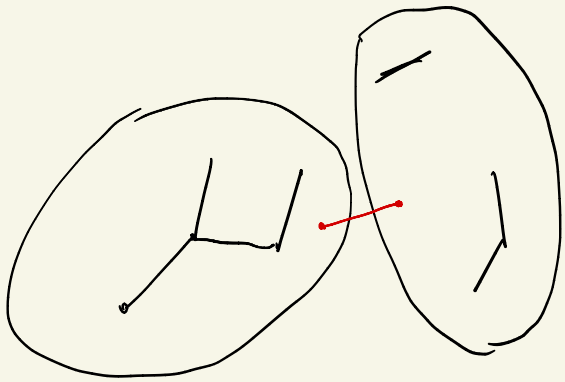
Lec 7

Review. Cut property.



Prim. grow the tree from a single vertex.

T_i. partil tree after i-th iteration.

$S_i = V(T_i)$.

pick the lightest edge $e \in E(S_i, V-S_i)$.

$T_i = T_i \cup \{e\}$.

procedure prim $(G=(V,E), \{w_e\}_{e \in E})$.

for all $u \in V$.

cost $(u) = 0$

pre $(u) = NIL$.

pick an initial vetex $u_0 \in V$.

cost $(u_0) = 0$.

H = priority queue (V).

```
while H ≠ φ.
    v = deletemin(H).
    for each {v, z} ∈ E.
        if cost(z) > w(v, z)
            cost(z) ← w(v, z)
            pre(z) = v.
```

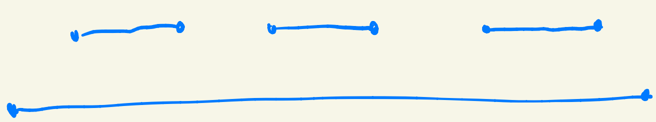
$O(m \log n)$.

Task assignment

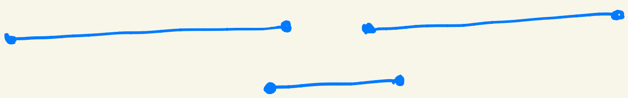
* Given n tasks, each with starting time $s(i)$ and finishing time $f(i)$.

* Find a way to complete as many tasks as possible.

Strategy 1: always choosing the earliest task.
(smallest starting time)



Strategy 2. Choose the shortest task.



Strategy 3. Choose the one with earlest ddl.

This is a correct algorithm.

Suppose $S = \{s_1, \dots, s_\ell\}$.

One of Optimal $O = \{t_1, t_2, \dots, t_m\}$.

Goal. Prove that $\ell = m$.

" S is always ahead of O ".

$\forall i \le \min(\ell, m)$. $f(s_i) \le f(t_i)$.

proof by induction.

Huffman encoding

How to encode a message efficient?

Symbol	Frequency
A	70 m
B	3 m
C	20 m
D	37 m.

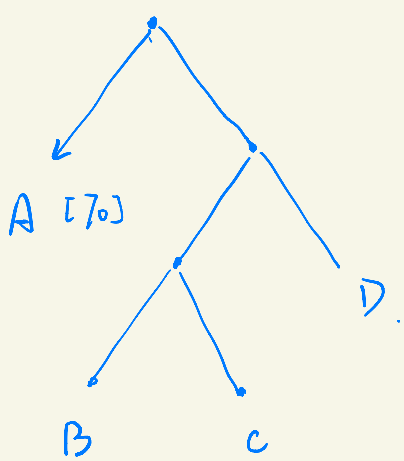
00 = A.
01 = B
10 = C
11 = D.

$\Rightarrow 2 \times ((70 + 3 + 20 + 37))$
$= 260.$

Intuition short codewords for frequent letters.

"prefix-free". $\{0, 01, 11, 001\}$. 001 is ambiguous

Full binary tree representation.

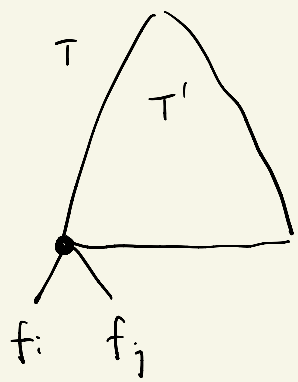


$$\text{Cost} = \sum_{i=1}^{n} f_i \cdot (\text{depth of } i).$$

= sum of all nodes except the root.

prop. two symbols with smallest frequencies are children of the lowest internal vertex.

prop. Cost of that optimal tree. = $\text{Cost}(T') + f_i + f_j$



procedure Huffman(f).

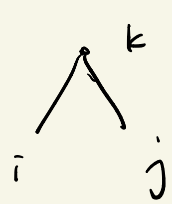
H = priority queue of $[n]$. w.r.t. f.

for $k = n+1$ to $2n-1$.

  i = deletemin(H),

  j = deletemin(H).

  create

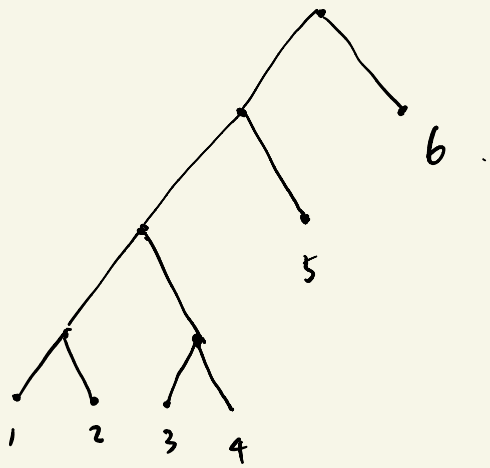


$O(n\log n)$.

  $f(k) = f[i] + f[j]$.

  insert (H, k).

When $f_i = m \cdot 2^{-k_i}$ for every $i \in [n]$, $k_i \in \mathbb{N}$.



$m = 16$.

$f_1 = f_2 = \cdots = f_4 = 1$

$f_5 = 4$

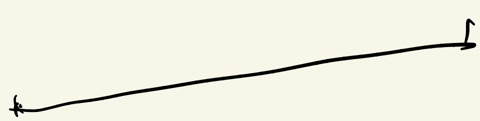
$f_6 = 8$.

$$\text{Cost} = \sum_{i=1}^{n} f_i \left(\text{depth of } i\right) = m \boxed{\sum_{i=1}^{n} k_i \cdot 2^{-k_i}}$$

$$= \underbrace{\sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i}}$$

Entropy of $(p_1, \dots, p_n)$.

Entropy v.s. Compressibility.



$p_1, \dots p_t$. $t = 2^s$.

$m = 2^k$

$p_2 = 1 - 2^{-k}$.

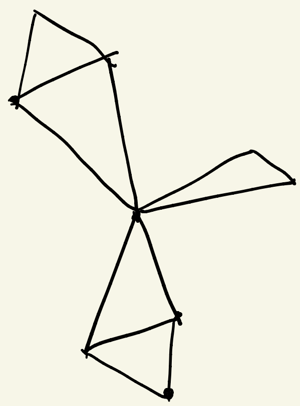
$p_1 = 2^{-k}$,

$p_1 \log_2 \frac{1}{p_1} = 2^{-k} \cdot k$.

S.

$p_2 = 0$.

Set Cover



Input: Universe U.

$S_1, \dots, S_m \subseteq U$.

Output: $i_1, \dots, i_k$

such that $\bigcup_{j=1}^{k} S_{i_j} = U$.

Greedy Pick S_i with largest uncovered elements.

prop. If the optimal solution uses k sets, the solution of the greedy algorithm uses at most $k \ln n$ sets.

pf. n_t. # of elements not covered after t iteration.

$n_0 = n$

$$n_{t+1} \le n_t - \frac{n_t}{k} = n_t\left(1 - \frac{1}{k}\right) < n_t e^{-1/k}$$

$$\Rightarrow n_t < n \cdot e^{-t/k} \qquad t > k \ln n \Rightarrow n_t < 1.$$

Dinur & Steurer 2013. $\forall \varepsilon > 0$

no $(1-\varepsilon)\ln n$ approx

unless $NP = P$.