Data Structures

Linear

{arrays, linked list, stacks, Quenes}

A [o] A[1] A[2] A[3]

Non-linear

{trees, Graphs

root

G

E

A

head

1

2

3

A

leaf nodes

100

7/500

100

**8** 200

500

ऋ

10

200

top2

L

9

10

L

Stack

/7/3/-4/47

78

I

front

1 **2**

ricar

A

течни поводу -Parent

child

Siblings

- Leaf

Enternal

B

-

E

1

Root

!

Levelő

D

Level I'

Internal

Nodes

"

Level 2

f

G

-

I

Path

-

K

Level '3'

•Leaf

Nodes

~ Predecessor / Successor - Degree

Ancestor / Descendant

Sub-Tree

-

Height of node

(No. of edges in longest

(No. of

childre

-Degree of tree

(Man⋅ degree among

nodes

Depth node

१.

path from node to leaf) (Edge from Root to nay

Root =

Leaf Nodes =

Internal Nodes =

Child of G

Parent

=

of G

=

Predecessor of G =

Successor

of

G=

Ancestor

of G=

=

Descendant of G

Height of G =

Depth of G =

Level

of

G =

Degree of

Tree =

Meight of Tree =

(1)

A

B

D

Dy

L

K

Ꭰ

B

B

A

៣)

с

A

|D

XDA XEX

4

ل

XFX

Hierarchial Model

Binary

Tree & its types

Each node can have 'atmost' two

children

Children

- 0, 1, 2

*о*

(i) (ii),

(iii)\_

(iv).

(~)

(vi)

Level

о

Max. No. of

Nodes

2° = 1

1

21=2

2

2

2=4

.3

23=8

Man**.**

No. of

nodes at

any

level=21

Height of

Rost Nodu

tree = Height of

= No. of edges in longest path from

Root to the leaf

for this tree, height

Man

no

**.**

of

node

is 3

nodes in tree of height *'3*'

2°+2 +22 + 23 = 1 + 2 + 4 + 8

=

15

(2 \* -1)

Man**.** no.

of

nodes in tree

of height

'/'

=

11 11

2° + 22 + 22 +

= 2

**htl**

Min.

no.

1

-

of nodes in tree of

1

L

1

1

っち

+2

ht. 'l

= ht]

Man. ht.

=

||

4 (n+1)

when **min**. no

=n-1

Min. et = when max. no·

=

of nodes

when max. no. of nodes

[log (n+1)=1]

2

Types of Binary Tree

Full / Proper/Strict

All nodes hove 2 children,

the le except the

• No

**•**

of leaf

= No.

NL

0. of

NL

L

leaf nodes

nodes

nodes + 1

non-le of nod

・of non-te of nodes

Leaf

=2

nodes = 2 + 1

= 3

**ما**

Max. no.

*of*

nodes

=2

htl -1

лип

no

Min nt =

of nodes = 2h+1]

[log (n+1)-17

Manht &

: (-1)/2

Complete Binary Tree

1) All Levels

filled

completely

encept (the last

2) In the last

level, node

must be as

as

possible

left

Mar nodes = 2ht!\_1 Min nodes =2h

Man hat = [log (uti)=}]

Min ht = log

и

login