ADA 2023 second midterm

D&C

- Median of median:
  $T(n) = T(\frac{n}{5}) + T(\frac{7n}{10}) + O(n) = O(n)$.
- Karatsuba:
  $T(n) = 3T(\frac{n}{2}) + O(n) = O(n^{\log_2 3}) \approx O(n^{1.58})$
  $\begin{cases} \mathbf{C}_2 = A_1B_1, \mathbf{C}_0 = A_0B_0 \\ \mathbf{C}_1 = A_1B_0 + A_0B_1 = (A_0 + A_1)(B_0 + B_1) - C_2 - C_0 \end{cases}$
  For any **constant** k, $O(n^{\log_k 2k-1})$ multiplication is possible.
- Strassen's:
  $T(n) = 7T(\frac{n}{2}) + O(n^2) = O(n^{\log_2 7}) \approx O(n^{2.807})$
  $\begin{cases} \mathbf{M}_1 = (A_1 + A_4)(B_1 + B_4), \mathbf{M}_2 = (A_3 + A_4)B_1 \\ \mathbf{M}_3 = A_1(B_2 - B_4), \mathbf{M}_4 = A_4(B_3 - B_1), \mathbf{M}_5 = (A_1 + A_2)B_4 \\ \mathbf{M}_6 = (A_3 - A_1)(B_1 + B_2), \mathbf{M}_7 = (A_2 - A_4)(B_3 + B_4) \end{cases}$
  $\begin{cases} \mathbf{C}_1 = M_1 + M_4 - M_5 + M_7 \\ \mathbf{C}_2 = M_3 + M_5, \mathbf{C}_3 = M_2 + M_4 \\ \mathbf{C}_4 = M_1 - M_2 + M_3 + M_6 \end{cases}$
- Victor Pan's:
  $T(n) = 143640T(\frac{n}{70}) + O(n^2) \approx O(n^{2.795122})$
- Master Thm: Let $T(n) = aT(\frac{n}{b}) + f(n)$.
  $\begin{cases} T(n) = \Theta(n^{\log_b a}) & f(n) = O(n^{\log_b a - \varepsilon}) \\ T(n) = \Theta(n^{\log_b a}\log^{k+1} n) & f(n) = \Theta(n^{\log_b a}\log^k n) \\ T(n) = \Theta(f(n)) & f(n) = \Omega(n^{\log_b a}) \text{ and} \\ & af(\frac{n}{b}) < cf(n), c < 1 \end{cases}$
  Case 1
  $$T(n) = \sum_{j=0}^{\lceil \log_b n \rceil} a^j f\left(\frac{n}{b^j}\right) = O\left(\sum_{j=0}^{\lceil \log_b n \rceil} a^j \left(\frac{n}{b^j}\right)^{\log_b a - \varepsilon}\right)$$
  $$= O\left(n^{\log_b a}\sum_{j=0}^{\lceil \log_b n \rceil} \left(\frac{n}{b^j}\right)^{-\varepsilon}\right) = O\left(n^{\log_b a}\right)$$
  Case 2 $\boxed{\mathbf{k \geq 0}}$, ext $k = -1 \Rightarrow \log\log n, k < -1 \Rightarrow 1$
  $$T(n) = \Theta\left(\sum_{j=0}^{\lceil \log_b n \rceil} a^j \left(\frac{n}{b^j}\right)^{\log_b a} \log^k\left(\frac{n}{b^j}\right)\right)$$
  $$= n^{\log_b a}\Theta\left(\sum_{j=0}^{\lceil \log_b n \rceil} \log^k\left(\frac{n}{b^j}\right)\right) = \Theta(n^{\log_b a}\log^{k+1} n)$$
  $$\left(\because \frac{\lceil \log_b n \rceil}{2}\log^k(\frac{n}{2}) \leq \sum \log^k\left(\frac{n}{b^j}\right) \leq \lceil \log_b n \rceil \log^k n\right)$$
  Case 3
  $$f(n) \geq \frac{a}{c}f\left(\frac{n}{b}\right) \geq \left(\frac{a}{c}\right)^2 f\left(\frac{n}{b^2}\right) \geq \cdots \geq \left(\frac{a}{c}\right)^{\lceil \log_b n \rceil} f(1)$$
  $$f(n) = \Omega\left(\left(\frac{a}{c}\right)^{\lceil \log_b n \rceil}\right) = \Omega\left(n^{\log_b a - \log_b c}\right) = \Omega\left(n^{\log_b a + \varepsilon}\right)$$
  Counter Examples: Non polynomial differences
  $$T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\log n}, T(n) = \Theta(n \log\log n)$$

DP

- Fib Sequence, Matrix Chain Mul,
- String Alignment Problem $O(|A||B|)$
  Input: string A, B, table $S : \Sigma \times \Sigma \to \mathbb{R}$.
  Output: $\max \sum S(A'_i, B'_i)$ (inserted some −).

String Matching

- $\begin{cases} Z_0 = \text{undefined} \\ Z_i = \max\limits_{S_{[0,d)} = S_{[i,i+d)}} d \end{cases}$
- Compute next Z_i: Let $j = \arg\max Z_j + j$. $Z_i \leftarrow \max\{0, \min(Z_j - (i - j), Z_{j-i}\}$.
- If $Z_i + i = Z_j + j$, bruteforce increase. Otherwise $S_{Z_i + i} = S_{i-j+Z_i} \neq S_{Z_i}$.

Greedy

- Huffman Code Optimality: Swap smallest two with deepest leaves.

MST

- For any vertex v, its smallest out-going edge is on the MST.

Fibonacci Heap

Prim's MST, Dijkstra.

Maintain a linked list where each node is a heap.

- `find_min`: $\Theta(1)$
- `merge`: concat list, $\Theta(1)$
- `insert`: use `merge` $\Theta(1)$
- `decrease`: if the node violates heap property, cut from parent and mark parent. If a node is marked twice cut the subtree and mark its parent again (and remove the mark). $\Theta(1)$ amortized.
- `extract_min`: remove the minimum (children become new trees) and keep linking two same degree nodes. $\Theta(\max\deg)$ + $\Theta(1)$ amortized.

Let the potential be #trees $+ 2$#marked (explains amortized $\Theta(1)$)

Lemma. Size of a subtree of degree k node has at least F_{k+2} nodes.
$k = 0 \Rightarrow F_2 = 1$. For $k > 0$ sort children c in dec. order of time. Then child $\deg c_i \geq i - 2$. $2 + \sum_{i=2}^{k} F_k = F_{k+2}$.

Remark. If we run $\log\log n$ Borůvka steps, then the remaining graph only needs $O(m + n)$ with Prim's Algorithm. Thus MST can be solved under $O(m \log\log n)$.