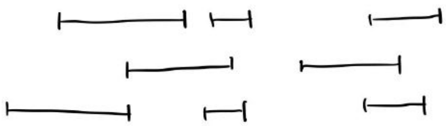
$\boxed{\text{Lec 14}}$ Treaps + Aug Table

Aug Tables

→ Treap with at each node : key, value, reduced value , size

Useful for... eg. interval problems



Where is there 2 overlaps?
Where is there ...

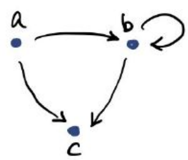
Graphs

Informal : verts connected by edges

More formally <u>directed graph</u> $G = (V, E)$, $n = |V|$, $m = |E|$

set of edges, represented by vert tuple
set of verts

Ex.



$G = (\{a, b, c\}, \{(a,b), (a,c), (b,c), (b,b)\})$

<u>Fact</u> $m \leq n^2$ (tight upper bound on num edges)

num distinct graphs with n verts... $2^{(n^2)}$

<u>Undirected graph</u> $G = (V, E)$, $E \subseteq \binom{V}{2}$

set of sets with 2 verts

<u>Types of Graphs</u>

- Multgraph $G = (V, E)$, E is multiset

- Hypergraph $G = (V, E)$, $E = \mathcal{P}(V)$

so an edge can link $\neq 2$ verts
normal graph ⇒ 2-uniform hypergraph

- Bipartite graph $G = ((U, V), E)$, $E \subseteq U \times V$, $|U| = n_u$, $|V| = n_v$

  Fact there are $2^{(n_u \cdot n_v)}$ distinct undirected bipartite graphs

Applications

- Utility graph — electricity, internet, water, gas, ...
  verts ← location   edges ← connections

- Dependence graph — compiler control flow,

- Social network graph

- Taxonomy graph — phyrogenetics, evolution

- Mesh network

- Markov chain

- documents with links

- state graph

Mathematical Defs

Def $N_G(u)$ is neighbourhood of u in G $= \{v \in V \mid \{u, v\} \in E\}$

$N_G^+(u)$ is the outgoing nbors $\{v \in V \mid (u, v) \in E\}$
$N_G^-(u)$ ... $\{v \in V \mid (v, u) \in E\}$] for directed

$\deg(u) = |N_G(u)|$

$\deg^+(u) = |N_G^+(u)|$
$\deg^-(u) = |N_G^-(u)|$

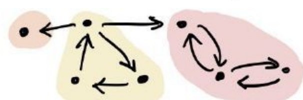
Def Path is an alternating seq of verts & edges
Length of path is num edges in path
Simple path is path without repeating vert nor edge
Cycle starts and end at same vert
Simple cycle cycle without repeating vert nor edge except at start

Def $\delta_G(s, v)$ = len of shortest path from s to v in G

$R_G(u,v)$ = v reachable from u to v
viz. $\exists$ path from u to v

<u>Connected component</u> is a subset of verts s.t. every vert is reachable from every other vert] undirected

<u>Strongly connected component</u> is subset of verts s.t. every vert is reachable from every other vert] directed



<u>Def</u> <u>Forest</u> — graph without cycle] undirected

<u>Tree</u> — forest with one connected component]

<u>DAG</u> — directed acyclic graph

Graph representations

- Edge set

  (V set , (V × V) set) for some set repr

  Edge membership query → whatever lookup cost is in set repr
  Check nborhood → filter edges set then map to extract... $O(m)$

- Adjacency matrix

  ((V × V) key , $\mathbb{B}$ value) table

  Good for dense graph
  Bad for sparse graph

  <u>Def</u> G is dense if $m \simeq n^2$
  sparse otherwise

- Adjacency set

  (V key , V set) table

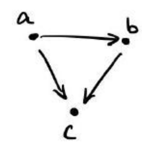
  $\{ a : \{b, c\},$
  $b : \{c\},$
  $c : \{\} \}$

- Adjacency seq

  Define $\{0, \ldots, n-1\} \leftrightarrow V$

  int seq seq

  $\langle \langle 1,2 \rangle, \langle 2 \rangle, \langle \rangle \rangle$



- Adjacency list