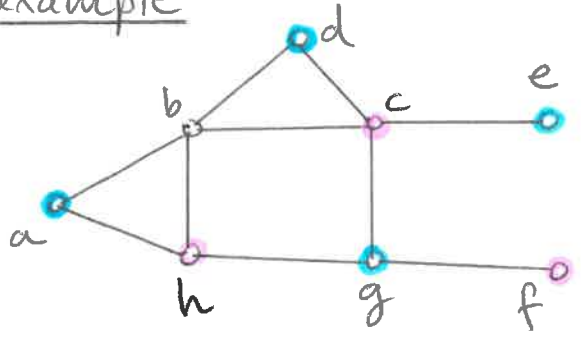
Solving a special case of the Independent Set problem using dynamic programming

Definition

An independent set (IS) of an undirected graph $G(V, E)$ is a subset of vertices $V' \subseteq V$ such that each edge in E is incident to at most one vertex in V'.

IS problem: given an undirected graph $G(V, E)$, find an independent set of maximum size (or maximum cardinality)

example



independent sets:

$\{h, c, f\}$ size 3

$\{a, d, g, e\}$ size 4 → maximum size

$\{a, d, e, f\}$ size 4

- NP-complete problem
- Brute-force algorithm with $RT = O(2^V \cdot E)$
  $n = |V|$, then we can express the RT as $RT = O(2^n \cdot n^2)$
- If we restrict the graph G to be a tree, then we can solve the IS problem in polynomial time using dynamic programming.

Special case — G is a tree.

Dynamic programming algorithm

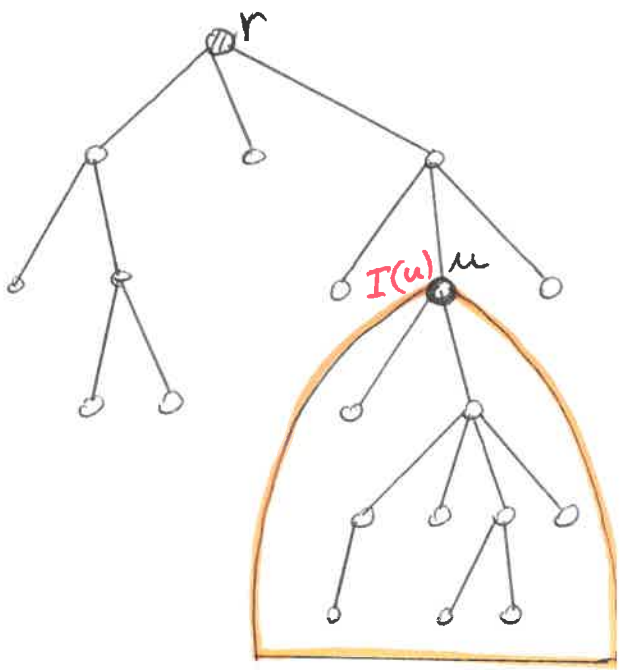
- let r be the root node
- compute the height of each node
- sort the nodes in increasing order of their height

- compute the IS of each node, by taking the nodes in sorted order

$I(u)$ – cardinality of a maximum-size IS of the subtree rooted at node u.

$I(u) = 1$ if the node u is a leaf (i.e. height = 0)

$$I(u) = \max\left\{ 1 + \sum_{\substack{w \text{ is a} \\ \text{grandchild} \\ \text{of } u}} I(w),\ \sum_{\substack{v \text{ is a} \\ \text{child of } u}} I(v) \right\}$$

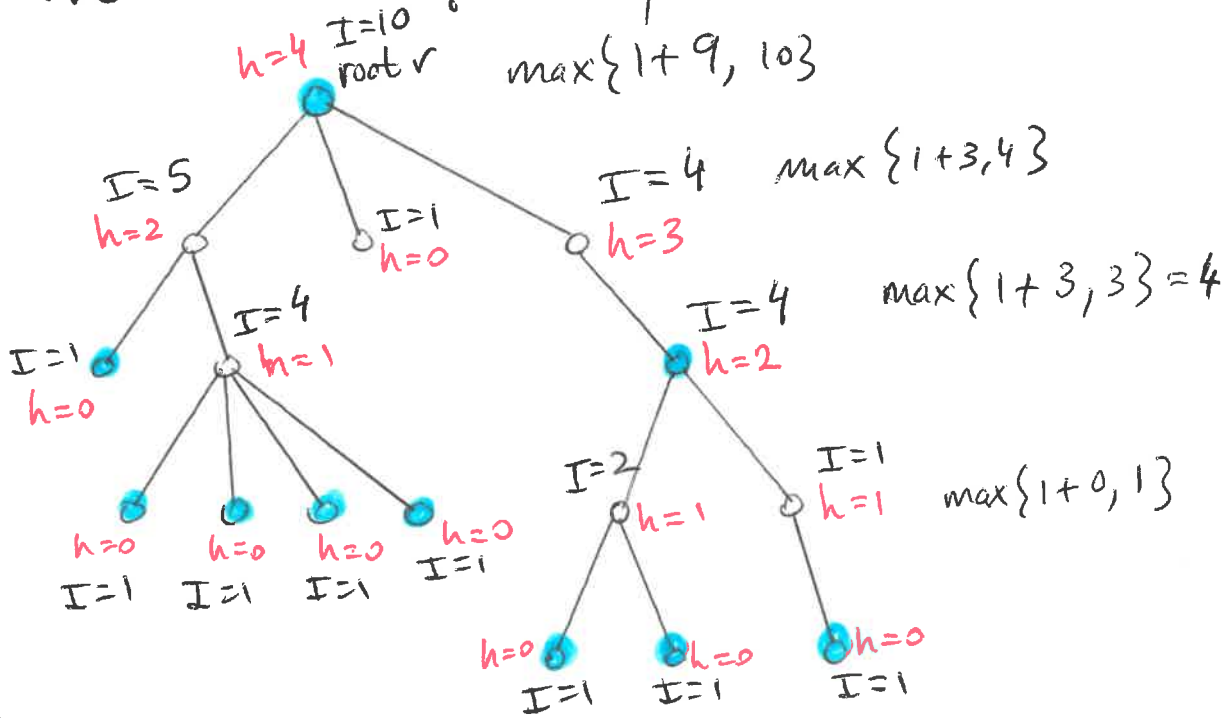


Two cases:
- node u is in the IS
- node u is NOT in the IS

$RT = O(n^2)$ where n – number of nodes

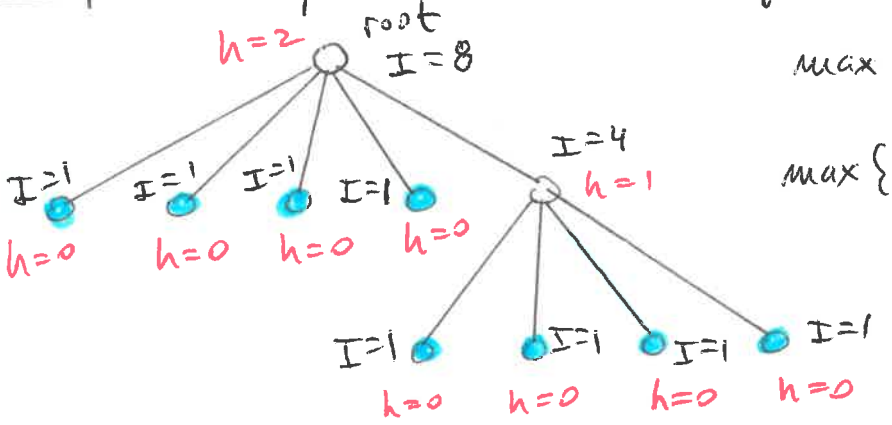
example

Compute the maximum-size IS for the tree:



cardinality of a max-size IS is 10.

example. Compute the maximum-size IS for the tree:



$\max\{1+4, \underline{8}\}$

$\max\{1, \underline{4}\} = 4$

Cardinality of a max-size IS is 8.