Homework #5:

"Closest pair"

string → $(\#_a, \#_b)$

abaaab → (3, 2)

Every String → A 2-dim point

$d(a,b)$ → distance between two points

1. Run Closest pair on all parts from the mapping.

n parts

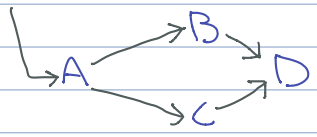
2. Target: $O(n \cdot m)$

Case 1: $\log(n) \leq m$ then I works.

Case 2: $\log(n) \geq m$ then it doesn't

Topological Sort on a DAG

directed acyclic graph



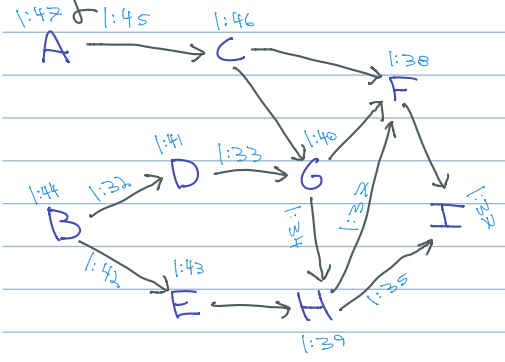
Example: ABCD

Not Example: ABDC

Given a DAG G, we can find a list of all nodes $V_1, V_2, \ldots V_n$ s.t. each edge $u \to v$ in G, the node u appears before node V in the list.

Such a list is called a Topological Sort of G.

Algrothim



1. Start at a node with in-degree = 0
2. Run depth-first search.
3. Output Resultant in order.

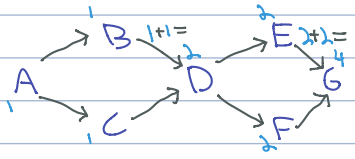
Result: ACBEDGHFI

Applications of Topological Sort (on final exam)

Graph = a high dimesion object

When you need an algrothim that runs in one direction, first run Topological Sort for a linear structure.

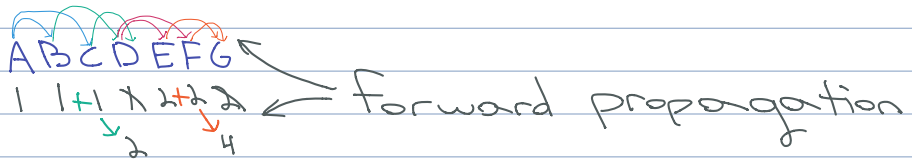
Example:



Question: How many paths from A to G?

2^k possible paths. So, cannot enumerate all paths.

Try Topological Sort.



So 4 total paths.