

## Chapter 12

- 1) Given a dynamic table that doubles its size when it needs more space. find the amortized runtime for inserting  $n$  elements.

a) use the aggregate method:

In aggregate method we consider the total cost of all insertions and calculate the average cost per insertion.

when inserting the  $i^{\text{th}}$  element, if a resize is not required then cost is  $O(1)$  because we just copy  $k$  existing elements to the new table of size  $2^i k$  ( $k$  is no. of resizes performed).

$$\text{Total cost} = O(n)k$$

$$= O(n \log n)$$

$$\text{cost per insertion} = O(\log n)$$

$$\text{Runtime per insertion} = O(\log n)$$

$$\text{Total time is } O(n) * \log(n+1).$$

b) Accounting method

The accounting method is used to analyze the amortized cost of a sequence of operation by charging some operations more than their actual cost and saving the extra credits.

Pseudocode :

from  $i = 1$  to  $n$

if table is full

new table = create new table with size  $2 * \text{current size}$

then copy elements from old table to new table  
 $\text{tabl}i$  - new table

insert element  $i$  into table

initiate charge = 0

for  $i \leftarrow 1$  to  $n$

charges + = 2

if  $\text{tabl}i$  double in size from  $m$  to  $2m$   
 $\text{credits} + = m$

Total charge =  $2^m = O(m)$

Total credits =  $m + 2m + \dots + \frac{m}{2} = O(m)$

$$\begin{aligned}\text{Amortized cost per insertion} &= \text{Total / } m \\ &= O(m/m) \\ &= O(1)\end{aligned}$$

Run times per insertion =  $O(1)$

Total time =  $O(m)$