Travis Laxson – CS162
Lab 10

Iterative vs. Recursive Fibonacci sequence Execution Time Analysis

Tools and Sources

- **Python3 mathplotlib and pandas libraries for stat line plots**
- **Excel (.xlsx) file to import data into python – "runtime.xlsx"**
- https://www.codeproject.com/tips/109443/fibonacci-recursive-and-non-recursive-c (Links to an external site.)
- https://www.codeproject.com/Articles/21194/Iterative-vs-Recursive-Approaches
- http://www.cplusplus.com/reference/ctime/clock/ (Links to an external site.)

Data Plot Summary

Below are the results of a single program run for both of the programmed iterative and recursive Fibonacci sequences as N(x) increases, where x represents the number of iterations. The plots also show the change in clock ticks (dC) over the change in iterations, or executions (dN). The number of iterations for this test scenario is 50, hence N(50), but note the following relationship was also established at varying N values as low as N(10) and high as N(100). Only N(50) is shown here.

As expected, lower N(x) values show almost equivalent or negligible runtime differences. This is evidenced below in Figure 1 for N(10). However, runtime differences become more pronounced as N(x) increases. Figures 2 and 3, for N(15) and N(20) respectively, demonstrate an increasingly exponential trend as the y-axis expands accordingly to accommodate the expansion in recursive runtime. As Figures 4-8 confirm, the recursive function runtime (in clock ticks) increases on a seemingly exponential scale. The iterative function remains plotted, but the runtime never exceeds the value 8 from N(1) to N(50). The iterative max runtime occurs at N(43) while the recursive max runtime is ever increasing as N(x) increases.

Figure 1

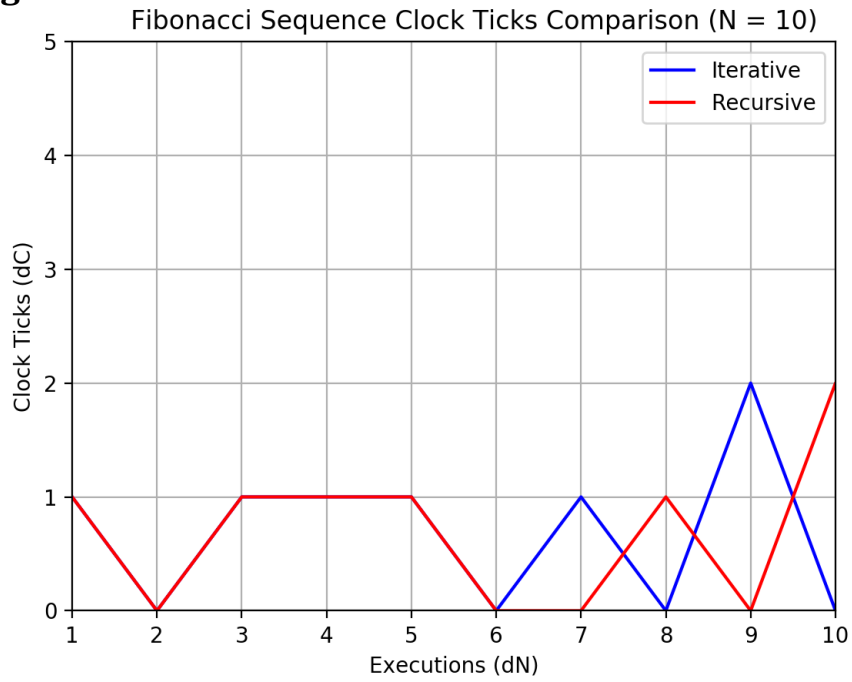


Figure 2

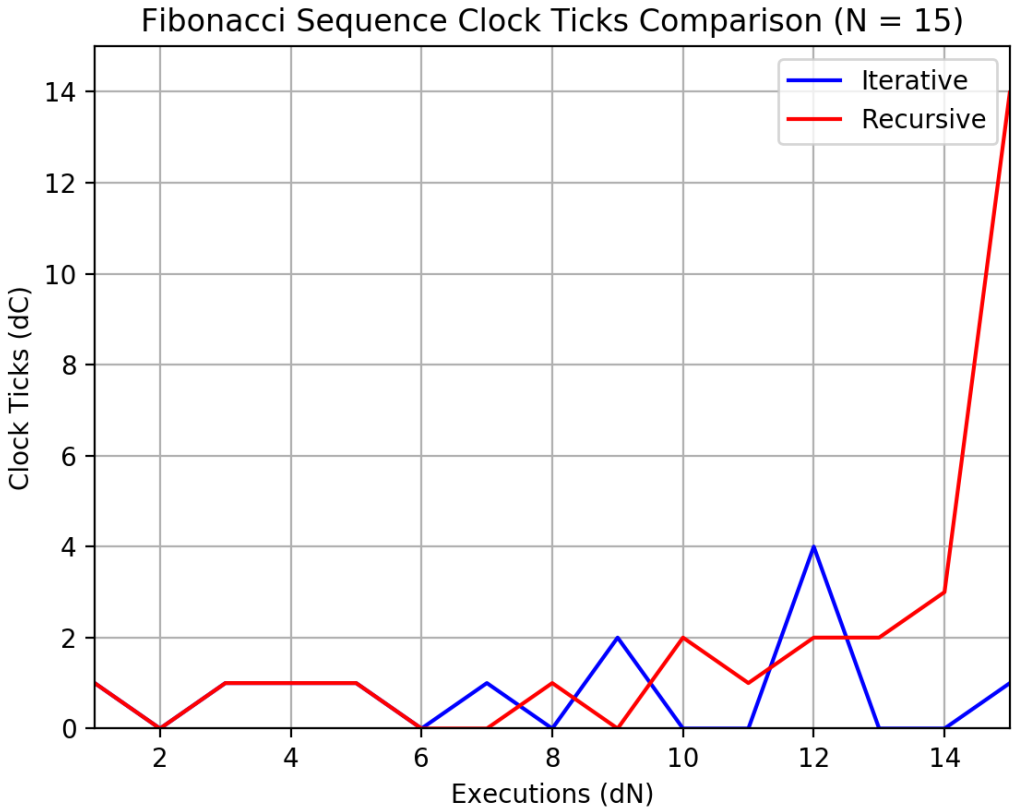


Figure 3

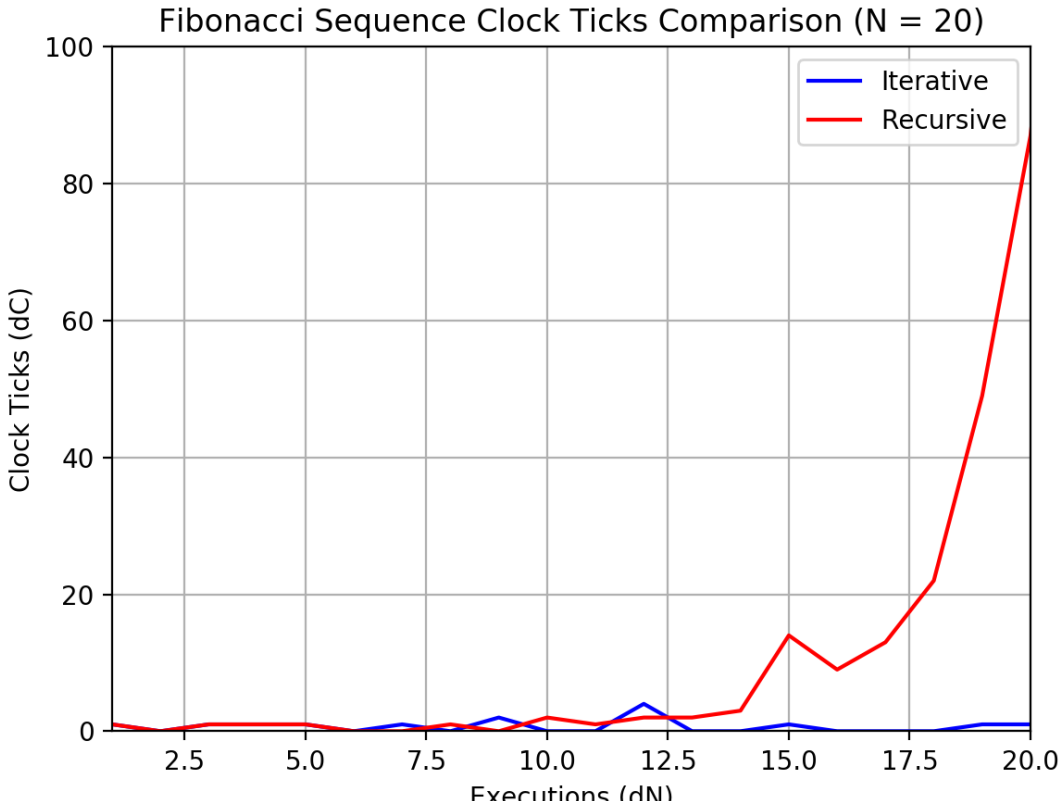


Figure 4

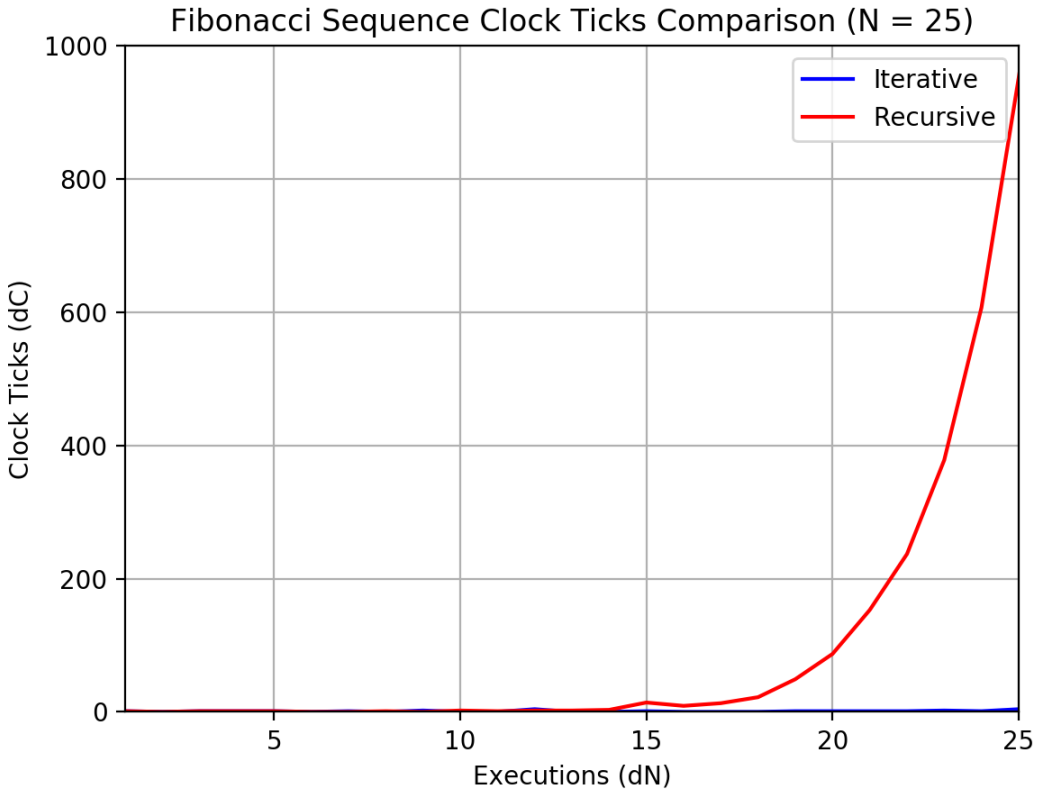


Figure 5

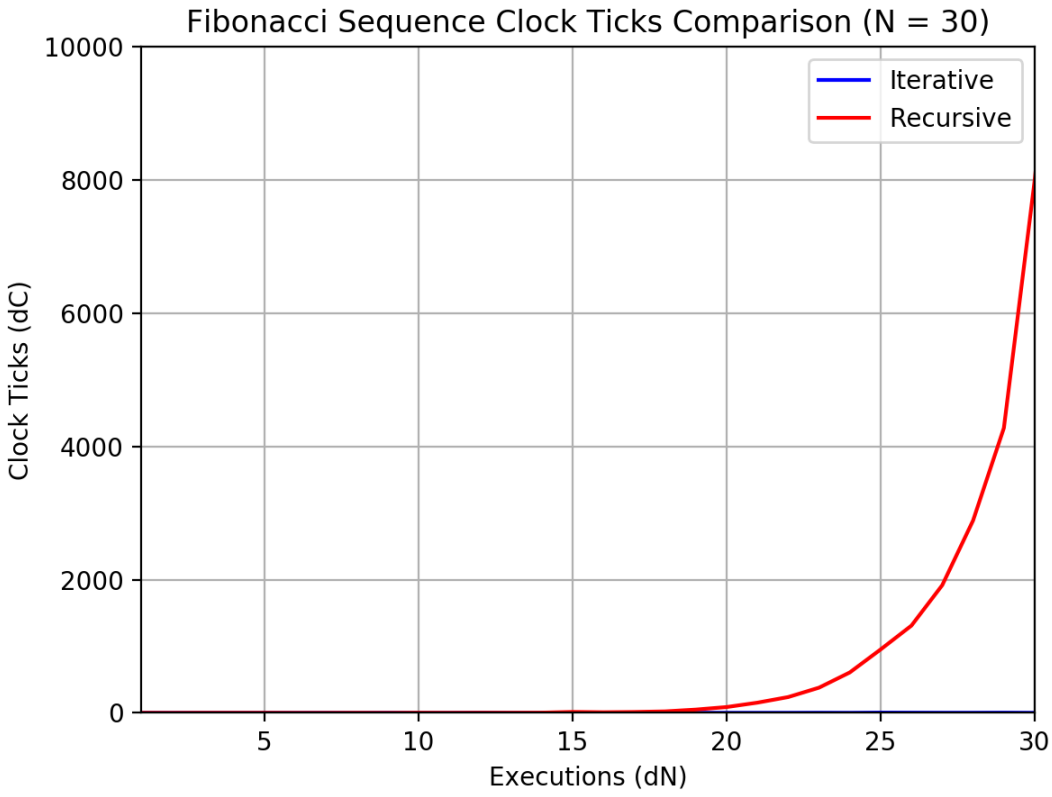


Figure 6

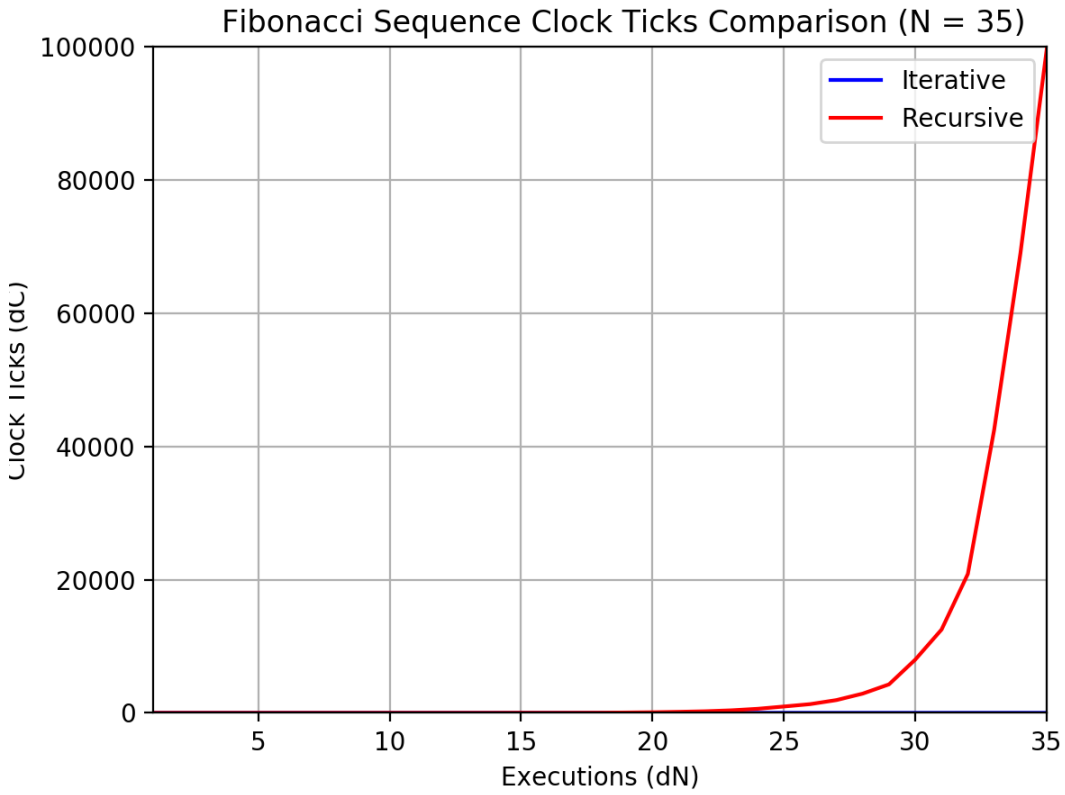


Figure 7

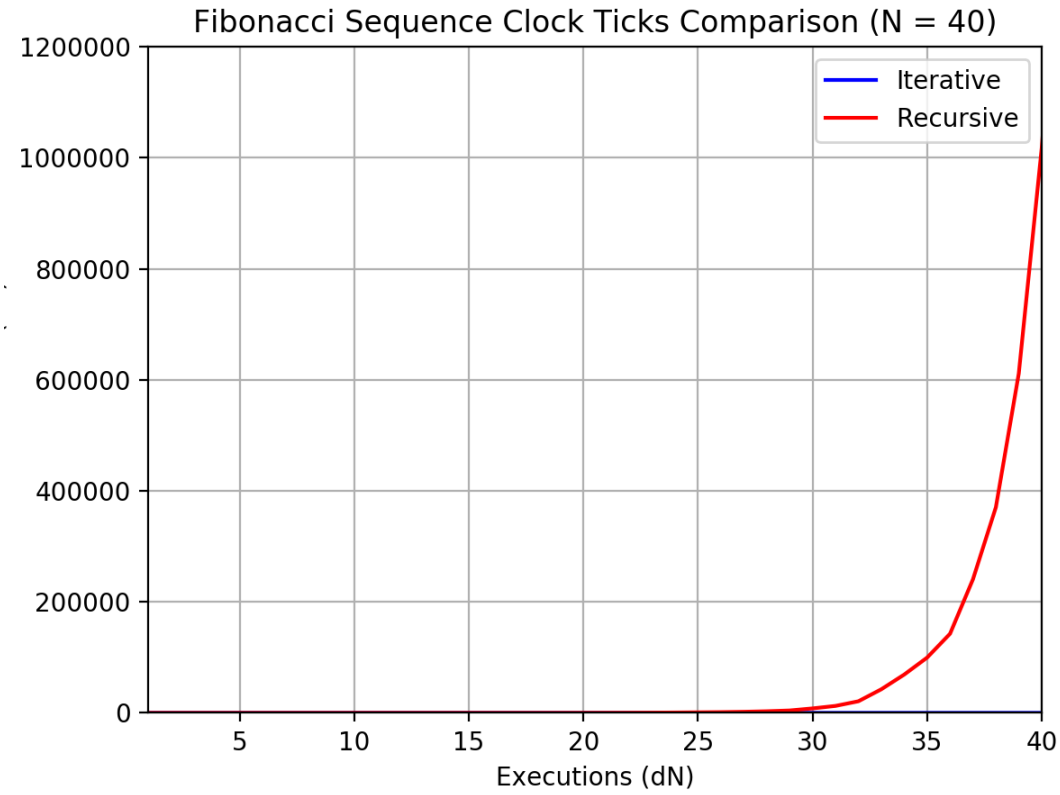
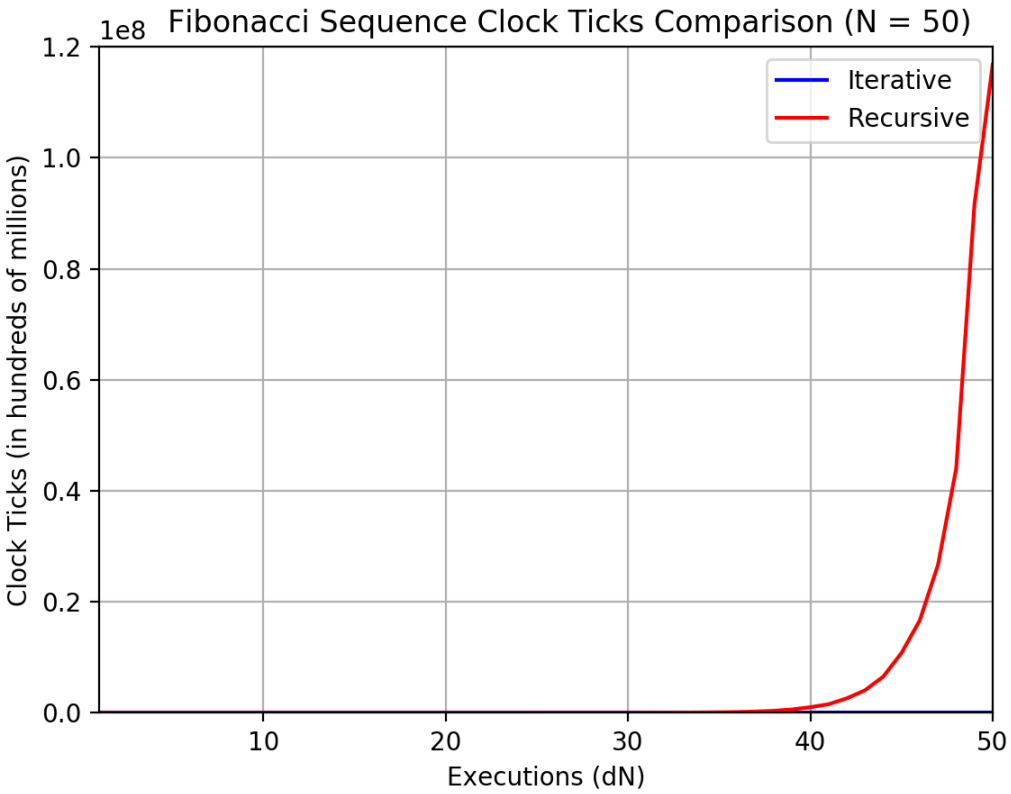


Figure 8


Fibonacci Sequence Clock Ticks Comparison (N = 50)
1e8
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Clock Ticks (in hundreds of millions)
10
20
30
40
50
Executions (dN)
Iterative
Recursive