CS222 - Algorithm Design

Dr. Arpita Korwar

Assignment – 1 : Sorting

Authors :

- Sanjay Marreddi 1904119
- Prakhar Mathur 1906328
- Rishabh Tripathi 1904129

1. Merge sort :

The values obtained using our code are as follows:

i	2^i	T(2^i)	i*2^i	c_i
1	2	2	2	1
2	4	3	8	0.375
3	8	5	24	0.20833
4	16	8	64	0.125
5	32	20	160	0.125
6	64	44	384	0.11458
7	128	81	896	0.0904
8	256	200	2048	0.09766
9	512	353	4608	0.07661
10	1024	711	10240	0.06943
11	2048	1401	22528	0.06219
12	4096	3102	49152	0.06311
13	8192	6806	106496	0.06391
14	16384	13400	229376	0.05842
15	32768	24593	491520	0.05004
16	65536	44224	1048576	0.04218
17	131072	56544	2228224	0.02538

mergesort_output_1906328_190411

Observations about the fraction $C_i = T(n)/nlogn$

- The Expected value of C_i is 0.070271 ms.
- The Variance of C_i is 0.000589 ms^2.
- Note that the above values of Expectation and Variance keep changing due to the randomness, but order will be same.
- The values of the C_i**'s** is becoming smaller as the value of **i** increases.
- The ci values are essentially a comparison of time taken and the idealistic $nlogn$ time complexity.
- So, the reducing value of C_i**'s** signify that our algorithm is working fine.

Complexity Analysis of our ***Merge Sort*** *:*

Time Complexity :

Best Case : O(**nlogn**)

Average Case : O(**nlogn**)

Worst Case : O(**nlogn**)

Space Complexity : O(**n**)

2. Quick sort :

The values obtained using our code are as follows:

i	2^i	T(2^i)	i*2^i	c_i
1	2	15	2	7.5
2	4	10	8	1.25
3	8	11	24	0.45833
4	16	14	64	0.21875
5	32	19	160	0.11875
6	64	36	384	0.09375
7	128	60	896	0.06696
8	256	133	2048	0.06494
9	512	311	4608	0.06749
10	1024	511	10240	0.0499
11	2048	1118	22528	0.04963
12	4096	2491	49152	0.05068
13	8192	5903	106496	0.05543
14	16384	12409	229376	0.0541
15	32768	26411	491520	0.05373
16	65536	58557	1048576	0.05584
17	131072	99169	2228224	0.04451

quicksort_output_1906328_190411

Observations about the fraction $C_i = T(n)/nlogn$

- The Expected value of C_i is 0.060093 ms.
- The Variance of C_i is 0.000394 ms^2.
- Note that the above values of Expectation and Variance keep changing due to the randomness, but order will be same.
- The values of the C_i**'s** is becoming smaller as the value of **i** increases.
- The ci values are essentially a comparison of time taken and the idealistic $nlogn$ time complexity.
- So, the reducing value of C_i**'s** signify that our algorithm is working fine.

Complexity Analysis of our ***Quick Sort*** *:*

Time Complexity :

Best Case : O(**nlogn**)

Average Case : O(**nlogn**)

Worst Case : O(**nlogn**) *{ Though it is O(n^2) in general, we reduced by using Median-of-3 method for randomly picking Pivot}*

Space Complexity : O(**logn**) *{Due to function call stack }*

----- The End -----