CS345 Special Assignment Report

Raktim Mitra(150562)
Pranjal Prasoon(150508)

13 October 2017

Running Time of Brute force algorithm:

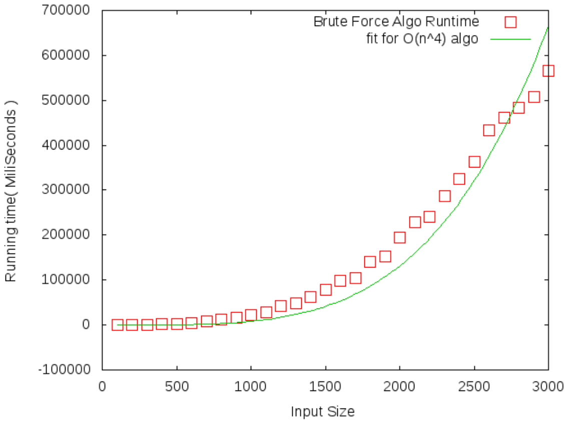

Figure 1: Runtime vs Input Size plot for the brute force algorithm with $O(n^4)$ curve fit.

Inference: As expected the running time grows very fast with increase in the input size. And we are also able to fit a cn^4 curve showing that it is following its expected time complexity of $O(n^4)$.

Running Time of Randomised Incremental algorithm:

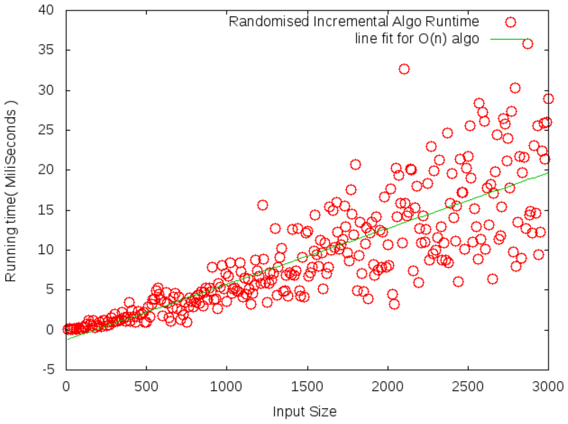

Figure 2: Runtime vs Input Size plot for Randomised Incremental algorithm with linear fit.

Inference: This plot, although is linear in general but as expected has significant devations from the line fit. Which is as expected since although its expected runtime is $O(n)$ but it can deviate from that because of its randomised nature depending on the order the points are processed by the algorithm.

Combined plot of the two Running time plots:

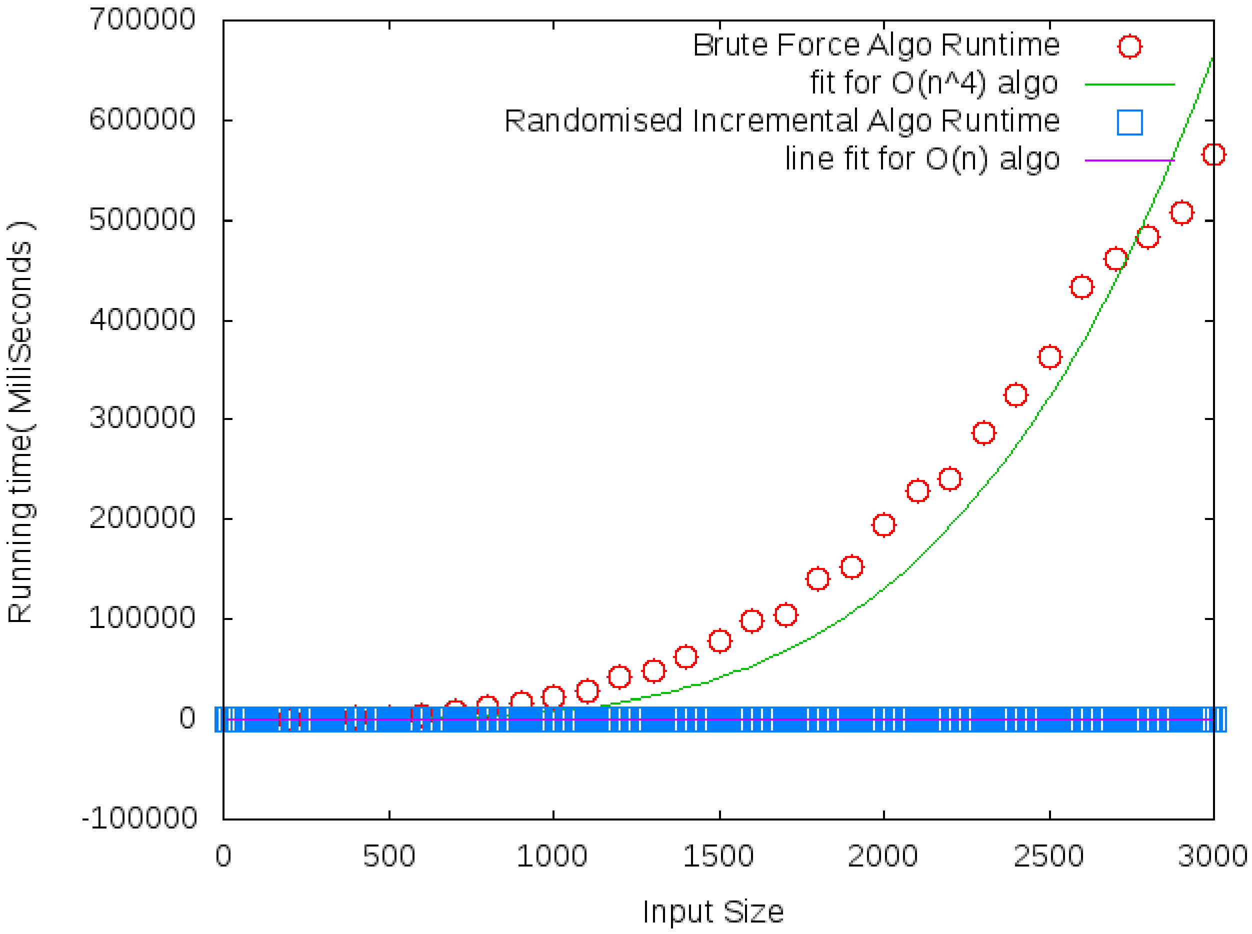

Figure 3: Combined plot of the two Running time plots .

Inference: We can see the huge difference of efficiency clearly in this plot as relative to the fast increase of the brute force running time the randomised algorithm runtime barely increases at all.

Note: We have made a recursive implementation of the algorithm too. Looks like it is a bit slower than the loop(iterative) implementations because of function calling and function stack manipulations. Nevertheless its running time remains $O(n) = cn$ with a larger c, as can be seen in the plots shown below.

Running Time of Randomised Incremental algorithm(Recursive Implementation):

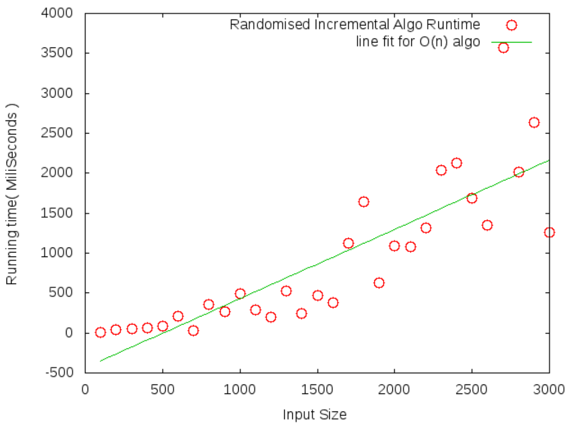

Figure 4: Runtime vs Input Size plot for Randomised Incremental algorithm with linear fit.

Inference: This plot, although is linear in general but as expected has significant devations from the line fit. Which is as expected since although its expected runtime is $O(n)$ but it can deviate from that because of its randomised nature depending on the order the points are processed by the algorithm.

Combined plot of the Brute Force and O(n) Algo Recursive Implementation Running time plots:

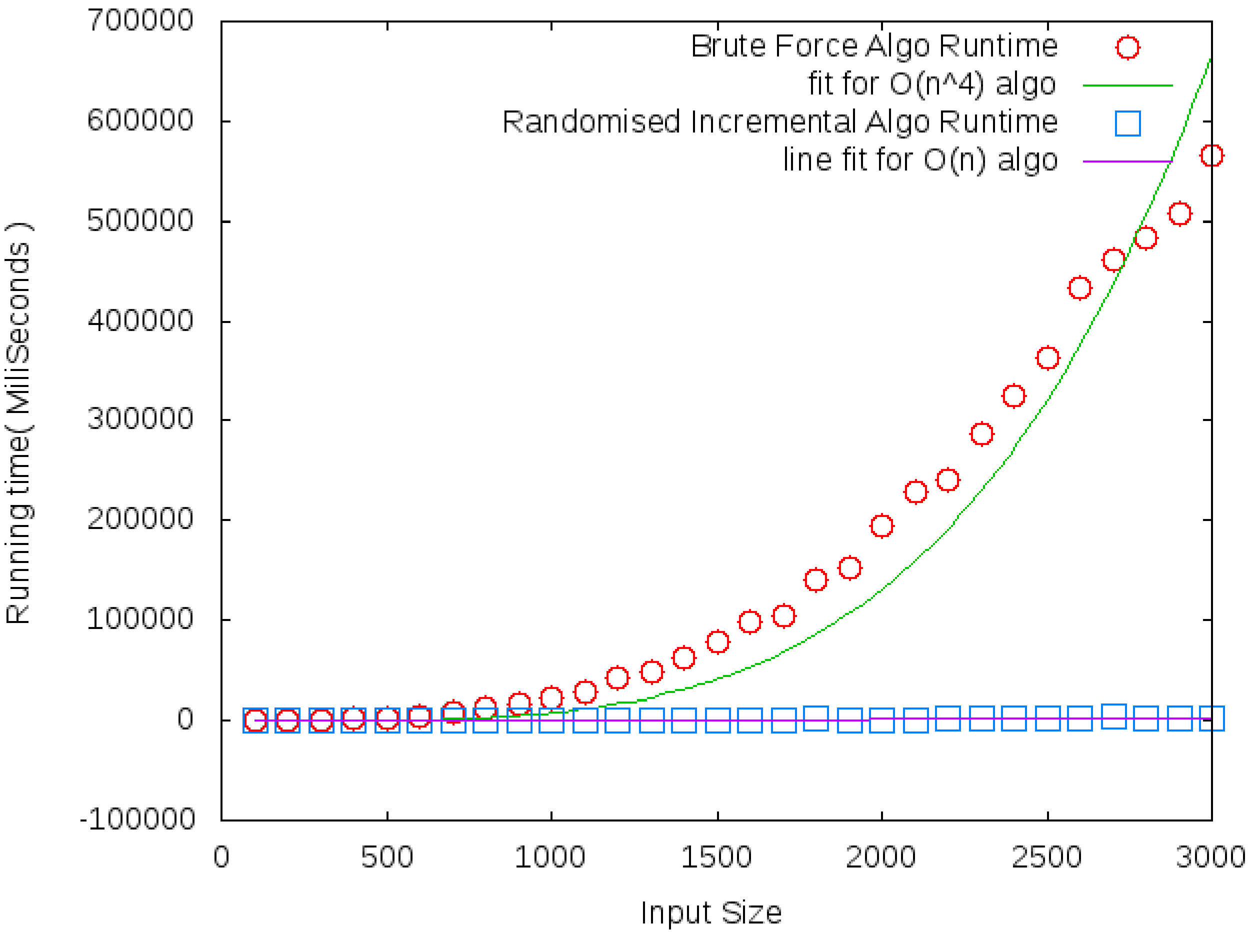

Figure 5: Combined plot of the two Running time plots .

Empirically found distribution of Randomised Algo:

For Iterative Implementaion:

We ran our program with 10000 input size , 1000 times and plotted an histogram of the running times found for the iterative implementation.

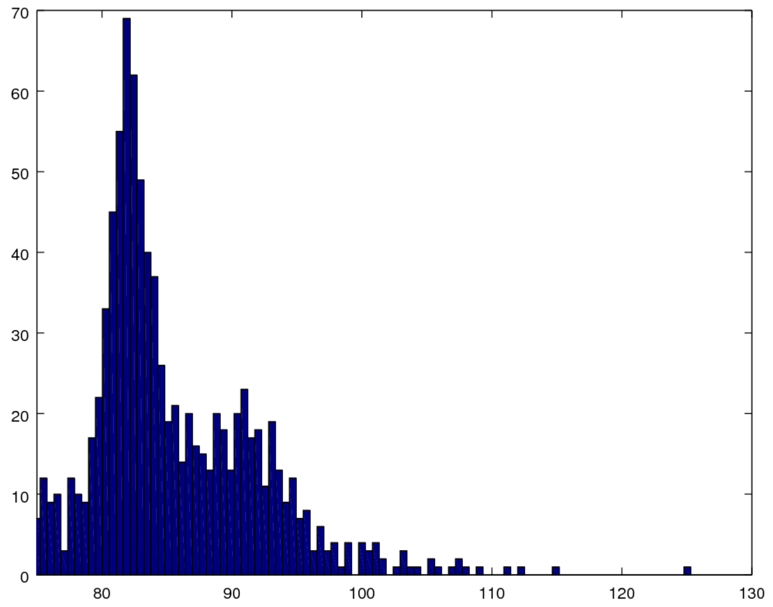

Figure 6: Histogram showing distribution of runtimes of the randommised algorithm for input size 1000 for Iterative Implementation.

Inference:

As shown, this looks like a normal distribution which is skewed to the left. Which is reasonable since although the expected(mean) runtime is O(n), it can shoot up to much higher values in cases.

Mean : 84.445 miliseconds Standard Deviation : 7.0187 miliseconds 162 out of 1000 runs gave a higher runtime than mean + SD. Which gives a 16.2% significant deviation. Coefficient of variability(SD/mean) is 0.083.

For Recursive Implementation:

We ran our program with 1000 input size , 1000 times and plotted an histogram of the running times found for the iterative implementation.

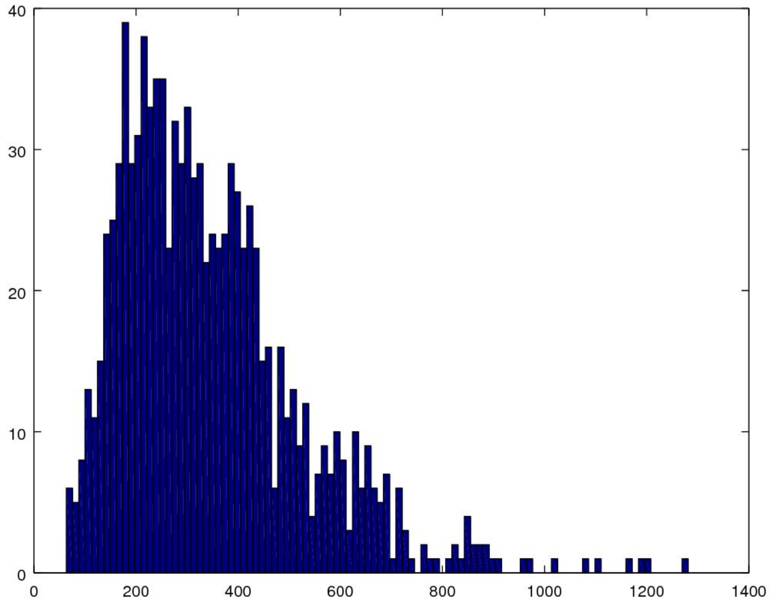

Figure 7: Histogram showing distribution of runtimes of the randommised algorithm for input size 1000 for Iterative Implementation.

Inference:

As shown, this looks like a normal distribution which is skewed to the left. Which is reasonable since although the expected(mean) runtime is O(n), it can shoot up to much higher values in cases.

Mean : 345.68 miliseconds Standard Deviation : 179.32 miliseconds 144 out of 1000 runs gave a higher runtime than mean + SD. Which gives a 14.4% significant deviation. Coefficient of variability(SD/mean) is 0.518.