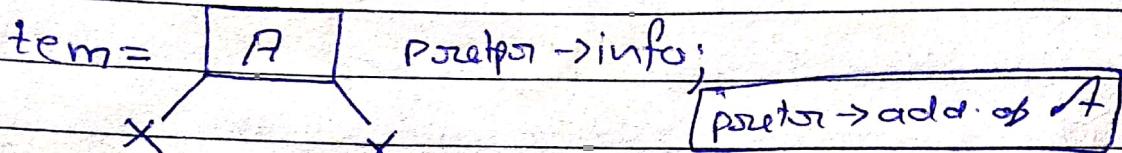


(1) → (inptor, preptror, 11)
if (num == 10) num = 11
return NULL;



if (num == 1)
return temp;

q = inptor; $\boxed{q = \text{add.G}}$

loop q=0 $\boxed{q = \text{add.G}}$

①

i=1 $\boxed{q = \text{add.D}}$

i=2 $\boxed{q = \text{add.H}}$

i=3 $\boxed{q = \text{add.B}}$

i=4 $\boxed{q = \text{add.E}}$

i=5 $\boxed{q = \text{add.I}}$

$\boxed{i=6}$ $\boxed{q = \text{add.A}}$

loop break;

temp p->lchild = construct(inptor, preptror, &s)

add. G

add. D

num = 6

Date: 11 Page No.:

(2) \rightarrow (inptor, pnextor->next, 6)
if(num == 0)
~~return NULL;~~

temp =

B

 pnextor->info
X X
if(num == 1)
~~return temp;~~
q = inptor; $[q = \text{add. } \circ G]$

loop i=0 q = add. G.
i=1 q = add. D
i=2 q = add. H
i=3 q = add. B
break;

temp \rightarrow lchild = construct(inptor, pnextor->next, 3)

(3) \rightarrow (inptor, pnextor->next, 3)
(add. G) (add. D), num = 3
if(num == 0)
~~return NULL;~~

temp =

D

 pnextor->info
X X
if(num == 1)
~~return temp;~~

q = inptor; $[q = \text{add. G}]$

loop: $i = 0$ $q = \text{add. G}$

$i = 1$ $q = \text{add. D}$

break.

$\text{temp} \rightarrow \text{lchild} = \text{construct}(\text{inptr}, \text{preptr} \rightarrow \text{next}, 1)$

(4)

($\text{inptr}, \text{preptr} \rightarrow \text{next}, 1$)

add. G

add. $\rightarrow \text{G}$

num = 1

if (num == 0)

return NULL;

temp =

~~G~~

preptr \rightarrow info

X

X

✓

if (num \neq 1)

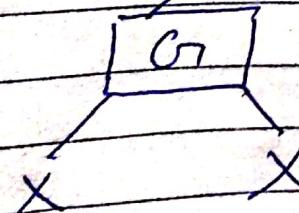
return temp

gt means this temp is return to the step.
no. 4's $\text{temp} \rightarrow \text{lchild}$

gt means

D

(3)



In continue to the step ④

Date: / / Page No.: _____

⑤

loop ($j=1, j \leq 2; j++$) num = 3

prefor = add. G

prefor = add. H

i = 1

3 - 1 - 1

3 - 2 = 1

temp \rightarrow schild = construct (q \rightarrow next, prefor, num - i - 1);

⑥ (add. H, add. H, 1)

if (num == 0)

return NULL;

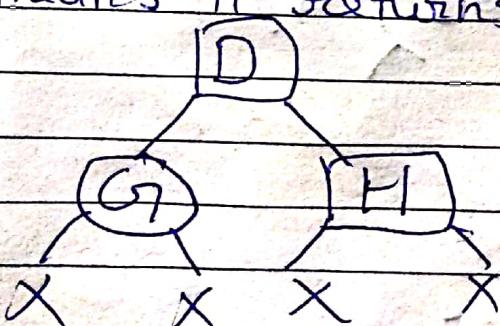
temp = H · prefor \rightarrow info

X X

if (num == 1)

return temp; ✓

It means it returns to the temp \rightarrow schild.



if returns to it by ④

schild of B.

return temp;

⑦ for ($i=1; i \leq 4; i++$) num = 6.

prefor = add. G, add. E, G, H, E

prefor = add. E

temp \rightarrow schild = (q \rightarrow next, prefor, num - i - 1)

6 - 3 - 1

6 - 4 = 2

num =

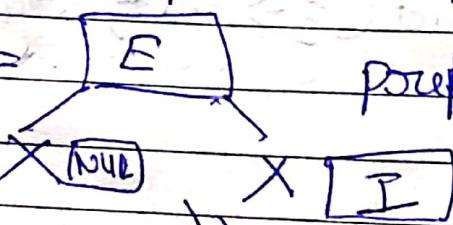
i =

Date: / / Page No.:

⑧ (add.E, add.E, 2)

if(num == 0) num = 2

return NULL;

temp =  pointer → info.

 X NULL X I

if(num == 1) return temp;

q = add.E;

E != E

loop (i=0

-temp->lchild = (add.E, add.I, 0);

⑨ if(num == 0)

return NULL;

continue

⑩ for(j=1; j<=1; j++)

pointer = add.I

temp->rchild = construct(add.I, add.I, 2-0-1) ⑪

⑪ (add. I, add. I, 1)

num = 1

Date: / / Page No. _____

if(num == 0)
return NULL;

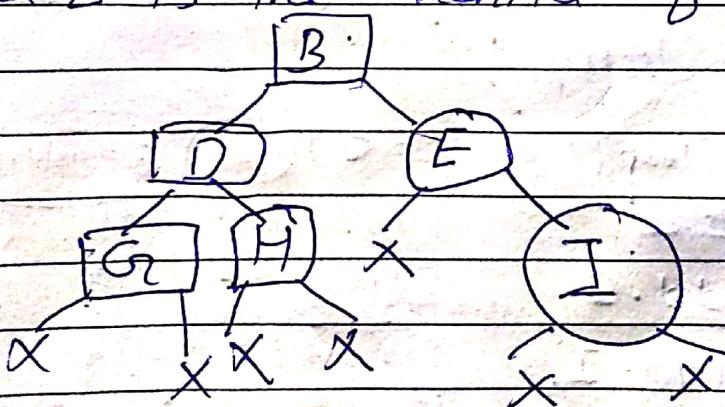
temp = I ppreptr->info
X X

if(num == 1)
return temp;

Yes, So, it will return to the lchild of E

return temp;

E & F is the lchild of B.



when it returns to the lchild of B.

all of it returns to the lchild of A

now we have $i \geq 6$ num = 11

$q = \text{add. } A$

$\text{ppreptr} = \text{add of } A$

~~for (j=1; j<=7; j++)~~

~~preceptor = B, D, G~~

Date:

Page No.:

~~preceptor = add.C~~

11-6-1
③④

~~temp->child = construct (add.B, add.C, 4)~~

~~(add.C, add.C, 4)~~

~~num = 4~~

~~if (num == 0)~~

~~return NULL;~~

~~temp = [C]~~

~~if (num != 1)~~

~~return temp;~~

~~q = add.BC~~

~~loop (i=0; C!=C)~~

~~B!=C~~

~~q = add.C~~

~~i=0, B!=C ++~~

~~i=1, E!=C ++~~

~~i=2, I!=C ++~~

~~i=3, A!=C ++~~

~~i=4, q = add.C~~

~~for (i=0; q!=C)~~

~~i=1, q = add.E~~

~~i=2, q = add.I~~

~~i=3, q = add.A~~

~~i=4, q = add.C~~

~~temp->child = construct (add.B, add.F, 4)~~

(odd E, add. F, 3)

~~null~~ num = 0

if (num == 0) /

return NULL; ✓ lchild of C

temp | F

prop1 -> info

loop (j[≥] 1, j <= 1 ; j++)

prop2 = (add. F)

X X
if (num == 1)

return temp;

temp->lchild = (add. J, add. F, 3)

num = 4 - 0 - 1

③

q = add. B

loop.

i = 0

q = add E

i = 1

q = add D

i = 2

q = add A

i = 3

q = add C

i = 4

q = add J

i = 5

q = add F

(add. J, add. F, 3)

num = 3

if (num == 0)

return NULL;

temp = F

temp->lchild = (add. B, add. D, 3) return temp;

q = add. J.

loop i = 0 j < F

i = 1

F != F

q = add. F

break;

temp->lchild = (add. J, add. J, 1)

②

$\rightarrow (\text{add}\cdot J, \text{add}\cdot J, 1)$

$\text{num} = 1$

$\text{if } (\text{num} == 0)$

~~return null;~~

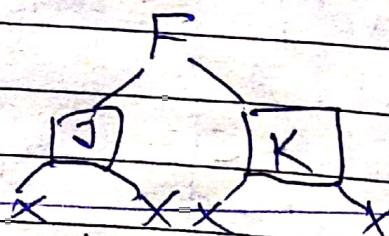
$\text{temp} = \boxed{J}$ proptor \rightarrow info.

$x \quad x$

$\text{if } (\text{num} == 1) \quad \text{Yes.} \quad \checkmark$

return temp;

So it means it will return to the .lchild
of F



$\text{for } (j=1; j \leq 2; j++)$

(3)

proptor \rightarrow add.K

$\text{num} = i - 1$

$3 - 1 - 1 = 1$ (1)

$\text{temp} \rightarrow \text{g1child} = (\text{add}\cdot K, \text{add}\cdot K, 1)$

$(\text{add}\cdot K, \text{add}\cdot K, 1)$

$\text{num} = 1$

$\text{if } (\text{num} == 0)$

~~return null;~~

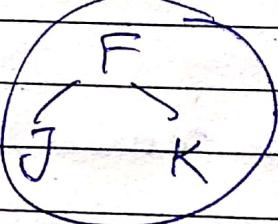
temp  = pseptor->info

Date: / / Page No.:

if (num~~s~~ == 1) Yes ✓
return temp;

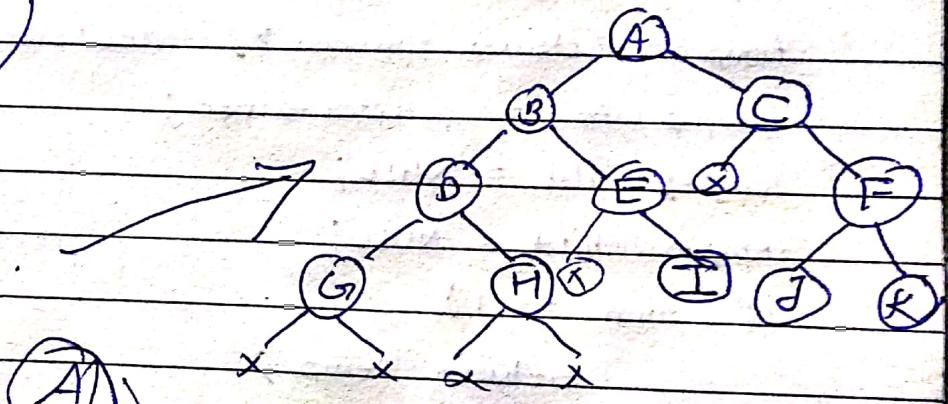
So it will return to the sibling of F

return temp.

So know  will return to F's

sibling of C.

Now will get



Traversal and Insert & Delete

Preorder : A B D G H E I C F J K

Inorder : G D H B E I A C J F K

Postorder :

