

Curs 6

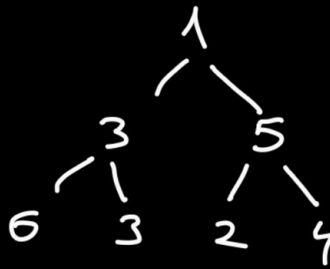
Arbori : Aplicații

• Arbori Binari Parțial Ordonați

→ heap sort (representare prin tablou)

$$\begin{cases} h_i \leq h_{2i} \\ h_i \leq h_{2i+1} \end{cases} \quad (\text{copii săi})$$

„5” nu e arborizabil



1

3	5	6	3	2	4
---	---	---	---	---	---

aplic din nou
algoritmul

1 2

3	4	3	5	6
---	---	---	---	---

și tot așa...

Aleg cel mai din dreapta
subarbor care nu e frunză.
Verific dacă e arborizabil.

Bubble Sort mi spune dacă tabloul e deja
sortat.

Problema Concordanței

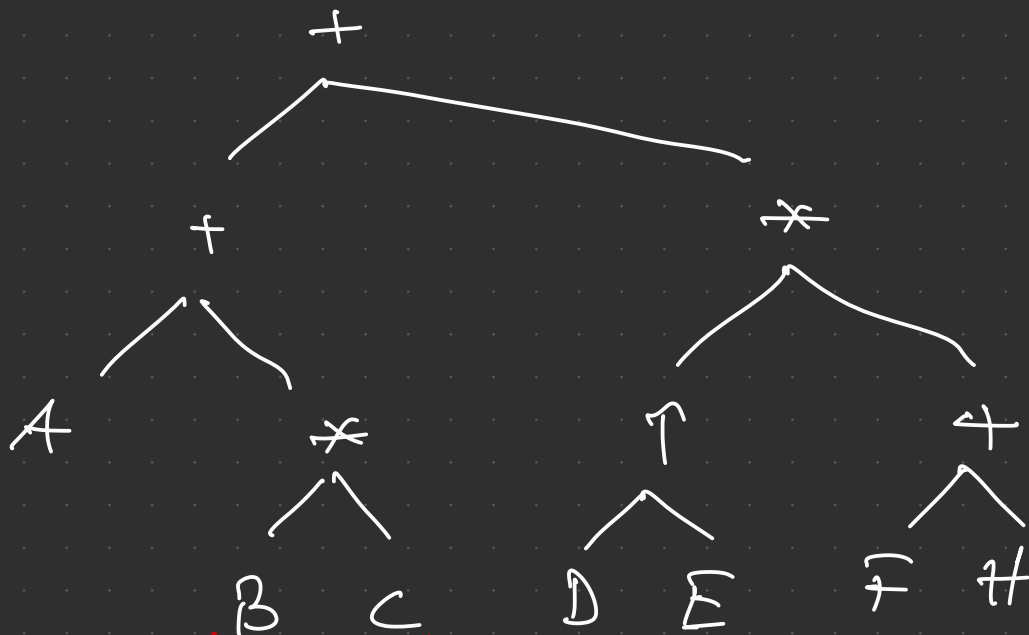
→ def. frecvenței de apariție a cuvintelor într-un text (nu hashmap <3)

→ arbore binar ordonat

- se sortează alfabetic șirul de cuvinte

Pentru Expresii

$A + B * C + D \uparrow E * (F + H)$



reverse Polish notation

Polish notation

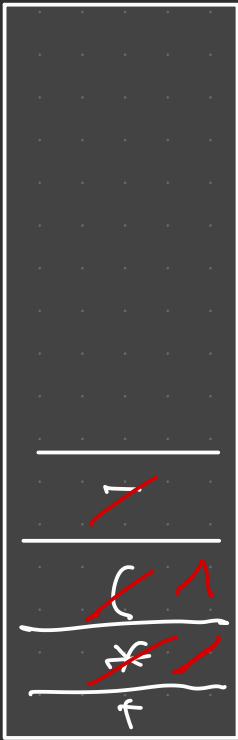
Preordine: $++A * BC * \uparrow DE + FH$

Inordine \rightarrow ^{algebraică} reconstituție a expresiei

Postordine $\rightarrow ABC * + DE \uparrow FH + * +$

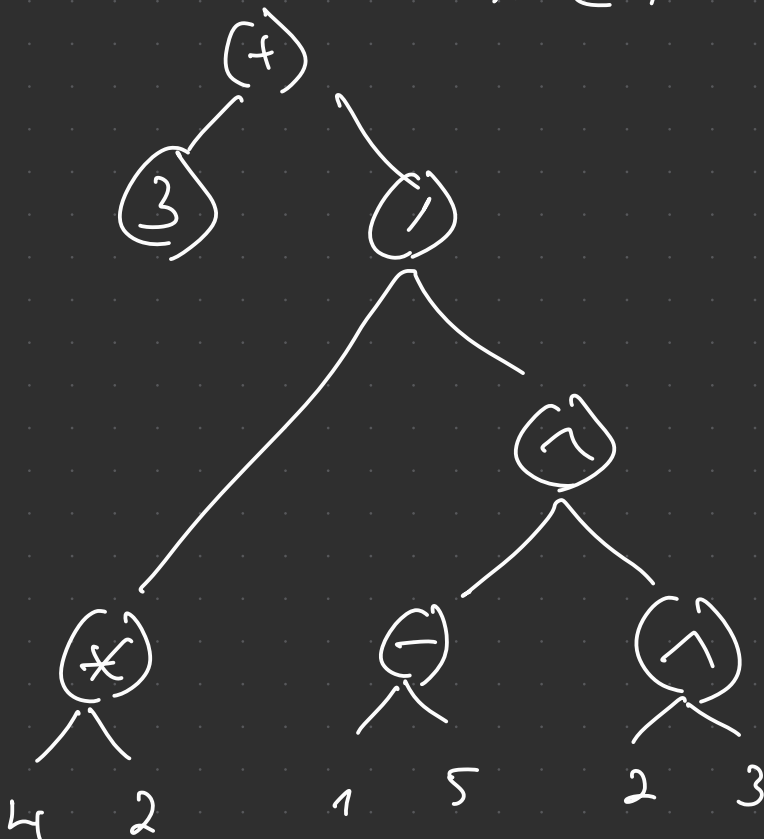
$$3 + 4 \times 2 / (1 - 5) \wedge 2 \wedge 3$$

$$3 \ 4 \ 2 \ * \ 1 \ 5 \ - \ 2 \ ^ \ 3 \ ^ \ / \ +$$



(s max precedență

egale sau
precedență mai mare \rightarrow push
mică \rightarrow pop + operație



$$3 \ 4 \ 2 \ * \ 1 \ 5 \ - \ 2 \ ^ \ 3 \ ^ \ / \ +$$

$$3 + 8 / 16^3$$

