GT username: sanne31@gatech.edu

Code Snippet for calculating run times for given functions: (assumed range of values to lower runtime)
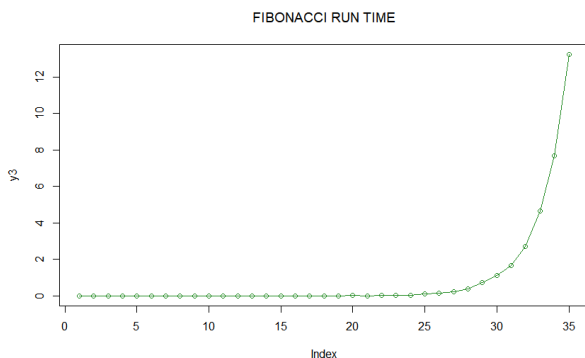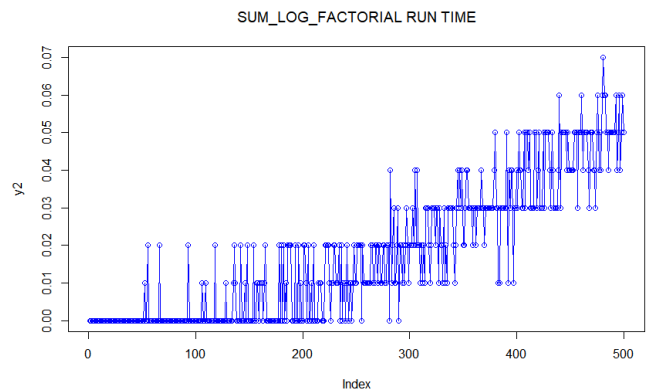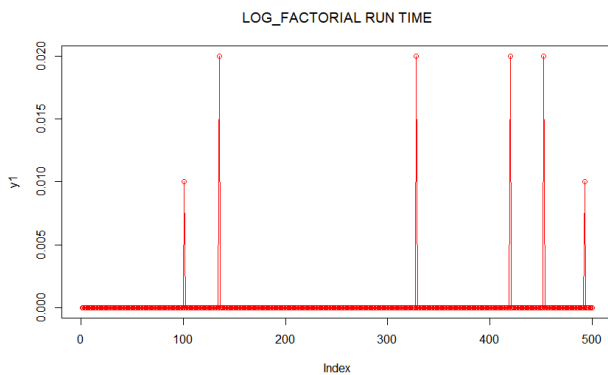
```
y1 = NULL
y2 = NULL
y3 = NULL
times1 = NULL
times2 = NULL
times3 = NULL
for (i in 1:500) {
  times1 = system.time(log_factorial(i))['elapsed']
  y1 <- append(y1,times1)
  times2 = system.time(sum_log_factorial(i))['elapsed']
  y2 <- append(y2,times2)
}

for (j in 1:35) {
  times3 = system.time(fibonacci(j))['elapsed']
  y3 <- append(y3,times3)
}
```

Time Complexities for functions:

    A.   log_factorial(n) -> O(log n)
    B.   sum_log_factorial(n) -> O(log n)
    C.   Fibonacci(n) -> O(2^n)

Plots for runtime of functions varying with n:

References: ->
http://rfunction.com/archives/2087 ;
https://mellowd.co.uk/ccie/?p=6122 ;
https://www.harding.edu/fmccown/r/