Ramaneek Gill 1000005754

CSC263 Problem Set 5

An important thing to observe in this question is that DELETE will never reduce the size of the array. This means that DELETE will always run at O(1) time.

Amortized complexity = WCSC / m

-Worst case run time for DELETE is O(1)

-unknown for APPEND.

Let's compute APPEND's worst case run time by using the accounting method:

DELETE's charge and cost are 1. It needs no credits.

APPEND's actual cost is 1 but we will charge 2+10k. //k is the amount of times a new array is dynamically created

Credit Invariant/Breakdown:

1 is used right away to append

1 credit is used to copy itself over into the new array in the future

1 credit (from the 10k credits part) is used to help copy over previous elements which don't have credits left to the new array

-this one might seem confusing, let n refer to the size of the array

Using induction:

Invariant:

each element after being appended has 1+10*k credits where k is the amount of times array has been dynamically increased

if no more spaces left in array create a new array with 10 more spaces and copy all previous elements over

base case:

k = 0

//array hasn't been dynamically increased yet, so the first 10 elements have 1 extra credit each for the next move over

Second base case: //first dynamic increase occurs

k = k+1

elements use 1 credit to copy over

all elements now have 0 credits //this is right before the 11th element is appended

Induction Step:

suppose invariant holds and we need to dynamically increase array size by 10

the next 10 elements have 1 credit stored for themselves for the future move

AND

10*k credits stored for all the previous elements to move over (since they have none)

-this process repeats every time the array size needs to be increased

-the last 10 elements in the array always give a credit to previous elements to make the move

So total cost <= total charge = 2 + 10*k.

So for n insertions the amortised cost is O(n). For n deletions the amortised cost is O(1). So for any sequence of insertions
and deletions the amortised cost will be O(n).