CS210 Assignment-1

Jayant Agrawal-14282 , Shubham Pandey-14679

1 Maximum Values of n for which $fib(n)$ can be computed in a given Time Interval

- Recursive fibonacci Rfib

Time	.001s	.01s	.1s	1s	5s	60s	600s
Maximum Value of n	24	28	35	40	43	48	53

- Iterative fibonacci Ifib

Function	.001s	.01s	.1s	1s	5s	60s	600s
Maximum value of n	5.6×10^4	1.4×10^6	1.4×10^7	1.5×10^8	7.8×10^8	9.0×10^9	9.0×10^{10}

- Clever Fibonacci Clever-algo-fib

Function	.001s	.01s	.1s	1s	5s	60s	600s
Maximum Value of n	$> 10^{18}$	$> 10^{18}$	$> 10^{18}$	$> 10^{18}$	$> 10^{18}$	$> 10^{18}$	$> 10^{18}$

2 Time Complexity

Function	**Slope by graph**	**Order**	**Calculated Steps for** n
rfib	0.4745	O(1.607^n)	-
ifib	6.8×10^{-5}	O(n)	3n
cleverfib	3.274×10^{-8}	O(log(n))	35log(n)+11

3 Inference

- rfib is exponentially growing function wrt n so graph of n vs log of time gives a straight line with slope 0.47 . one can easily varify that no. of steps is $> 2^{.5n}$.from order we can say this sequence of time for larger n is similar to fibonacci series
- while Ifib is linear in time and efficient than rfib
- the cleverfib is most efficient in all three and takes less than a milisecond for whole long long int range hence we are getting a very less slope of the curve

4 Graphs

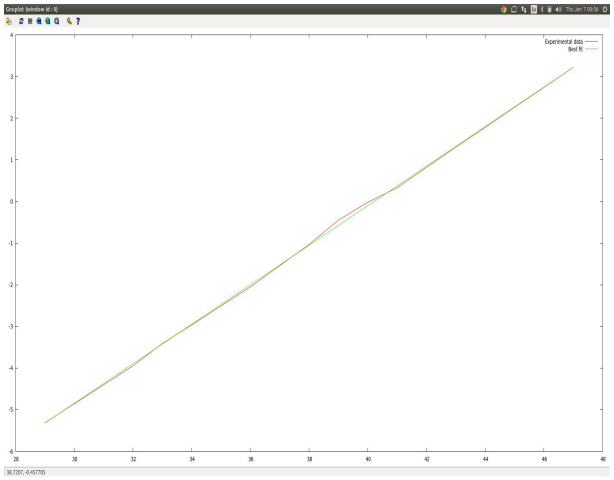

Figure 1: Log (Time taken by Rfib) as a function of n

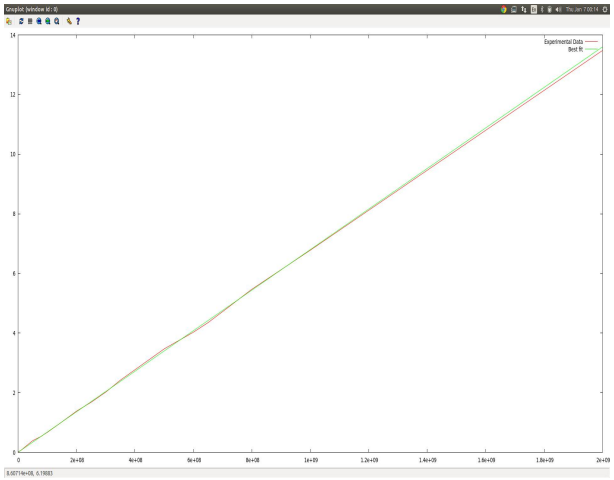

Figure 2: Time taken by ifib as a function of n

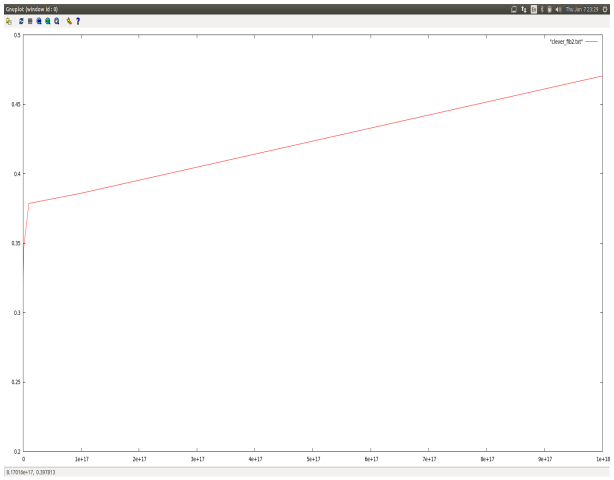

Figure 3: Time(in 10^{-5} s) taken by clever-algo-fib as a function of log(n)