

## Przykład zastosowania (sort. leksykograficzne)

Dane:  $T_1, T_2$  - drzewa ukorzenione  
↑ jest wskazany korzeń

Problem: czy  $T_1 \cong_{\text{izo}} T_2$ ?



↑ będziemy sprawdzać, czy na każdym poziomie jest tyle samo wierzchołków tego samego typu

Opis drzew:

• 0

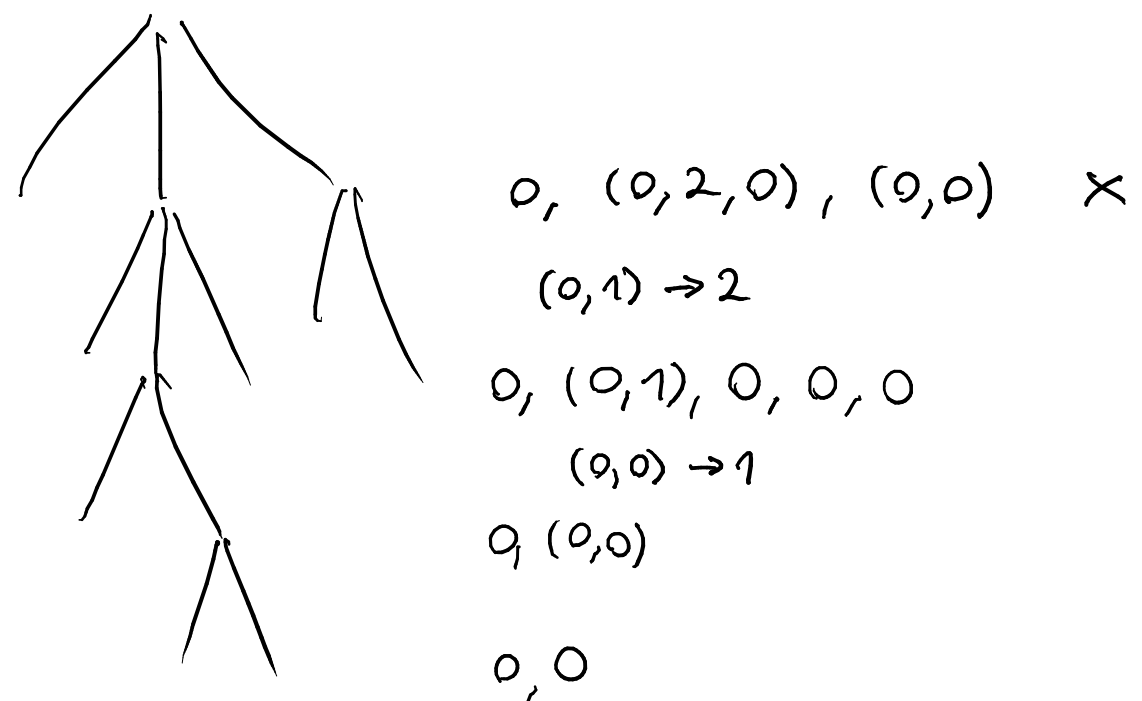
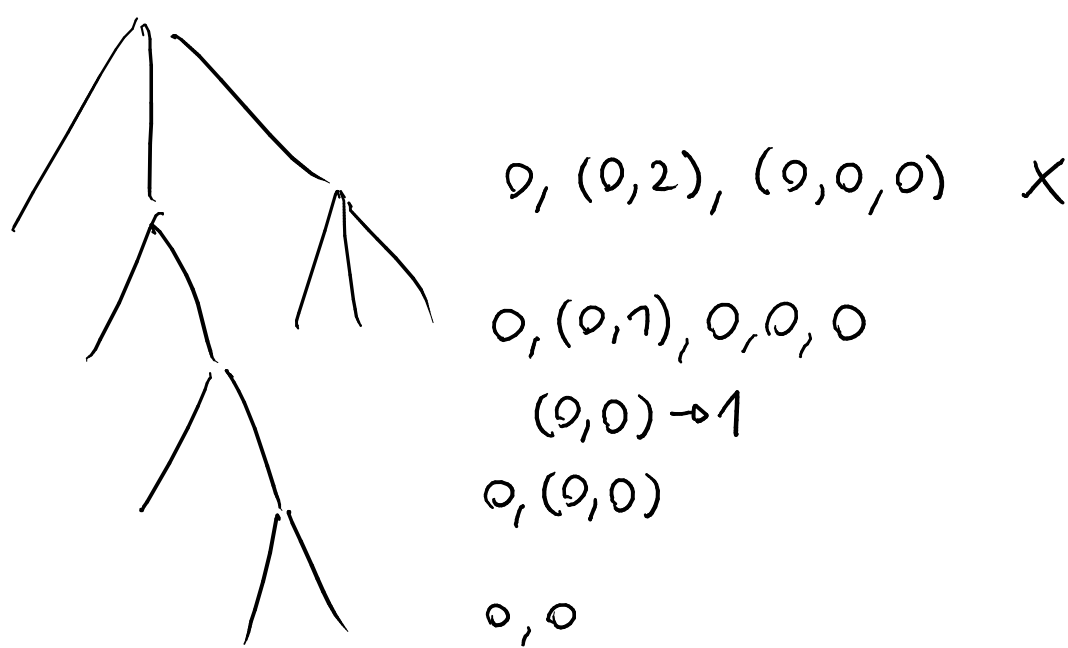
^ (0,0)

^ ((0,0),0)

• opisy robią się długie

→ będziemy je kodować

np. (0,0) → 1



Zał.

$h(T_1) = h(T_2)$

na każdym poziomie  $T_1, T_2$  mają tyle samo liści

$\forall v \in T_i \quad \text{kod}[v] \leftarrow 0$

for  $j$  in  $[1.. \text{depth } T_1]$

$S_j \leftarrow$  zb. wierz.  $T_i$  z poziomu  $j$ ,  
które nie są liśćmi

foreach  $v$  in  $S_j$

$\text{klucz}[v] := \text{sort} \circ \text{map } \text{kod} \text{ } \$ \text{ children } v$

end

$L_i = \text{sortOn}^* \text{klucz } S_j$

↑ tutaj sortujemy leksykograficznie  $O(k + \sum L_i)$

$L'_i = \text{map } \text{klucz } L_i$

if  $L'_1 \neq L'_2$

then return False

else foreach  $v$  in  $L_i$

$\text{kod}[v] := 1 + \text{rank}(\text{klucz}[v], \{\text{klucz}[u] \mid u \in L_i\})$

end

$L_i := (\text{liście } T_i \text{ z poziomu } j) \uplus L_i$

end

return True

jaka na tym nie tracisz czasu?

tworzyć to tam:

mając uporządkowaną listę, patrzymy na ojca każdego wierzchołka i w ten sposób tworzymy posortowaną listę kodów dzieci ojca

$\text{sortOn} :: \text{Ord } b \Rightarrow (a \rightarrow b) \rightarrow [a] \rightarrow [a]$

↑ co to?  
Rozmiar alfabety

A jeśli drzewa nie są ukorzenione?

(Liczenie)

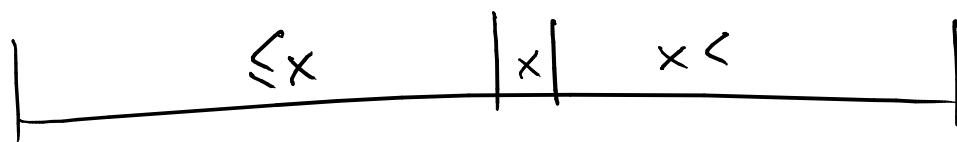
→ ukorzenie w wierzchołkach centralnych

(maksymalnie 2 kandydatów na drzewo)

A co w grafach?

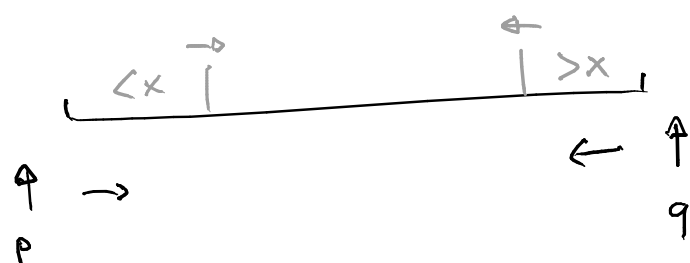
Nie znamy algorytmu wielomianowego, ale  
nie wiadomo, czy jest NP-trudny

# Quicksort



partition

• jak zrobić efektywnie (pamięciowo)?



choose Pivot

- deterministycznie
  - pierwszy element:  $O(n^2)$   
ale oczekiwane  $O(n \log n)$
  - mediana  
istnieje algorytm  $O(n)$ !  
ale spora stała
  - mediana z 3 el.

• losowo (można mieszać)

pesymistycznie  $O(n^2)$

```
do  
  x ← random xs  
while do  
  a, b ← random xs  
  return  $\min[a, b] \leq x \leq \max[a, b]$ 
```



U    □    □  
⊕    ⊕    ⊕

$\text{rank}(p)=1 \vee \text{rank}(p)=2 \rightarrow \text{podział } 1 \text{ } (n-1)$