ESO207 Assignment 1

Prof. Surender Baswana

7th August 2022

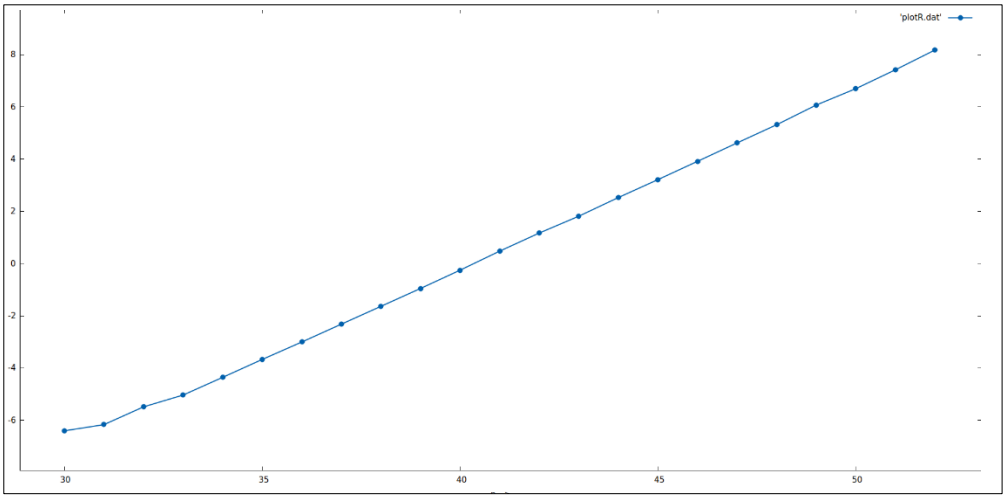
Divyanshu Gangwar (190316)
Atharva Umbarkar (190923)

1.

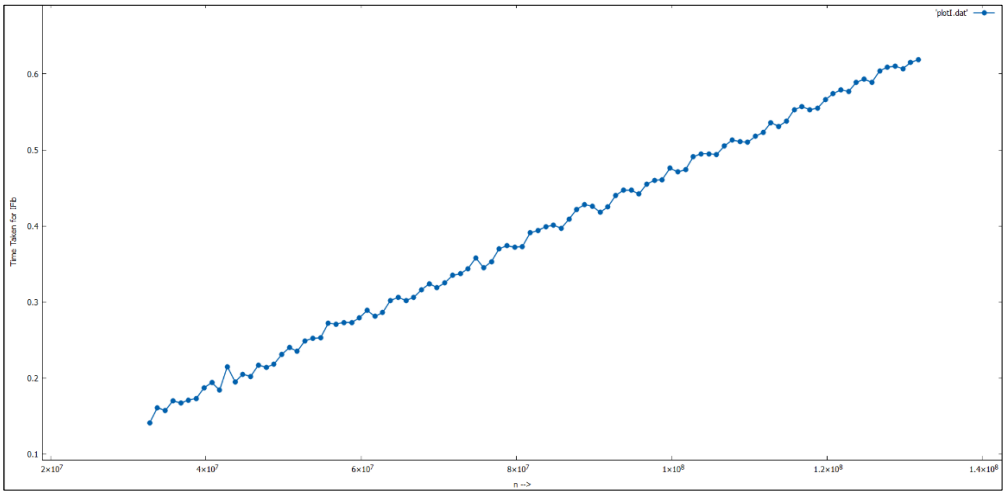
Time	0.001	0.1	1	5	60	600
RFib	26	36	41	44	49	54
IFib	≈ $2x10^5$	≈ $2x10^7$	≈ $2x10^8$	≈ 10^9	≈ $2x10^{10}$	≈ $2x10^{11}$
CleverFib	>10^{18}	>10^{18}	>10^{18}	>10^{18}	`>10^{18}	>10^{18}

2.

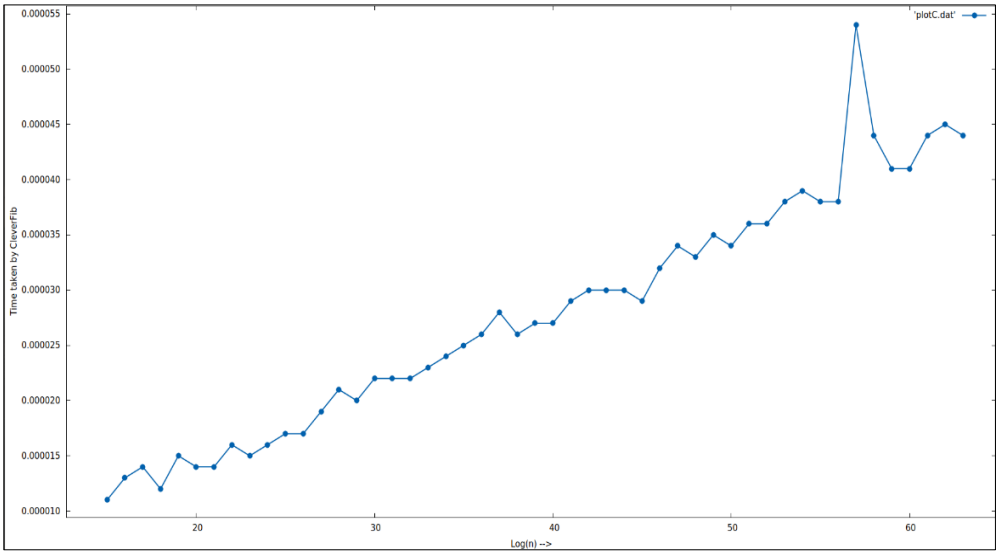
I. Log_2(Time taken by RFib) as a function of n



II. Time taken by IFib as a function of n



III. Time taken by CleverFib as a function of $Log_2(n)$



a. Time taken by RFib increases exponential with n (as per the Word RAM model) and hence Log_2(Time taken) varies linear with n.
Time taken by IFib is a linear function of n (as per Word RAM model) and hence the linear plot.
Time taken by CleverFib varies as Log(n) (as per Word RAM model) and hence the linear plot.

b. Slope of RFib line = 0.684
Slope of IFib line = 5.880×10^{-9}
Slope of CleverFib line = 3.582×10^{-7}
These slopes are different for each algorithm because each one of them performs different number/nature of operations in each step of recursion / loop (as the case may be).

c.

i. These factors very minutely influence the running time of CleverFib. The algorithm is logarithmic in 'n' and even for largest 'long long int' value in C, it finishes in practically no time (around 10^{-5} sec). These greater number of operations in CleverFib do increase the running time by some factor but it remains of the order of microseconds because of Logarithmic nature.

ii. When it comes to relative speed of CleverFib w.r.t RFib or IFib, there is no influence. RFib goes to about 10 sec for an input as small as 45 and IFib starts taking 10s around 'int' max value. But for CleverFib, even if you scale 1 microsecond by 100, it still runs within 1 ms for whole range.

3. Running time of an Algorithm – No
Comparing a pair of Algorithms - Yes

4. a) This assignment was an eye-opener for me. I found it immensely useful.
   I was able to appreciate the difference between ideal World RAM model and the intricacies in calculating algorithm time in real life.
   An Observation: Running time of CleverFib turns out to be 0.0 s (using the <time.h> library of C) while running the code on Windows. However, when I switch to Ubuntu, I get some appreciable value of the order of 10^{-4} s. I am assuming this is because Ubuntu is inherently slower than Windows and that my Windows is installed in an SSD while Ubuntu is in HDD.