March 7, 2019 10:59 AM

no discussion sessions tomorrow

Dynamic Prog. (Cont'd)

Last Ex: Parsing

Given CFG $G = (V, T, P, S)$ &
string $x = a_1 a_2 \cdots a_n \in T^*$,
decide whether $x \in L(G)$

Note: may assume that all productions in P
are of the form

$$A \to BC \qquad (A, B, C \in V)$$

$$\text{or} \quad A \to a \qquad (A \in V,\ a \in T)$$

Chomsky Normal Form (CNF)

e.g.

$S \to AS \mid AB$

$B \to SA \mid AC \mid 1$

$C \to AB \mid CC \mid 0$

$A \to BC$

10110111 $\in L(G)$?

(e.g. $A \to BCDE$ can be converted to
$A \to XY$, $X \to BC$, $Y \to DE$)

CYK Alg'm (Cocke-Younger-Kasami '70)

Define subproblems: $(1 \le i \le j \le n,\ A \in V)$

Define subproblems: ($\underline{1 \le i \le j \le n,\ A \in V}$)

$f(i,j,A)$ = true iff the substring $a_i a_{i+1} \cdots a_j$ can be generated by A.

Final Ans: $f(1,n,S)$

Base cases: $f(i,i,A)$ = true iff $A \to a_i$ is in P.

Recursive formula:

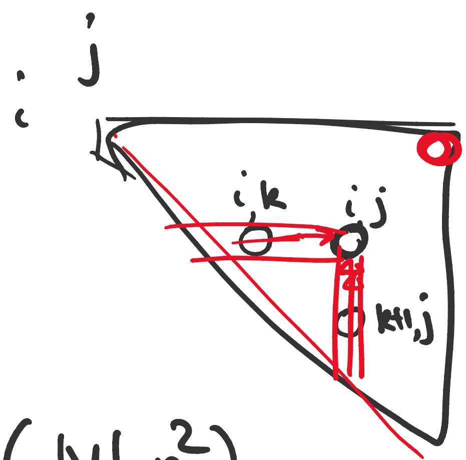
choices for 1st deriv. step in opt sol'n for $f(i,j,A)$:

- which production $A \to BC$
- split $\underbrace{a_i \cdots a_k}\underbrace{a_{k+1} \cdots a_j}$

$$\boxed{f(i,j,A) = \bigvee_{\substack{A \to BC \\ \text{in } P}} \bigvee_{k=i}^{j-1} \big(f(i,k,B) \wedge f(k+1,j,C)\big)}$$

Evaluation order:

increas. order of $j-i$



Runtime:

\# subproblems $= O(|V|\,n^2)$

time per subproblems $= O(|P|\,n)$

$\Rightarrow O(|V|\,|P|\,n^3)$ total time more carefully, $\boxed{O(|P| n^3)}$ time

(since each production $A \to BC$ is considered once)

Rmk: can be improved to $O(|P| n^{2.373})$ (Valiant '75)

for most PL, there are faster linear-time alg'ms ...

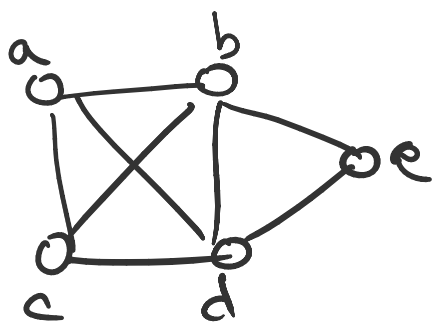
Graph Alg'ms

Def A graph G is (V, E), $n = |V|$, $m = |E|$

V = set of vertices

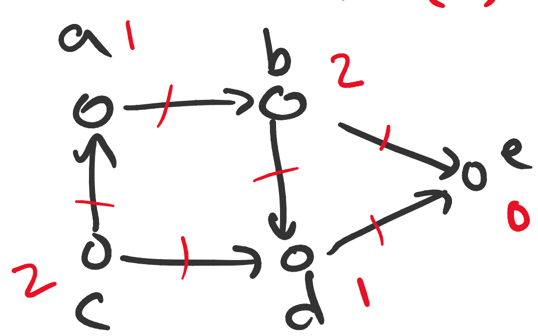
E = set of edges of form (u,v) ← if directed   with $u,v \in V$

or uv ← if undirected (or $\{u,v\}$)



undirected

$G = (\{a,b,c,d,e\}, \{ab, ad, ac, bc, \dots\})$



directed

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

Basic concepts: adjacency, incidence, paths, connectedness, cycles, trees, ...

(appl'ns: road networks, social networks,

(appl'ns: road networks, social networks, web page links, ...)

Note: $\underbrace{n-1 \le m}_{\text{if connected}} \le n^2$

Representation

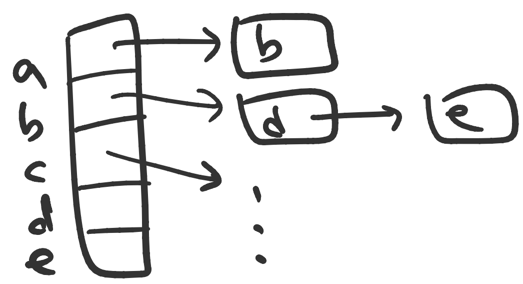
- adjacency matrix

  $A[u,v] = \begin{cases} 1 & \text{if } (u,v) \in E \\ 0 & \text{else} \end{cases}$

  $\Theta(n^2)$ space (good for dense graphs when m close to n^2)

- adjacency lists

  for each $u \in V$,
  store linked list $Adj[u] = \{v : (u,v) \in E\}$



Space $O\left(n + \sum_{u \in V} |Adj[u]|\right)$

$= O\left(n + \underbrace{\sum_{u \in V} \text{out-deg}(u)}\right)$

$= O(n + m)$

(good for sparse graphs when m close to n)

Basic Problems:

is there a path from s to t?

- is there a path from s to t?
- find all vertices reachable from s?
- is graph connected?
- find all connected components

directed?

⋮