* Binomial Heaps and Fibonacci Heaps

Data structures presented by these 2 heaps are kno as mergeable heaps, which support the followin 5 operations :

1. Make_Heap( ) : It creates and returns a new
2. heap containing no elements.

2. Insert (H, x) : It inserts node x whose key field has already ~~being~~ been filled in, into heap H.

3. Minimum (H) : It returns a pointer to the no in heap H whose key is minimum.

4. Extract_Min (H) : It deletes the node from heap whose key is minimum, returning a pointer to th node.

5. Union (H_1, H_2) : It creates and returns a new heap that contains all the nodes of heaps H_1 and H_2. Heaps H_1 and H_2 are 'destroyed' by this operations.

In addition the data structures present in these 2 heaps also support the following 2 operatio

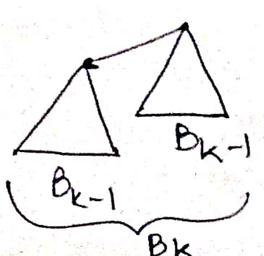
1. Decrease_key (H, x, k) : It assigns to node x within heap H. the new key value k which is assumed to b no greater than it's current key value.

2. Delete (H, x) : It deletes node x from heap H.

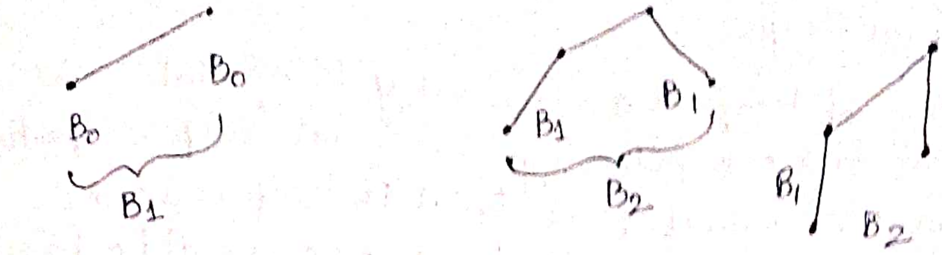
* Binomial Trees :

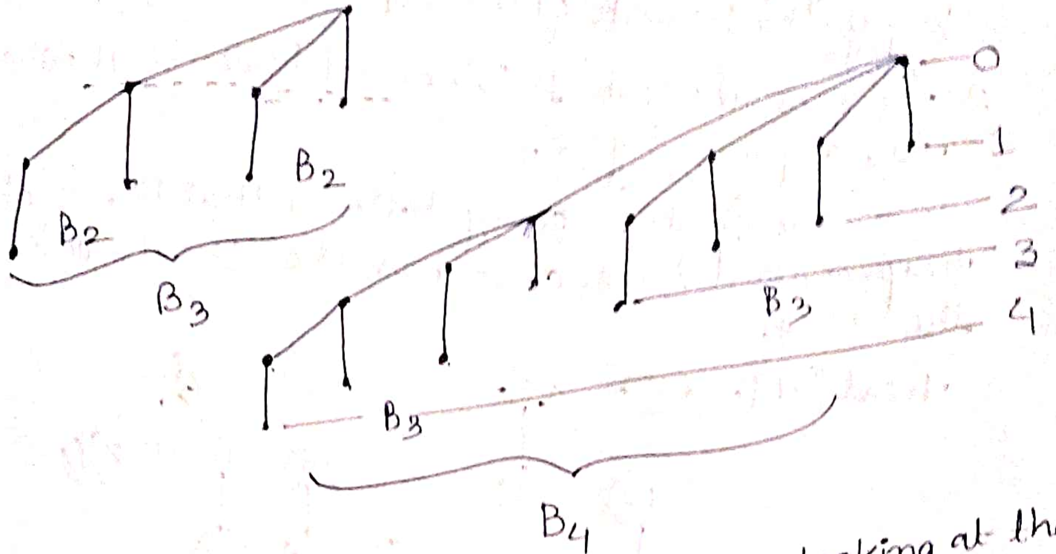
A Binomial heap is a collection of Binomial trees. The Binomial tree B_K is an ordered tree defined recursively.

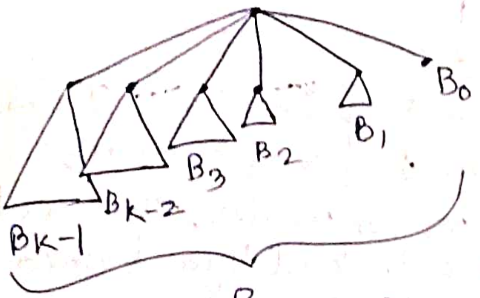
• ; the Binomial tree B_0 consists of a single node.
B_0



; The Binomial tree B_K consists of 2 Binomial trees B_{K-1} that are linked together : the root of one is the left child of the root of the other.







; Another way of looking at the binomial tree Bk.

Properties B_K of Binomial Trees:

• For the Binomial Tree Bk,

1. there are 2^k nodes, [proove by induction][$k c_i$
2. the height of the tree is k, ["]
3. There are exactly $\binom{k}{i}$ nodes at depth i, whe $0 \le i \le k$, and
4. the root has degree B_k k which is greater than that of any other nodes. moreover if the children of the root are numbered from left to right by k-1, k-2, ..., 1, 0, then child i is the root of a sub-tree Bi.

The maximum degree of an n node binomial tree is lg n. The proof is immediate by propertis 1 and 4 above. lg (Bax 2) ln (Bax e)

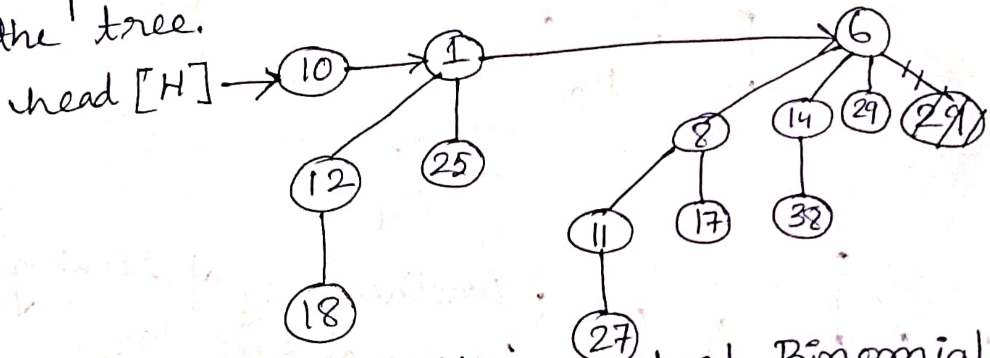
The term "Binomial Tree" comes from property 3 since the terms $\binom{k}{i}$ are the Binomial coefficients

* Binomial Heaps

A Binomial Heap H ~~a~~ is a set of Binomial Trees that satisfies the following. Binomial heap propert

1. Each Binomial tree in H is heap-ordered: The key of a node is greater than or equal to to the key of its parent. (Min Heap).
2. There is atmost 1 Binomial tree in H whose has a given degree.

The first property tells us that the root of heap-ordered tree contains the smallest key in the tree.



The Binomial heap H consist of Binomial Trees B_0, B_2 and B_3 which have 1, 4 and 8 nodes respectively and in total $n=13$ nodes. The roots of the Binomial trees are linked by a linked list in order of increasing degree.

The second property implies that an n node Binomial heap H consists of atmost of Binomial trees.

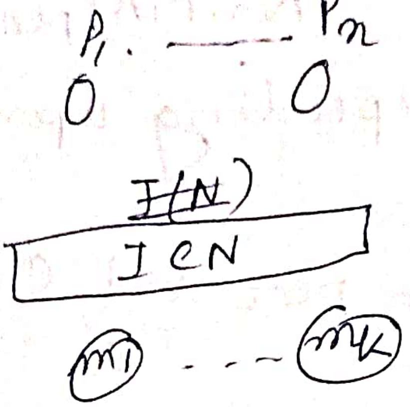
Process migration:
Light weight process - thread.

P_1 ------ P_n UMA

I(N)

ICN

m1 --- mk



Very Large Instructio
Architechture (VLI

prephase queue buffer

9/12/2

Assignment: 16

perform the operation of 'Binomial_Heap Extract_Min(H)' for a Binomial heap that comprises at least 6 Binomial trees where the minimum key ~~is~~ value is present in a node i.e. belonging to the largest ~~heap~~ tree or the next to the largest ~~heap~~ tree.

B_K consist of 2 copies of B_{K-1} So, B_K

$2^{K-1} + 2^{K-1} = 2 \cdot 2^{K-1} = 2^K.$

By induction hypothesis the max depth of B_K

B_K = max depth of $B_{K-1} + 1$

$= \cancel{(B_K-} (K-1)\cancel{)} + 1$

$= K.$

i = 1 $\quad {}^4C_1 \quad$ i = 2 $\quad {}^4C_2$

$(a+b)^1 = a+b$

$(a+b)^2 = a^2+2ab+b^2$

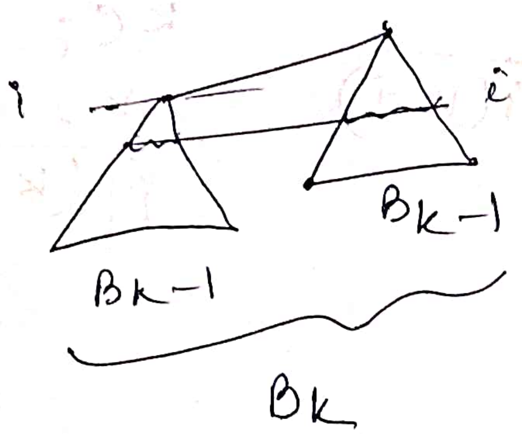
$(a+b)^3 = a^3+3a^2b+3ab^2+b^3 \; (a+b)$

$(a+b)^4 = (\cancel{(a+b)^3})^2 \cancel{= (a^2+2ab+b^2)}$

$\cancel{= a^4 + 3a^3b + 3a^2b^2 + ab^3 +}$

$= a^4 + 4a^3b + 6a^2b^2 + 4ab^3 + b^4.$

In terms of no. of nodes in a level Binomial tree is completely structured.



~~i-1 level~~ $\cancel{2}^{i-1}$

nodes in i-1 level

nodes in level $\cancel{2^{i}+2^{i-}}$

$\cancel{2^0 + 2^1 + \dots + 2^{i-1} + 2^i}$

2^{k-1} $B^4 =$

Let, $D(k,i)$ be the no. of nodes at depth i of B_k. Thus, $D(k,i) = D(k-1,i) + D(k-1,i-1)$

$= \binom{k-1}{i} + \binom{k-1}{i-1}$

${}^{n}c_i + {}^{n}c_{i-1}$

$= {}^{k-1}C_i + {}^{k-1}C_{i-1}$

$= \cancel{{}^{k-1}C_i}$ ${}^{n+1}c_i$

$= {}^{k-1+1}C_i$

$$= \frac{(k-1)!}{i!\,(k-i-1)!} + \frac{(k-1)!}{(i-1)!\,(k-i)!}$$

$$= (k-1)!\,\cancel{k}\left(\frac{1}{i(i-1)!\,(k-i-1)!} + \frac{1}{(i-1)!\,(k-i)(k-i-1)!} \right)$$

$$= \frac{(k-1)!}{(i-1)!\,(k-i-1)!} \left(\frac{1}{i} + \frac{1}{k-i} \right)$$

$$= \frac{(k-1)!}{(i-1)!\,(k-i-1)!} \; \frac{k\cancel{-i}\cancel{+i}}{i(k-i)}$$

$$= \frac{k\,(k-1)!}{(i-1)!\,(k-i-1)!\; i(k-i)}$$

$$= \frac{k!}{i!\,(k-i)!}$$

$$= {}^{k}C_{i}$$

$${}^{k}C_{i} = \frac{k!}{i!\,(k-i)!}$$