Rohith Jamsadkar
1002226469

Hands On 11

1) Given a dynamic table (see section 17.4) that doubles in size when it needs more space. Find the amortized runtime for inserting n elements.

a) Use the aggregate method in the aggregate method we consider the total cost across all the insertions and calculate the average (amortized) cost per insertion.

When inserting the i^{th} element, if a resize operation is not needed the existing happens cost(i) as it involves copying the existing elements to the new table of size 2^k (k is number of resizes performed).

Total Cost = $O(n)$ k $\Rightarrow$ $O(n \log n)$

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

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

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

b) Accounting method:- In the accounting method, we assign each insertion a higher "amortized" cost the store "credits" that pay for future resizing costs.

Pseudo code:-

```
for i=1 to n
    If table is full
        newtable = create new table
        with size then copy elements
        from old table to new table
        table = new table
    insert element i into table
    initial charge = 0.
```

for i = 1 to n

charges += 2

If table doubled in size from into 2m

credits += m

total charges = 2 * n = O(n)

total ~~cost~~ credits = m + 2m - - - - - $n/2 \times m$ = O(n)

Amortized cost pa insertion

= total / n

= O(n/n)

= O(1)

Runtime per Insertion O(1)

Total time O(n).