

Advanced Data Structure Batch - 5 Program ~ 9 Rutagent Ritik Rout 18M18CS151 9/12/20

Insert Cheap, value I?

Create new node with value or key as value create temporary heap

looping oner heaps until it becomes NULL:

it degree of original tree in heap

is less than degree of temporary tree in heap

Create new heap and add original tree

else create new heap and add original tree

add temporary tree to heap

if original heap has left over tree.

if temporary heap has lift over tree add all of them to new heap.

add them to new heap

If heap size < 1 return heap

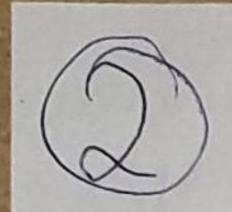
loop over new heap of heap on element remours.

else it degree tirst tree less than degree of second tree, merge.

else it degree ore some then binomial true ore some in heap.

return heap.

ongoing



get min Cheap) {
Stort from het tree in heap & check root of tree.

find min of all roots and return.

Extract min ( hap) {
 get min value by function get min ()
 start from first tree in heap.
 if tree toot is not minimum then create new heap.
 A odd tree to heap

remone minimum else from heap & con vert tree to heap.

Merge newly created heap without mini. element & heap that was created earlier.

return merged heap

De sassa