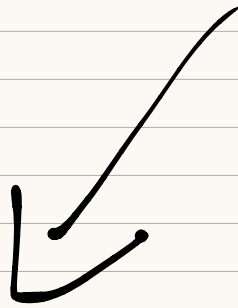


BST

[1, 2, 4, 3, 5]

ORD →



1

2

/

1

)

V

BILANCIATO

① RICERCA

② CANCELLAZIONE



PROPRIETÀ

DIST

(Mantieni la proprietà)

- BILANCIAMENTO

- AUTORE

→ ISBST

(INPUT)

→ BST,

(ptr)

BST (PUNTATORE
CHE RAPPRESENTA

BST (ROOT,
LEFT, RIGHT)

...

! = NULLPTR (vuoto)

POST ↘

RITORNA UN INODO S.T

(TAL E TAL)

- 1 se BST è tale

= 0 altrimenti

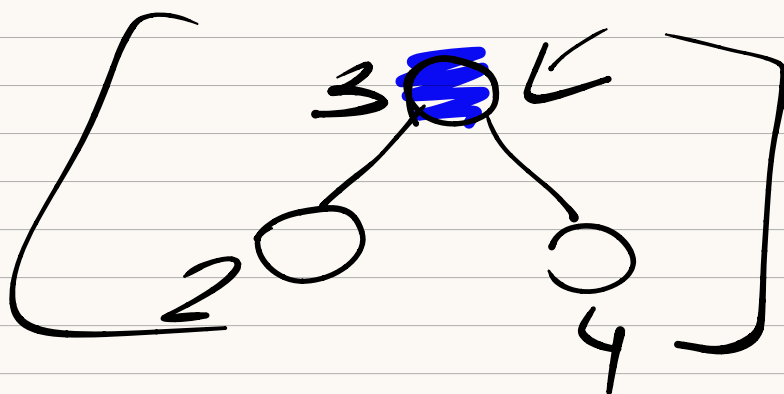
if (ptr == null) return 0

if (!bst → left ||
!bst → right)

// enter in fig 1.5
return 1

// BST root node is
BST

// 5x & 10x existance



[// TAIL RECURSION
(RECURSION IN CODE)
1 + FIB(N)

NO PER BST!

IF (ISBST (PTR → LEFT))

BST * COPY = PTR;
INT RIS;

IF (COPY → VALORE
→ =

ISBST (PTR → LEFT))

RIS = 1;

↑

ABBIAMO RICORSO

SOLO A SX,

MAGARI ASSUMIAMO

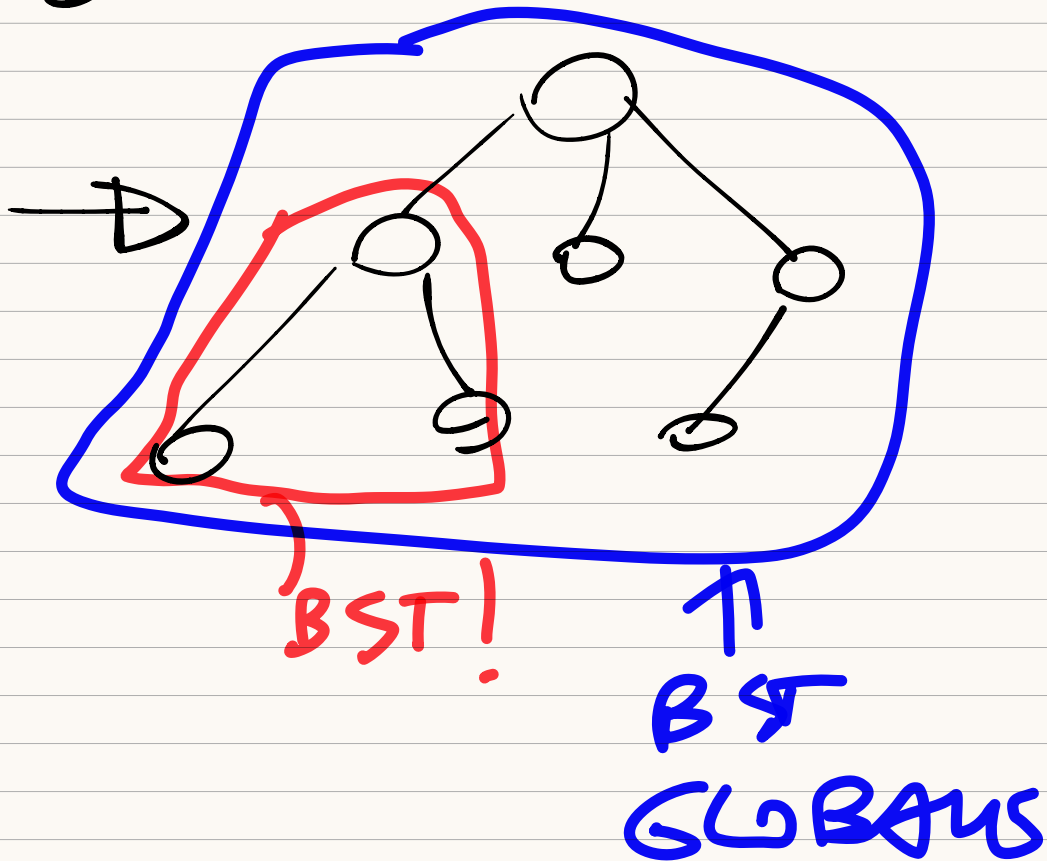
CHE NON CI SIA DX!

IF (!DX) → ADESSO
PARTE SOPRA

IF (!sx)

↓
IF (dx < root) etc.

Logique generale



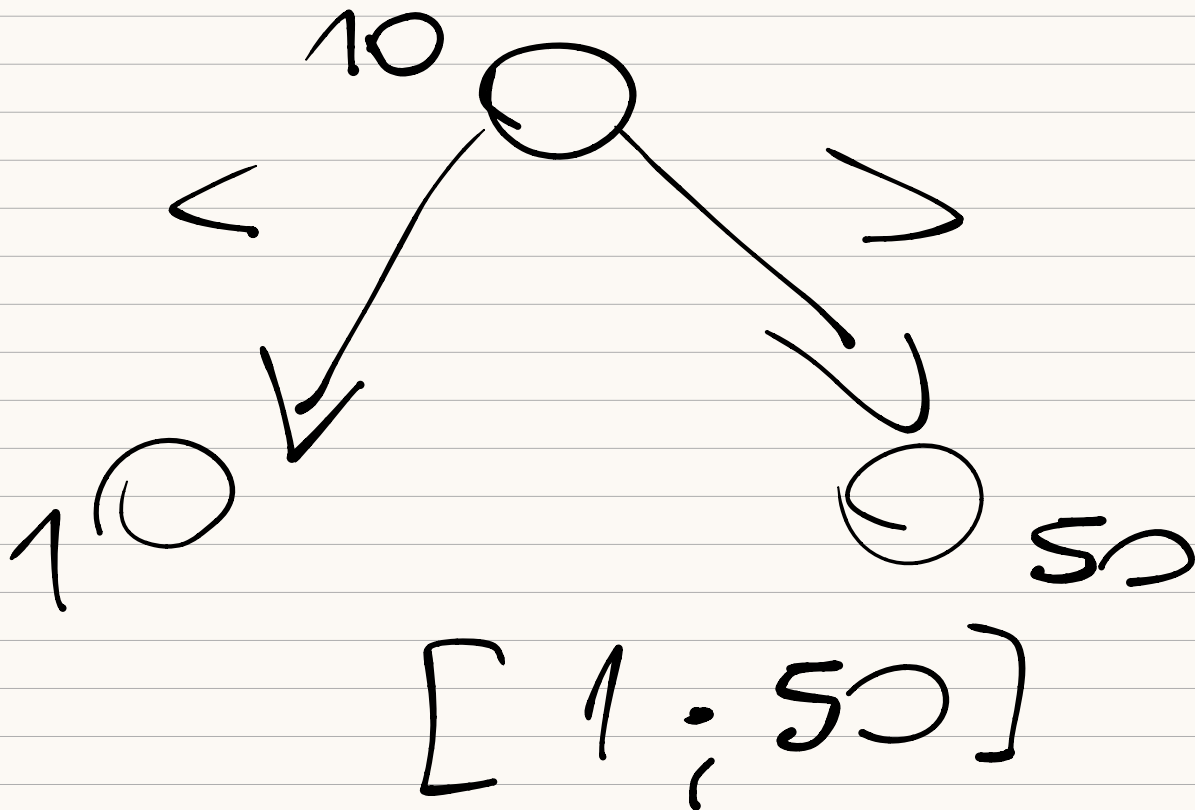
IF (RADICS)

IF (RADICS → sx)

IF (RADICS → VALORS
→ RADICS → LEFT)

$PLSD = RADICE$

→ RICORSIONE



SX (SERIE PIÙ PICCOLA)

DX (SERIE PIÙ GRANDE)



- NODI DI SX

- CONTINUA

RICORRERE A SX

SAPENDO CHE

SAZI SETTING

MIUOL

- PASSO A DX

CON MAGGIORI

// STRUTTURA

ISBST(BST)

① IF (BST == NULL)
RETURN NULL

IF (BST != NULL)

IF (BST -> LEFT)

// RICORR' A SX

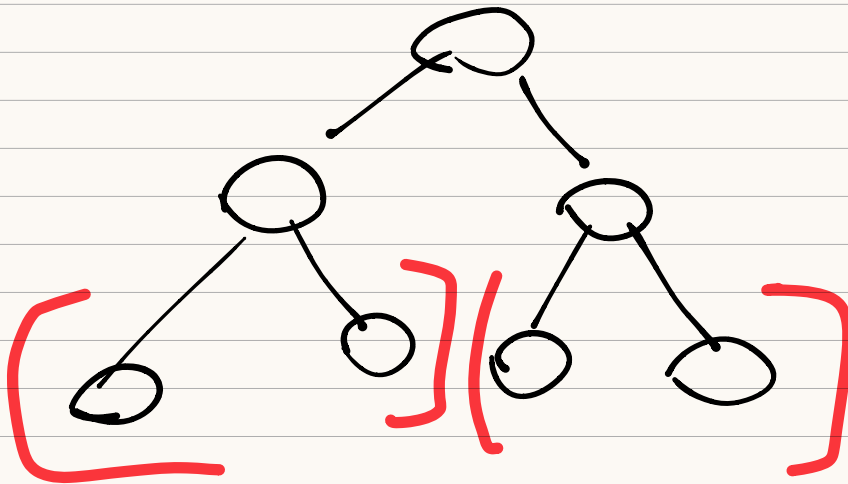
RETURN

$PTR \leq PTR \rightarrow LEFT$

QSS

$PTR > PTR \rightarrow LEFT$

RETURN 1



COUNT_LEAVES

IF (ROOT == NULL)

RETURN NULL

IF (ROOT == NULL)

COUNT_LEAVES (LEFT)

COUNT++

COUNTLEAVES (RIGHT)

COUNT++

RETURN COUNT

