pcell in P:
                                     P[][] = 0
                                  // s 表示 |S|
    for s = 2 to n:
        for all subset of {1...n} S whose size is s:
                                  // 视情况更新 P[S,i], 求max_j...
            for all j ∈ S:
                P[S,i] = max{P[S,i], P[S-{j}, j] + pref(j,i)}

    return max_i {S, i}
```

c) 共有子问题数 $2^n * n$ 个. 每个子问题花费时间 $O(n)$ 求最大值，故总时间复杂度 $O(n^2 \cdot 2^n)$

7. 近似算法

给定 $G=(V,E)$，$D\subseteq V$ 若有 $\forall u\in V$ 满足 $u\in D$ 或 $\exists(u,v)\in E$ 且 $v\in D$。近似算法如下：

```
D ← Φ
mark all vertices in V as "undominated"
while there are undominated vertices in V
{   pick a v from V (undominated)
    let Nv be the set of v and all neighbors
    D ← D∪Nv
    mark Nv and Nv's neighbors as "dominated"
end while
```

(a) 解释算法结束D为为支配集

(b) 记 Δ 为 G 中顶点的度最大，即 $\Delta=\max_{v\in V}\deg(v)$，证近似度 $1+\Delta$。

(c) 任一支配集 AD 必含每个 N_v 中至少一个顶点

解：

a) 根据算法终止性可知，$\forall u\in V$ 它在算法终止时被标记为"dominated"，即 $\exists N_v$，使 $u\in N_v$ 或 $\exists w\in N_v$，$(w,u)\in E$。这正是支配集的定义。故算法结束时，任一顶点 u 都处于被支配状态。

b) 设最优支配集 OPT，由于 $\forall v\in V$，它一定是 OPT 中的点或其中某点的邻居，一个 OPT 中的点各自支配点之并正是 V 全体。即 $|V|\le(1+\Delta)\cdot OPT$。而一个点至多支配 $1+\Delta$ 个点，所有 OPT 中的点各自支配点之并正是 V 全体。而 $D\le|V|$ 显然，故 $D\le(1+\Delta)\cdot OPT$

c) 反证法，若 $\exists v$ 及对应 N_v，$AD\cap N_v=\phi$，则此时顶点 v 即不在 AD 中，又无任何邻居在 AD 中，故 v 不会被 AD 支配，矛盾。故 $AD\cap N_v\neq\phi$，至少有一个点的交集。

(PS. 也 b 也可用 c 中，$D\le k\cdot(\Delta+1)$，k 为迭代轮次，每一次 $|N_v|\le\Delta+1$，而 $OPT\ge k$)