CS-261: Assignment 2 Written Analysis

Khrystian Clark

02/02/2021

Question1:

a) The calculation can be shown in the following table(when N = 40):

Append	**1**	**2**	**3**	**4**	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Write Cost	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Copy Cost				3			**6**						12							
Total Cost	1	2	3	7	8	9	**16**	**17**	**18**	**19**	**20**	21	34	35	36	37	38	39	40	41

Append	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Write Cost	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Copy Cost					24															
Total Cost	42	43	44	45	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85

b) As N (i.e., the number of appends) grows large, under this strategy for resizing, the average big-O complexity for an append can be derived using the following calculation:

Mathematically, total cost = total write cost (S_w) + total copy cost (S_N) .

85 = 40 + 45

The total write cost, $S_w = N$.

40 = N

Recall (from the table you prepared in part (a)) the copy cost is a sequence as follows:

3, 6, 12, … (a geometric sequence)

3,6,12,24

Now, what is the cost of the last resize (k^{th} term of the sequence)?

24

The k^{th} term is derived from the following equation, and you will assume the sequence contains k terms:

$a_k = a_1 r^{k-1}$ (where a_1 is the first term and r is the ratio). In this instance:

$a_k =$ (**you have to calculate it**)

(2^(4-1))(3)= 24

Now, what is the sum of k terms of the above sequence? You can derive that from the following formula:

$$S_k = a_1\left(\frac{1-r^k}{1-r}\right)$$

CS-261: Assignment 2 Written Analysis

In this instance:
$S_k =$ (**you have to calculate it**)
$3((1\text{-}16)/(1\text{-}2)) = 45$

The above equation should work for all integers, k, such that $k \geq 1$. In this instance, k represents the resize term number.
However, we are concerned with appends, so you must express the sum of the sequence in terms of N appends, or in other words, you must derive k resize operations given N appends.

You already have determined the k^{th} resize cost (a_k) before.
And, the k^{th} resize cost (a_k) is also equal to $N/2$ because the array capacity will double to achieve a size of N and there would be half that number of existing elements to copy to the new array. Putting the two together, **you will find out the value of k in terms of N**.

N=40 , 40/2=20

Now, use the k and sum equation above (S_k) to derive the total copy cost of the given N appends:
$S_N =$ (**you have to calculate it**)
$=45$

Recall that total cost = total write cost (S_W) + total copy cost (S_N) .
As we have conducted N appends, the average complexity can be derived from:

$$\frac{(S_W + S_N)}{N} = (\textbf{you have to calculate it}) = 2.125$$

This gives us an amortized complexity of O(**you have to determine it**).
O(1) because the avg cost of each append is still 1

Question2:

a) The calculation can be shown in the following table(when N = 40):

Append	**1**	**2**	**3**	**4**	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Write Cost	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Copy Cost				3		5		**7**		**9**		11		13		15		17		19
Total Cost	1	2	3	7	8	10	**11**	**19**	**20**	**30**	**31**	42	43	57	58	74	75	93	94	114

Append	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Write Cost	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Copy Cost		21		23		25		27		29		31		33		35		37		39
Total Cost	115	137	138	162	163	189	190	218	219	249	250	282	283	317	318	356	357	395	396	439

b) As N (i.e., the number of appends) grows large, under this strategy for resizing, the average big-O complexity for an append can be derived using the following calculation:

Mathematically, total cost = total write cost (S_w) + total copy cost (S_N) .
439 = 40 + 399

The total write cost, $S_w = N$.
40

Recall (from the table you prepared in part (a)) the copy cost is a sequence as follows:
3, 5, 7, 9, 11, … (an arithmetic sequence)
Now, what is the cost of the last resize (k^{th} term of the sequence)?
3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39

The k^{th} term is derived from the following equation, and you will assume the sequence contains k terms:
$a_k = a_1 + d(k-1)$ where a_1 is the first term and d is the difference. In this instance:

$a_k =$(**you have to calculate it**)

$3 + 2(19\text{-}1) = 39$

What is the sum of k terms of the sequence? You can derive that from the following equation:

$$S_k = k\left(\frac{a_1+a_k}{2}\right)$$

$399 = 19((3+39)/2)$

In this instance:

$S_k =$ (**you have to calculate it**)
399

The above equation should work for all integers, k, such that $k \geq 1$. In this instance, k represents the resize term number.

However, we are concerned with appends, so you must express the sum of the sequence in terms of N appends, or in other words, you must derive k resize operations given N appends.

You already have determined the k^{th} resize cost (a_k) before. Now you determine the k^{th} resize cost (a_k) in terms of N (For example, in part (a) when you had $N = 6$, the last resize cost was 5 (holds for all N, $N > 3$)). You can consider either of the cases (i.e., N is odd or N is even). **You will find out the value of k in terms of N.**

N=40, there are 19 k's, first term is 3

Now, use the k and sum equation above (S_k) to derive the total copy cost of the given N appends:

$S_N =$ (**you have to calculate it**)

Recall that total cost = total write cost (S_w) + total copy cost (S_N) .
As we have conducted N appends, the average complexity can be derived from:

$$\frac{(S_w + S_N)}{N} = (\textbf{you have to calculate it})$$

This gives us an amortized complexity of O(**you have to determine it**).

O(n) because the average cost of each append ~11, where the frequency of making a new array is about 50%