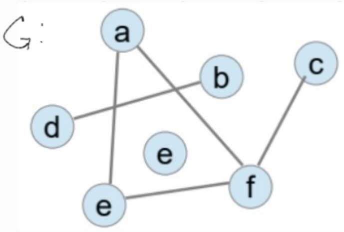
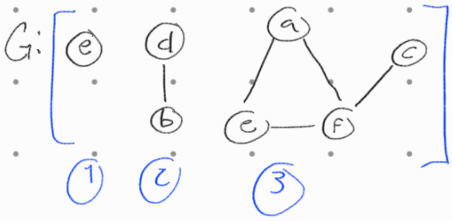
Exercise 2.1

What are the connected components of the following graph?

every connected graph has 1 component
every disconnected graph has 2 components





Three Connected Components

Exercise 2.2 - obligatory (6 points)

a) How many leaves has a complete binary tree of height h = 9?

2^(h-1) => 2^8 => 256 leaves

b) How many nodes has a complete binary tree of height h = 9?

2^h -1 => 2^9 - 1 => 511 nodes

c) Which minimum height has a binary tree consisting of 100 nodes?

n = 2^h -1 => 101 = 2^h => h log(2) = log(101) => h = log(101)/log(2) => min.height = 7

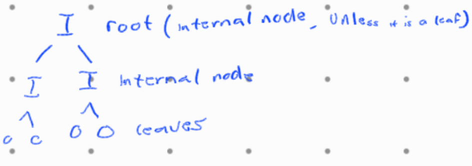
min.h = $\lceil \log_2(N + 1) \rceil$ => $\lceil 6.658 \rceil$ => 7

ceiling

d) Is the following statement true? (give a short justification).

"Every complete binary tree has more internal nodes than leaves."

No, a node with no children is a leaf of a tree; all non-leaves are internal nodes.



$\# \text{Internal Nodes} = \left(\frac{\text{Nodes} - 1}{2}\right)$

$\# \text{Leaves} = \left(\frac{\text{Nodes} + 1}{2}\right)$

$N = (2i + 1)$

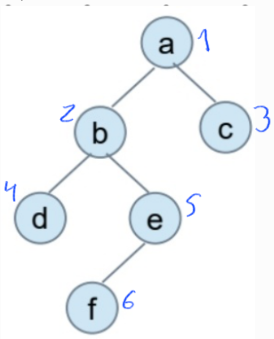
$i = \left(\frac{\text{Nodes} - 1}{2}\right)$

$N = 2L - 1$

$L = \left(\frac{\text{Nodes} + 1}{2}\right)$

Exercise 2.3 - obligatory (6 points)

a) Traverse this binary tree in

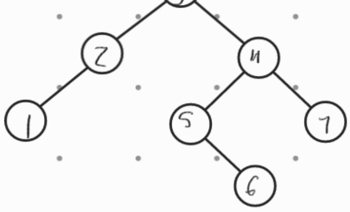


- Pre-order: a bdef c
- In-order: dbfe a c
- Post-order. dfeb c a

b) Traversing a binary tree yields this results:

- Pre-order: 3, 2, 1, 4, 5, 6, 7 MLR
- In-order: 1, 2, 3, 5, 6, 4, 7 LMR
- Post-order: 1, 2, 6, 5, 7, 4, 3 LRM

Reconstruct the tree from this traversal sequences.

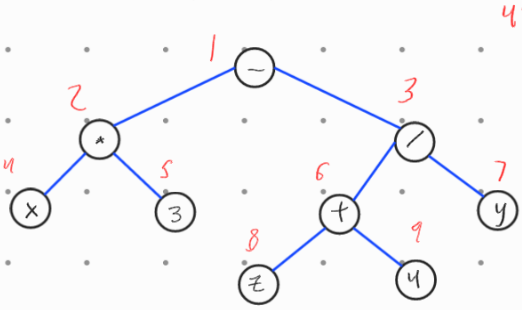


Exercise 2.4 - obligatory (3 points)

a) Draw an abstract syntax tree of the following expression:

x * 3 − (z + 4) / y

4 5 2 8 9 6 7 3 1



b) [optional, 3 bonus points] Generate stack-machine code for the evaluation of the expression (see lesson 5).

Post-order: x 3 * z 4 + y / -

	Code:
load x.	0: iload_0
const 3.	1: iconst_3
mult.	2: imul
load z.	3: iload_1
const 4.	4: iconst_4
add.	5: iadd
load y.	6: iload_2
div.	7: idiv
sub.	8: isub
	9: ireturn

Exercise 2.5

A simple two player game has the following rules: Player A begins and chooses a number from the set {1, 2}. Subsequently the players B and A choose alternatingly a number from the set {1, 2, 3}, but it is not allowed to select the same number as the adversary in the preceeding move. All of the choosen numbers from both players are summed up. If a player reaches the sum 6 he wins the game. If the sum exceeds 6, the player loses.

Example 1:
A: 1 sum = 1
B: 3 sum = 4
A: 2 sum = 6, Player A wins

Example 2:
A: 2 sum = 2
B: 1 sum = 3
A: 2 sum = 5

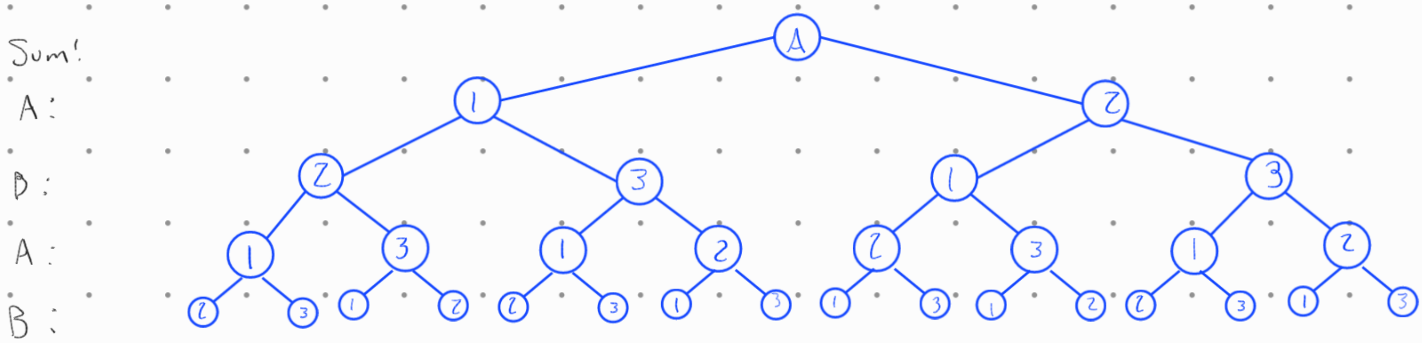
0: Player A chooses # set {1, 2} A
1: Player B Chooses # set {1, 2, 3} B
2: Player A chooses # set {1, 2, 3} A
3: Player B Chooses # set {1, 2, 3} B
4: Player A Chooses # set {1, 2, 3} A
5: Player B Chooses #. set. {1, 2, 3} B

B: 1 sum = 6, Player B wins

A so called decision tree can be used to depict all possible moves of the game. The nodes of the tree represent states of the game and the child nodes of a node are the states after the next possible moves.

height: 5

a) Draw a decision tree for this game.



b) Who will win if both players play in an optimal way?

The player whom begins the game will win if both players play in an optimal way

Exercise 2.6 - obligatory (5 points)

Let u1 = ab and u2 = bbb be string over alphabet S = {a,b}. Compute:

(1) u2u1 = bbbab

(2) u1u1 = abab

(3) $u1^{0}$ = ε

(4) | u1 * ε *u2 | = $|u_1 \cdot u_2| = |u_1| + |u_2|$

(5) | $u2^{5}$ | = $5 \cdot |u_2| \Rightarrow$ 5 * 3 = 15