Chap - 17

Mohammed Omer Ahmed
1002201193

1) inserting n elements using
a) aggregate method
The table doubles its size when it runs out of space.
so if the original size is 1 then after insertion it doubles in size to 2 after 2 more insertions it double to size 4 etc
→ In general after k doublings the size is 2^k
→ pseudocode:
initialize table with capacity = 1
for i = 1 to n:
if table is full
new table = create new table with size 2* current size
then copy elements from old table to new table
table = newTable
insert element i into table
let, $k = \log(n+1) - 1$
Total cost $= O(n) * k$
$= O(n \log n)$
cost per insertion $= O(\log n)$
Runtime per insertion is $O(\log n)$
Total time is $O(n) * \log(n+1)$

b) accounting method

charge 2 units for each insertion
when the table doubles in size from m to $2m$
credit m units.
the credit exactly pay for the copy cost of $O(m)$
total credit is $m + 2m + 4m + \ldots$
$n/2 * m = O(n)$

→ **Pseudo code:**

initialize table with capacity = 1
for i = 1 to n
    if table is full
    new table = create newtable
        with size 2* current size.
    copy elements from old
    table to new table
        table = new table
    insert element i into table
    initialize charges = 0
    initialize credits = 0
for i = 1 to n
        charges += 2
if table doubled in size
        from m to 2m
    credits += m

total charges $= 2^* n = O(n)$

Total credits $= m + 2m + \dots$

$n/2^* m = O(n)$

cost per insertion $=$ Total$/n$

$= O(n)/n$

$= O(1)$

Runtime per insertion

$= O(1)$

Total time $= O(n)$