DAA

Name: Navyasree Chagamreddy
ID: 1002197805

Chapter-17

1) inserting n elements using

a) aggregate method:

The table doubles in size when it runs out of space. So if the original size is 1, after insertion it doubles the size to 2 after 2 more insertions it doubles to size 4 etc.

In general after k doublings the size is 2^k.

pseudo code:

```
initialize table with capacity = 1
    for i = 1 to n:
        if table is full:
            newtable = create newtable with
                       size 2* current size
            copy elements then from old table
                 to newtable
            table = new table
        insert element i into table
```

Let, $k = \log(n+1) - 1$

Total cost $= O(n) k$

$= O(n \log n)$

cost per insertion $= O(\log n)$

Run time per insertion is $O(\log n)$

Total time is $O(n) * \log(n+1)$

(b) Accounting Methods:

Change 2 units for each ~~item~~ insertion when the table doubles in size from m to 2m, credit m units,

Total Credit is $m+2m+4m+ \dots$

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

Pseudo code:

```
Intialize table with capacity = 1
    for i = 1 to n:
      if table is full:
      newtable = create newtable
    with size x current size
     copy element from old table to new
     table
        table = newtable

  insert element i into table
       initialize changes = 0
       initialize credits = 0

for i = 1 to n:
        Charges += 2
      if table doubled in size
```

from m to $2m$

Credits $t = m$

Total Charges $= 2 \times n = O(n)$

Total Credits $= M + 2M + \cdots$

$= n/2 * M = O(n)$

cost per insertion $=$ total $/n$

$= O(n)/n$

$= O(1)$

Runtime per Insertion $= O(1)$

Total time $= O(n)$