Math 322

Prefix Code / Huffman Code

labels are leaves only.

(1) subtree$_1$, subtree$_2$, subtree$_3$

" " "

$x\%$ $y\%$ $z\%$

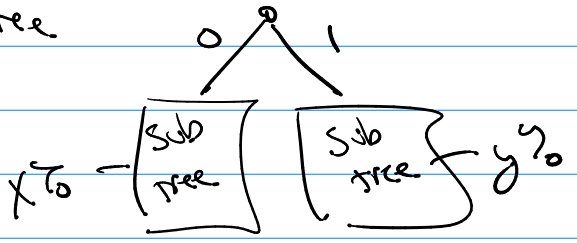
(2) pick the two %'s that are least

Ex: a (10%), 0/1 → q, r (12%), 0 (12%)

tie? take the taller one

(3) Make a new subtree

if $x < y$



and new % of subtree $= (x+y)\%$

Ex: a (10%), 0/1 → q, r (12%), 0 (12%)

0 (22%), 0/1 → a, 0/1 → q, r, 0 (12%)

(0) a, e, i, o, d, b, f
20% 18% 12% 25% 10% 8% 7%

(1) a, e, i, o, d, [0 1 / f b] 15%
20% 18% 12% 25% 10%

(2) a, e, o, [0 1 / d i] 22%, [0 1 / f b] 15%
20% 18% 25%

(3) [a, o], [0 1 / d i] 22%, [0 1 / 0 1 e / f b] 33%
20% 25%

(4) o, [0 1 / 0 1 e / f b] 33%, [0 1 / a 0 1 / d i] 42%
25%

(5)(6) [0 1 / 0 1 / a 0 1 / d i] [0 1 / 0 1 e / f b]

prefix code / Huffman code

a: 00
o: 10
e: 111
i: 011
d: 010
f: 1100
b: 1101

dad: 01000010

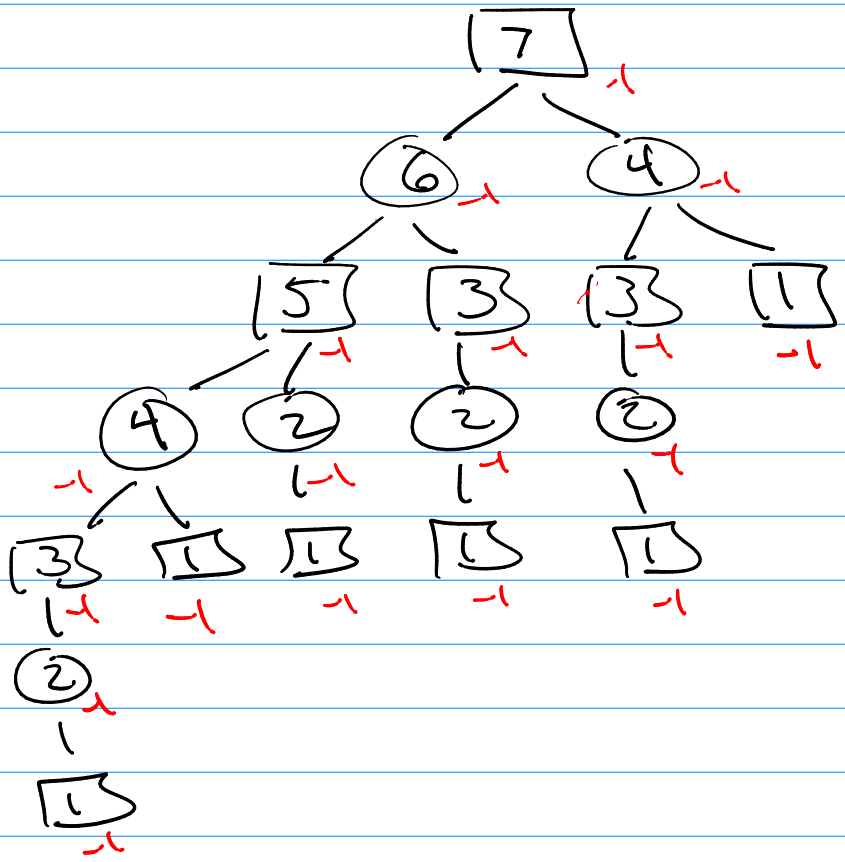
game tree (ex) Variation of nim.

(take last object (or all) = lose)

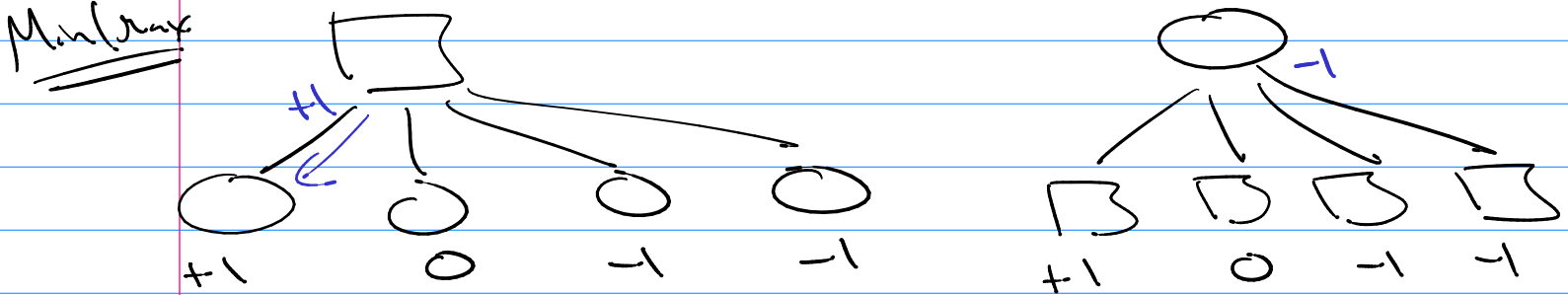
Rule: a) 1 pile

b) take 1 or 3

(ex) Start with 7 stones

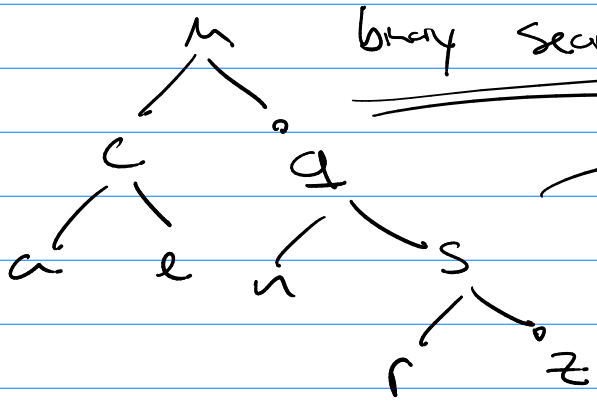


Min/max



getting elements out of trees

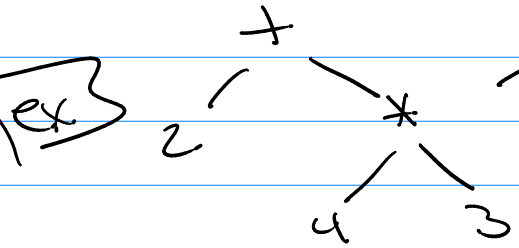
ex



binary search tree

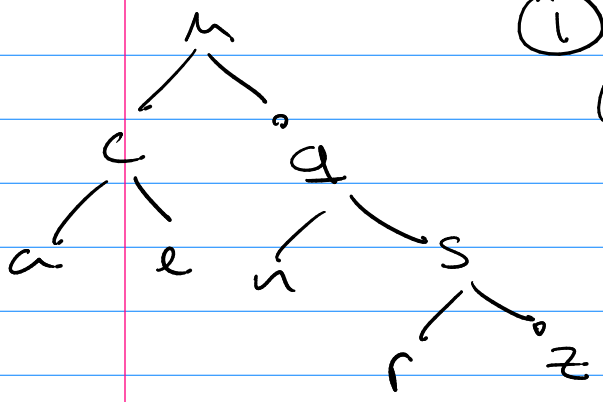
G → linear form?

ex

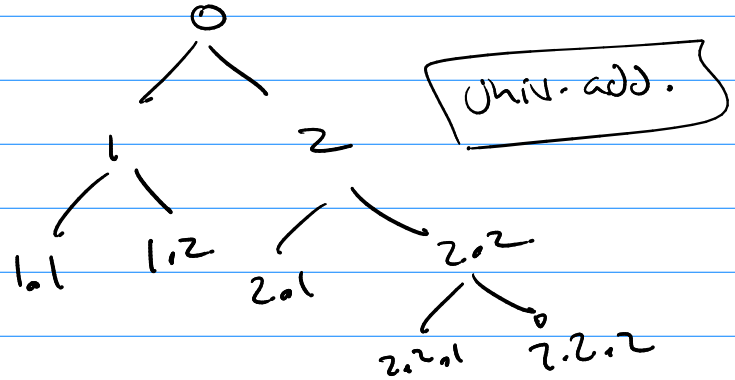


→ linear list & todo's

11.3 ## tree traversals



① Universal Address System
(plus) lexicographic ordering



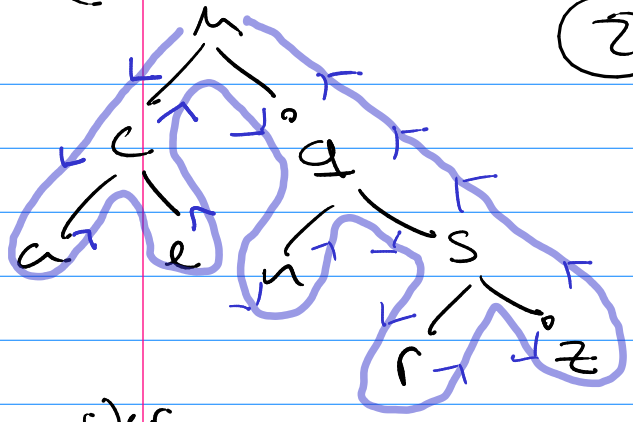
univ. add.

lexicographic sort

0, 1, 1.1, 1.2, 2, 2.1, 2.2, 2.2.1, 2.2.2

m, c, a, e, q, n, s, r, z

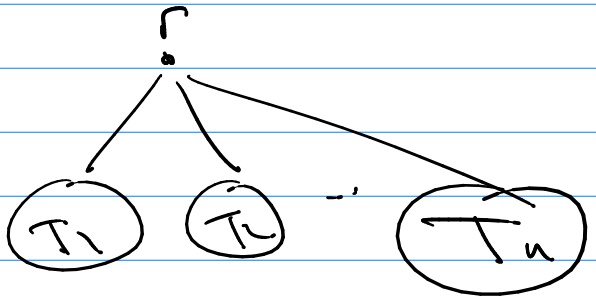
(traversals)



pre-order

m, c, a, e, q, n, s, r, z

(2) Pre-Order Traversal



(1) Visit r

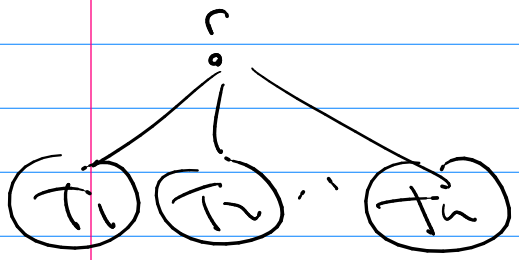
(2) Visit T_1 in pre-order

(3) Visit T_2 in pre-order

⋮

(n+1) Visit T_n in pre-order

(3) In-Order Traversal



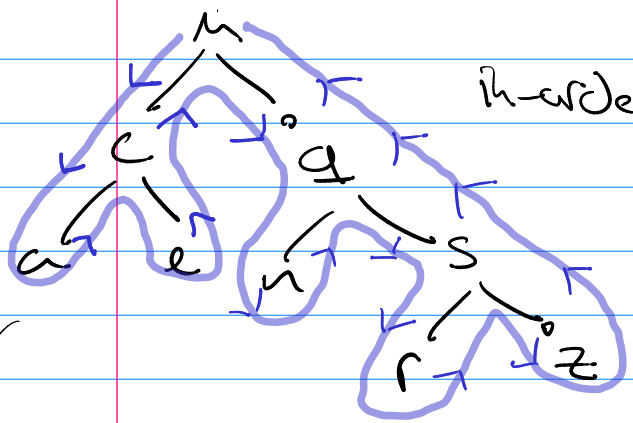
(1) Visit T_1 in in-order

(2) Visit r

(3) Visit T_2 in inorder

⋮

(n+1) Visit T_n in inorder



in-order: a, c, e, m, n, q, r, s, z

(4) Post-Order Traversal

post order: a, e, c, n, r, z, s, q, m

(1) Visit T_1 in post-order

(2) Visit T_2 in post-order

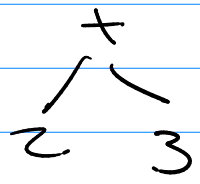
⋮

(n) Visit T_n in post-order

(n+1) Visit r

Applications of traversal: Math operations

(ex) binary operator



linear rep. of math operations

pre-order:	+, 2, 3	← prefix notation
in-order:	2, +, 3	← infix notation
post-order:	2, 3, +	← postfix notation

(ex) unary operator:

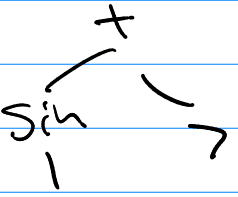
sin
1
3

prefix: sin, 3

infix: 3, sin

postfix: 3, sin

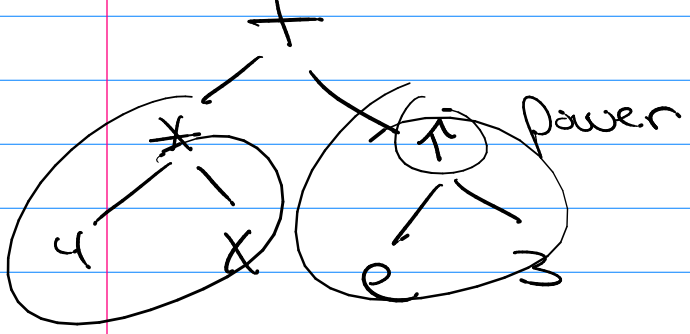
"book" notation: sin(3) + 7 →



prefix: +, sin, 3, 7

infix: 3, sin, +, 7

postfix: 3, sin, 7, +

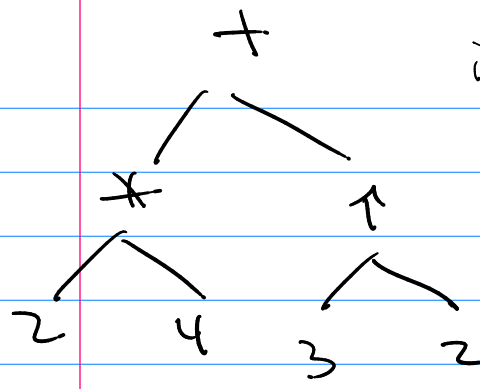


prefix: +, *, 4, x, ↑, e, 3

postfix: 4, x, *, e, 3, ↑, +

infix: (4, *, x) + (e, ↑, 3)

↑ must have grouping around subtrees



infix $(2,*,4), +, (3,\uparrow,2)$

postfix $2,4,*,3,2,\uparrow,+$

prefix $+,*,2,4,\uparrow,3,2$

Jan Łukasiewicz

prefix : Polish notation

postfix : reverse polish notation

$$\int\left(\frac{\sin(x^2)+3x-1}{\sqrt{x+2}}\right)dx$$