

• Fila (Queue)

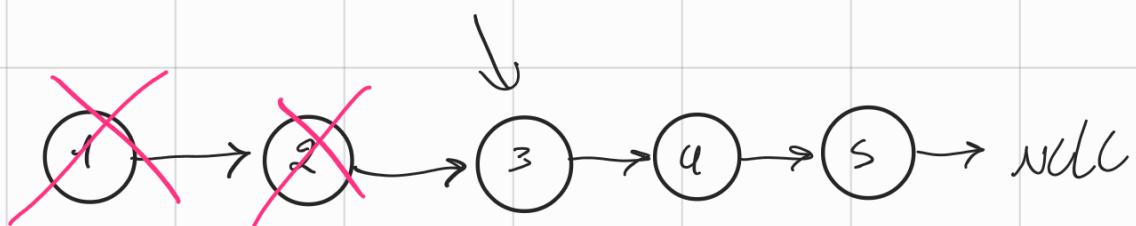
(1, 2, 3, 4, 5)

FIFO

→ empilhamento (push)

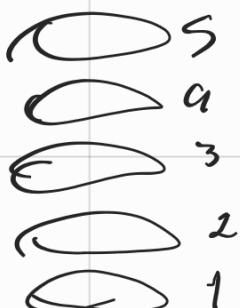
→ desempilhamento (pop)

[2, 3, 4, 5]
1

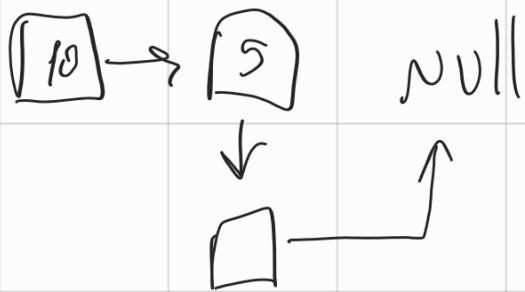


• Pilha (Stack)

LIFO

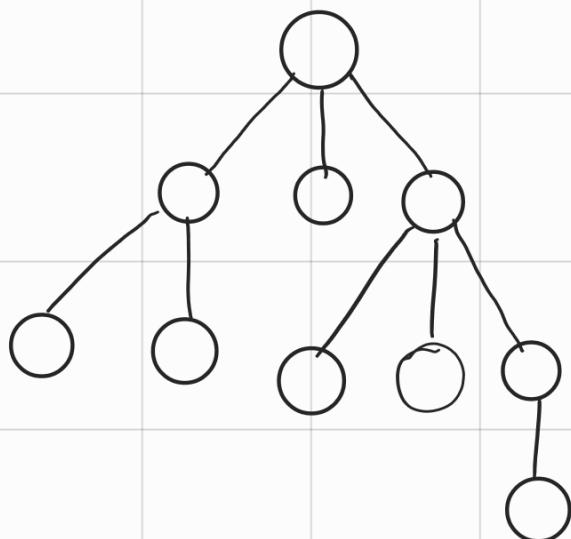


~~start~~ → ~~10~~ → ~~5~~ → null



Árvore:

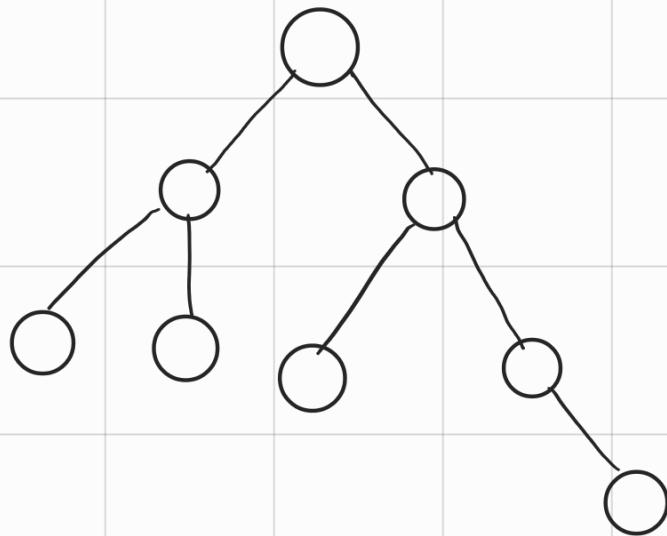
Vertices ligados por arestas, em que
há somente um único caminho entre dois nós



Árvore Binária

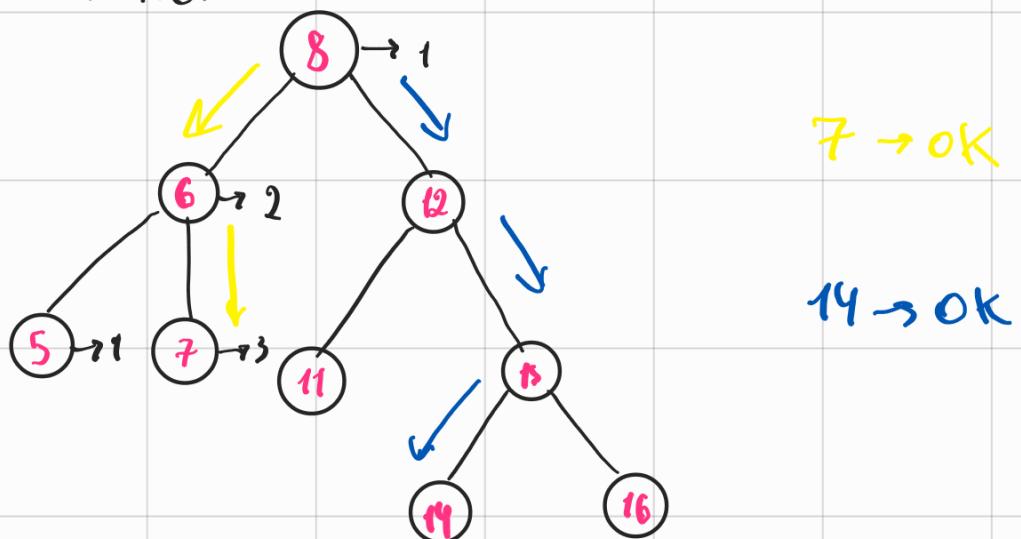
Árvore Binária:

Cada vértice pode ter até dois filhos:



Árvore Binária de Busca

O filho esquerdo é menor que seu Pai e o filho direito é maior que seu Pai:

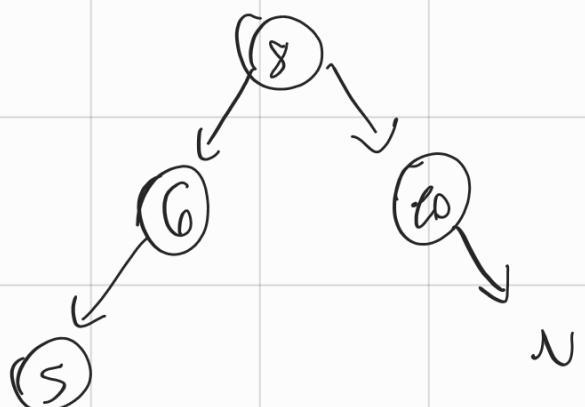
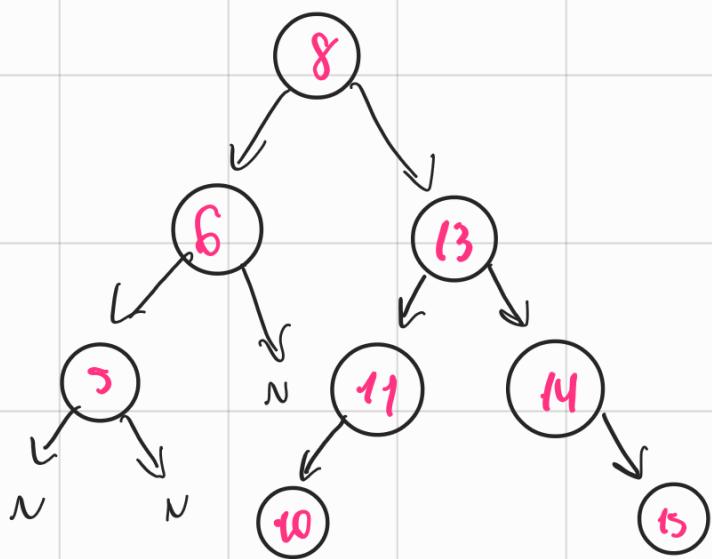


node {

node * left, * right;

}

return

$$\{1, 2, 3, 4, 5, 6\}$$


• BUSCA binaria

$[1, 2, 3, 4, 5, \boxed{6}, 7, 8, 9, 10]$



$[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]$

$8 \rightarrow \text{OK}$

$$l=0, h=9$$

$$m = (l+h)/2 = 4$$

$O(\log n)$

$$l = m + 1(s), t = q$$

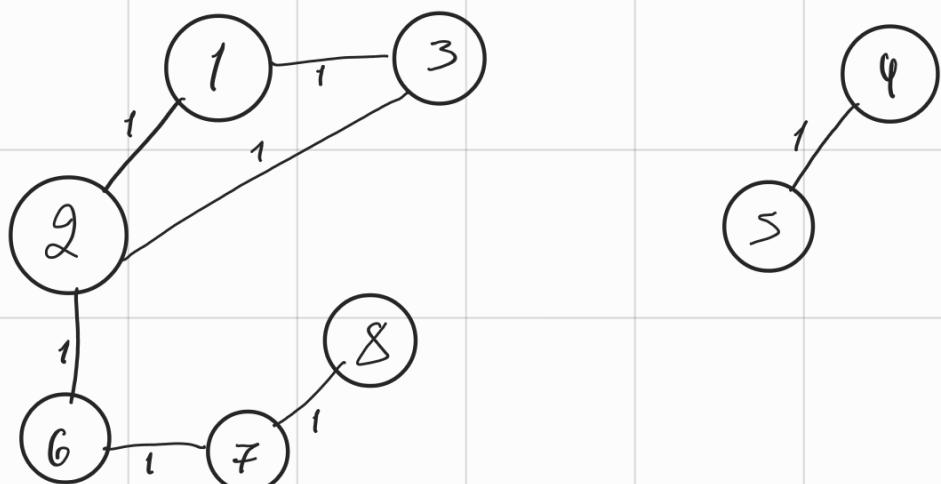
$$m = (s + q) / 2 = 7$$

return 1;

- GRAFOS

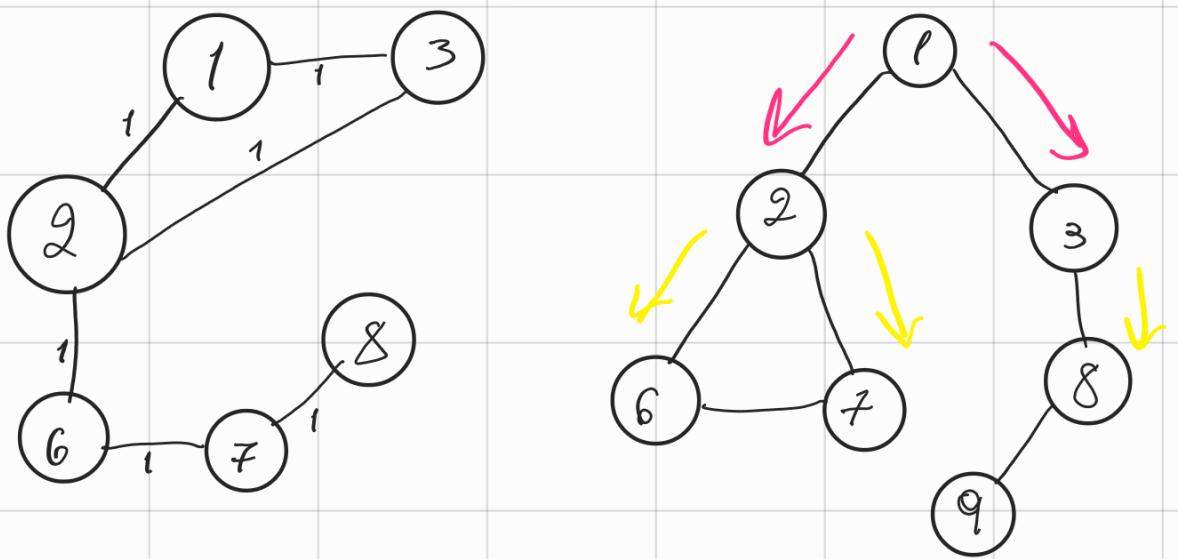
Vértices ligados por arestas, sem caminho único ou mesmo caminho definido

$$N = 5$$

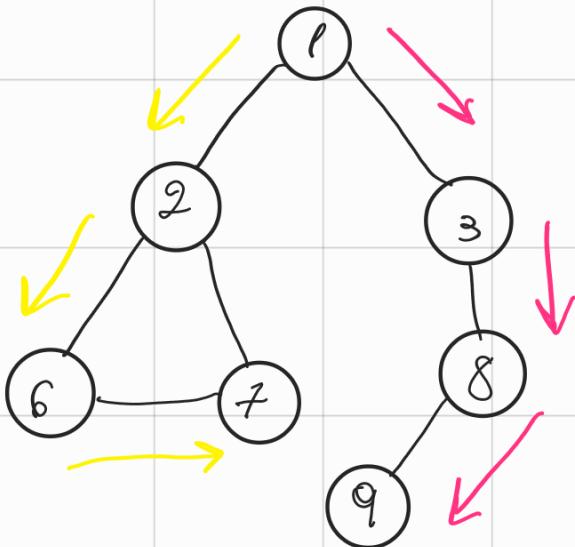


Traversias (BFS, DFS)

- Por largura (BFS)



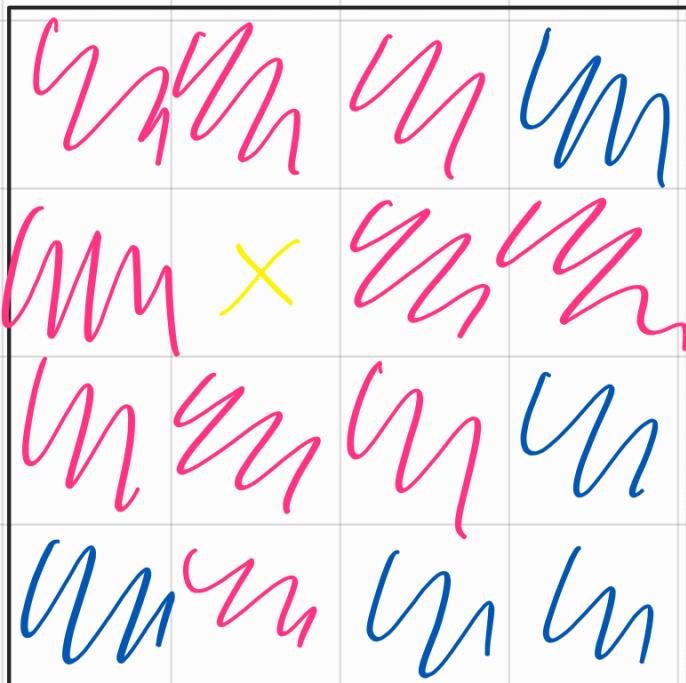
- Por profundidad (DFS)



- BFS

m	n	l	m
m	\times	l	m
m	m	l	m
m	m	l	m

• DFS

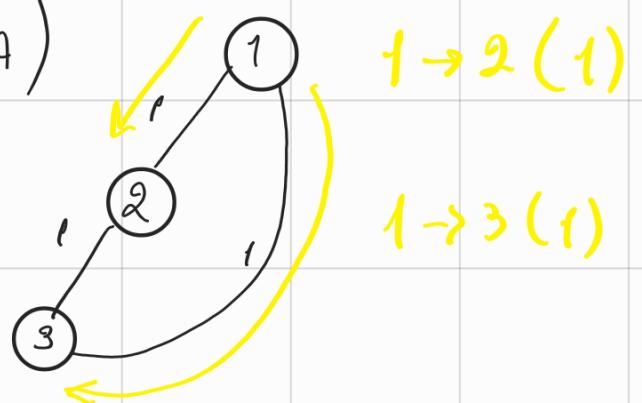


- $U \rightarrow V$; Consigo?

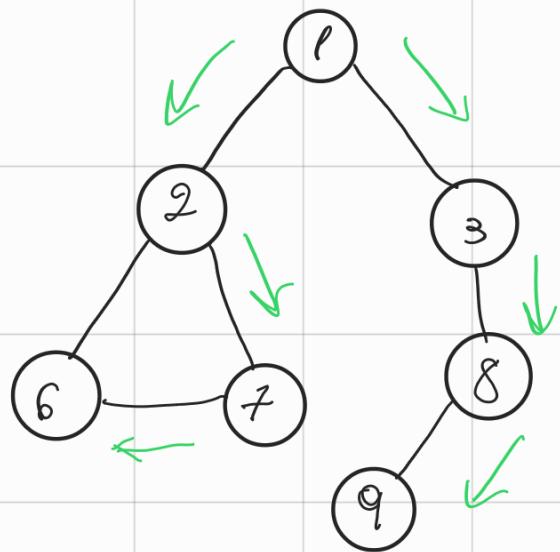
DFS (stack, pilha)

- $U \rightarrow V$; Caminho mais curto

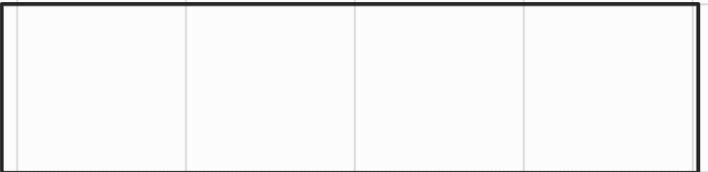
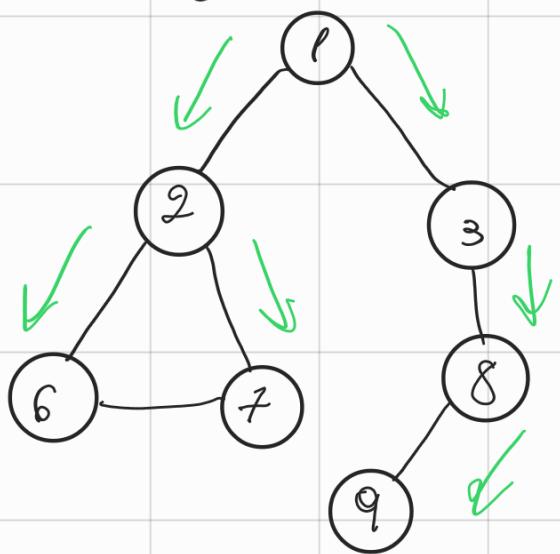
BFS (queue, fila)



• DFS



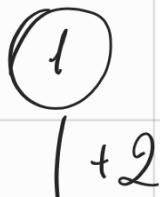
• BFS

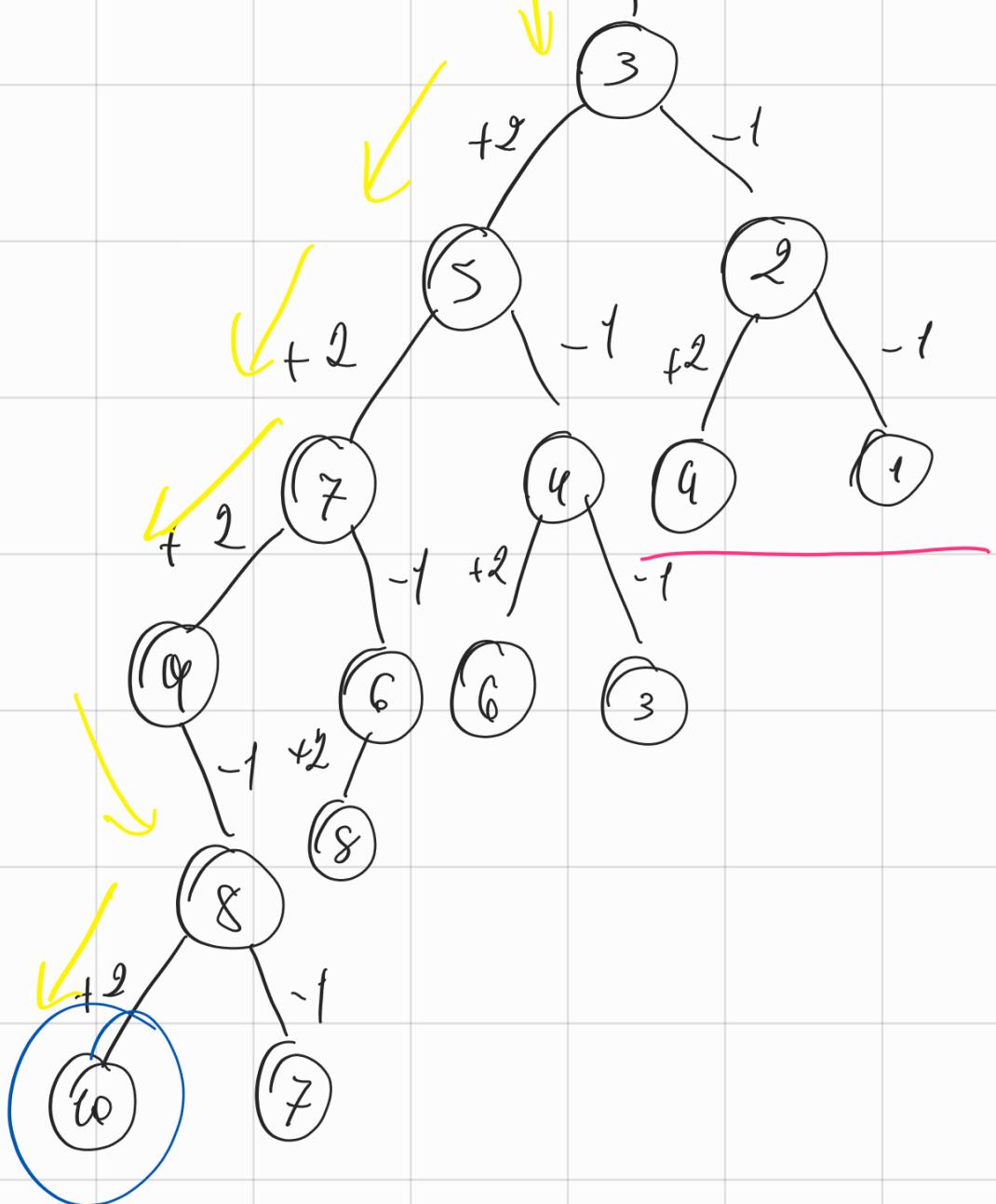


f s g u d
l o l o 2 1

l o 1 l o 2 1

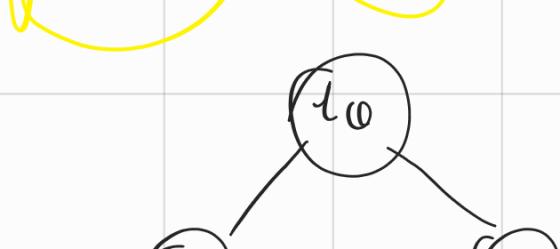
[1, 1, 1, 1, 1, 0, 1, 1, 1, 0]

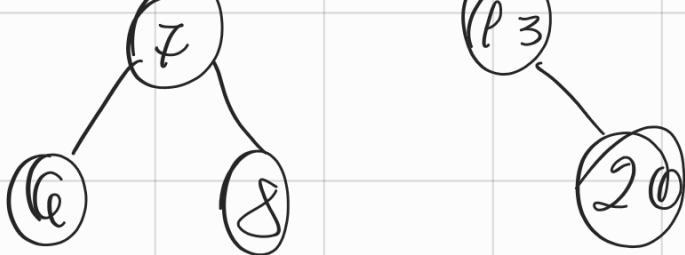




$[10, 7, 8, 13, 20, 6]$

$[6, 7, 8, 10, 13, 20]$





f() {

visitA

f(esq)

f(dit)

}

O(n)

f() {

visitA

if () f(esq)
else f(dit)

}

O(log n)