CS2XB3 - Lab 02

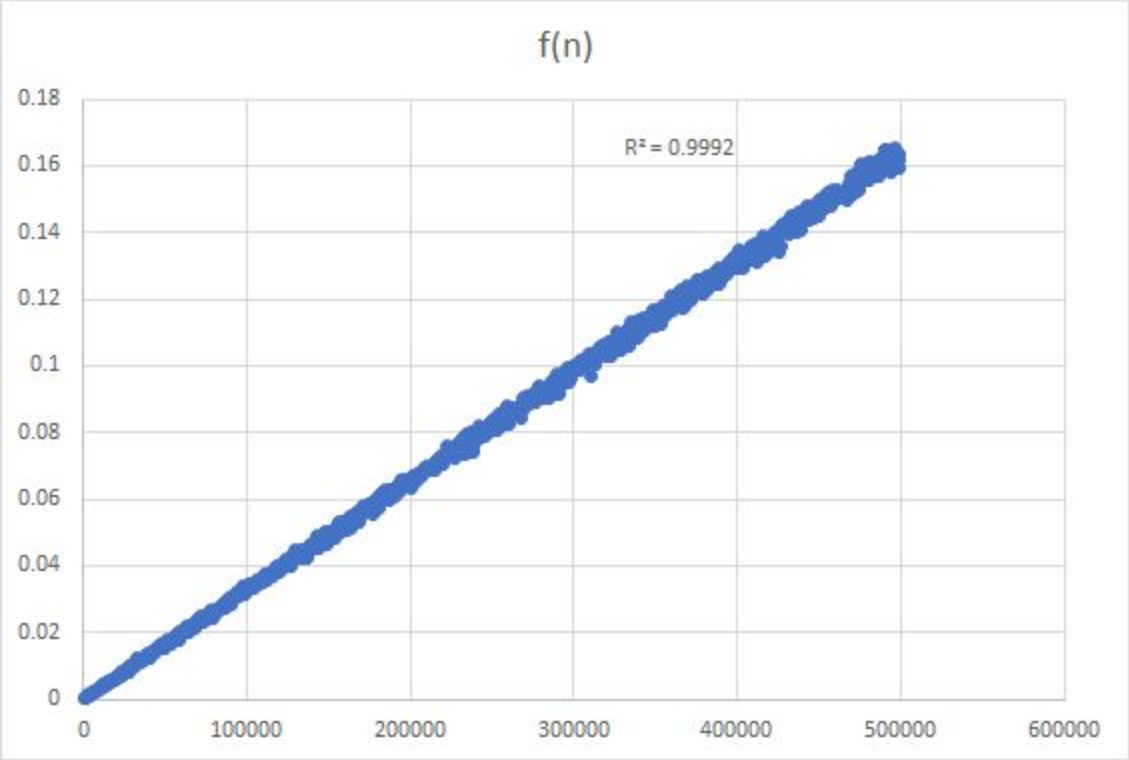
Group Number: 06
Lab Session: L02

Group Members:

Sarvesh Paarthasarathy - 400285716 | paarthas@mcmaster.ca
Joshua Sam Varugheese - 400255799 | varugj1@mcmaster.ca

Timing Data Report:

1. f(n):



Graph Type: Linear

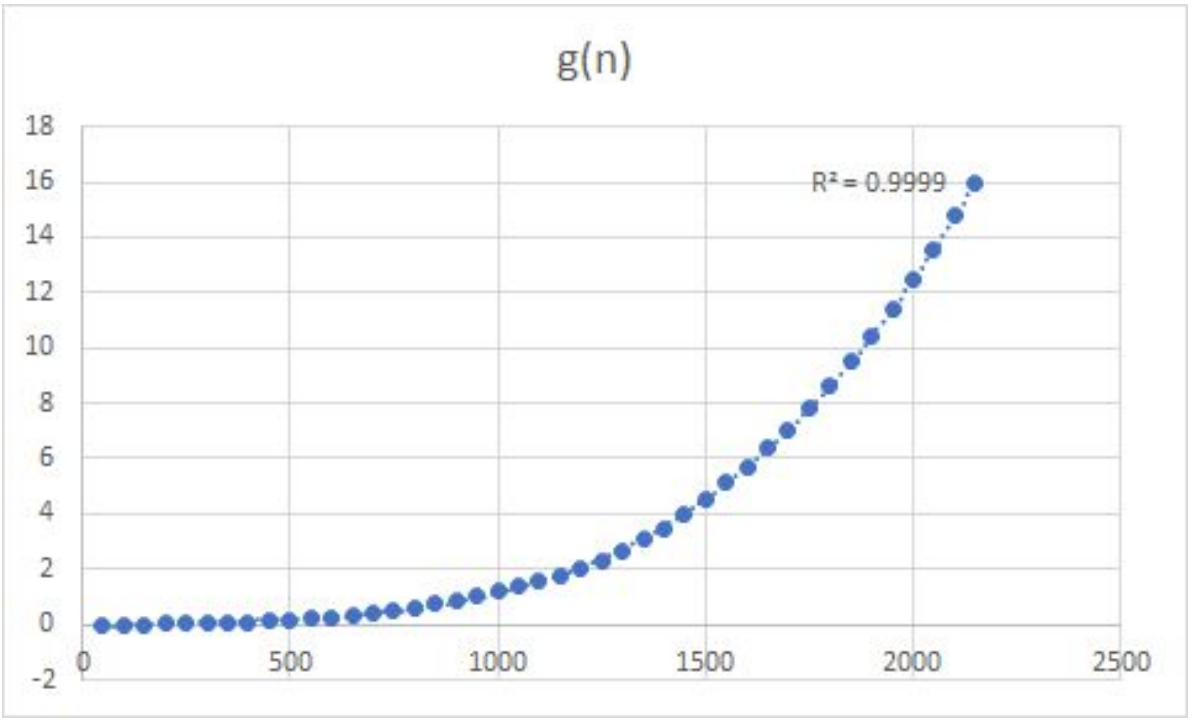
Equation on Chart: y = 3E-07x + 1E-05

R-squared value on Chart: $R^2 = 0.9992$

Description:

The given values when plotted on a graph gives us a linear relationship. That being said, both linear and polynomial functions were very close to each other with the same r-squared values. The final choice was linear as it was more convincing.

2. g(n):



Graph Type: Polynomial

Order (n) : 3

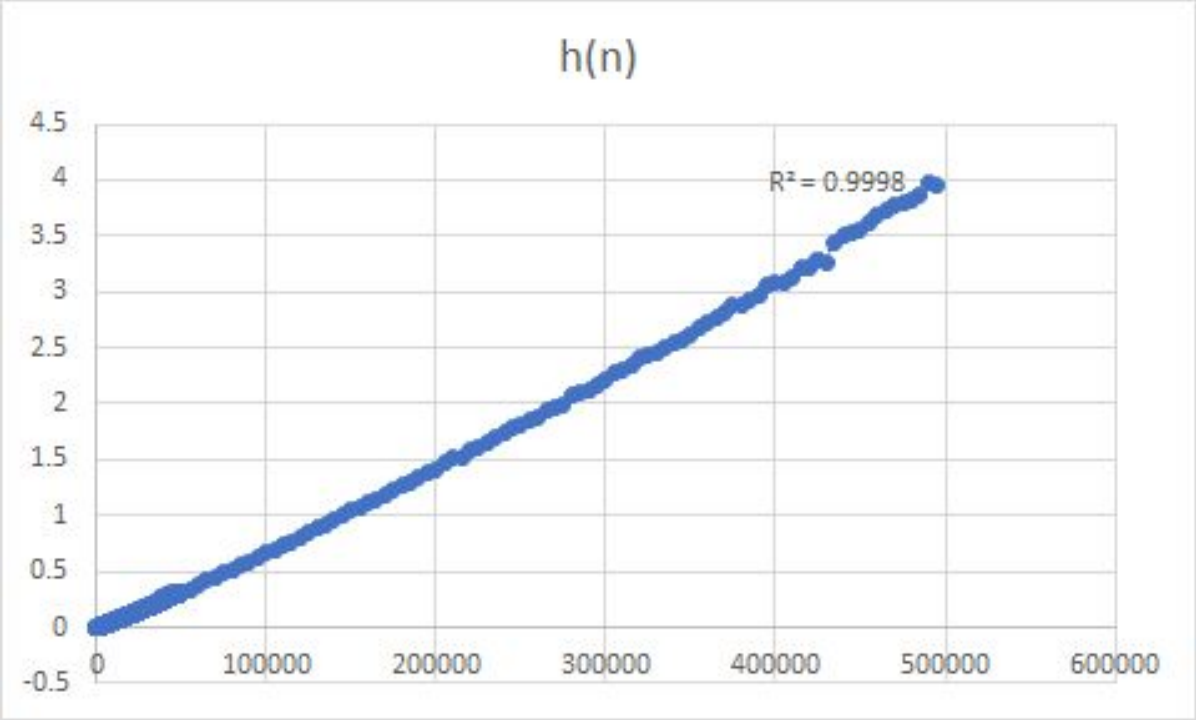
Equation on Chart: $y = 3E\text{-}09x^3 - 3E\text{-}06x^2 + 0.0015x - 0.1707$

R-squared value on Chart: $R^2 = 0.9999$

Description:

The given values when plotted on a graph gives us a polynomial relationship. The best approximation was given with r-squared value of 0.9999 of order, n=3.

3. h(n):



Graph Type: Polynomial

Order (n) : 2

Equation on Chart: $y = 3E\text{-}12x^2 + 6E\text{-}06x - 0.0118$

R-squared value on Chart: $R^2 = 0.9998$

Description:

The given values when plotted on a graph gives us a polynomial relationship.
The best approximation was given with r-squared value of 0.9998 of order, n=2.

Copy():

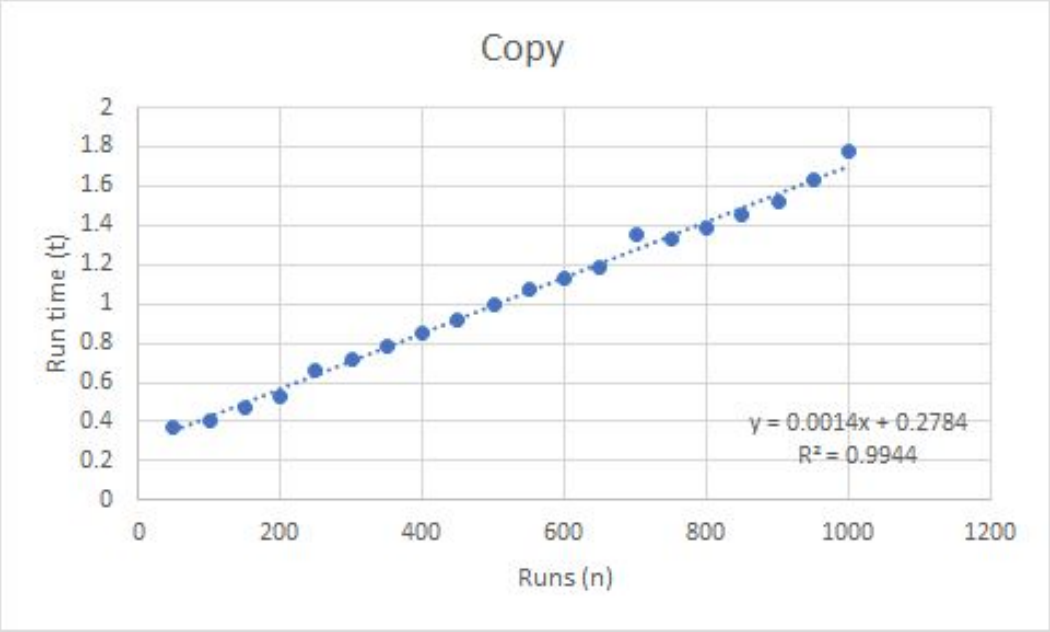
Description:

The copy method is tested using a lambda function and timit to extract the time taken to copy an arbitrary list. The copy method is tested for 1000 values starting from 50 to 1050 on various list sizes.

Observations and Conclusions:

The time complexity of the copy function is O(n) as the evidence depicts a linear graph with an R-squared value of 0.9944

Evidence:



Explanation:

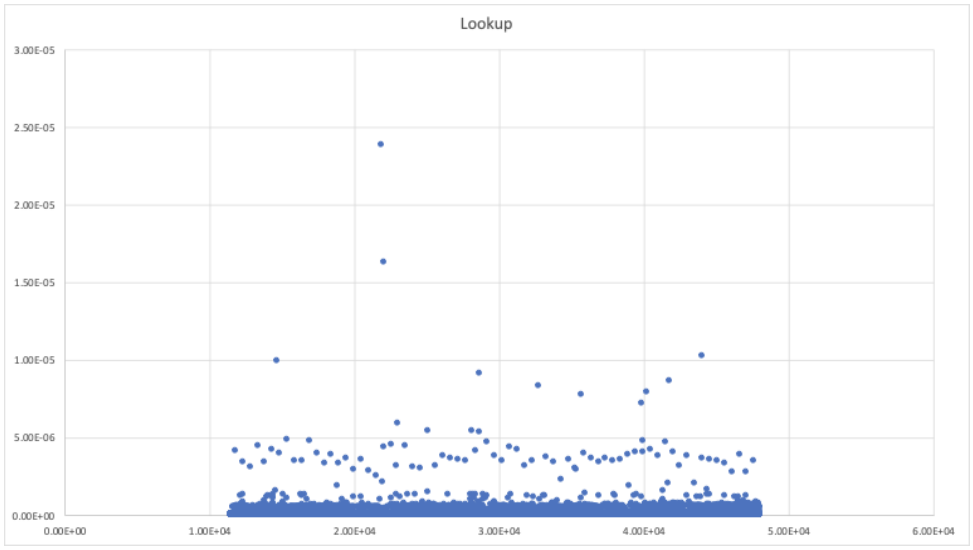
The copy function has a linear graph since the input list size is being altered each time the copy method is called.

Lookup():

Predictions:

The common prediction regarding the data is that the runtime complexity of lookup is O(1) and the graph being constant.

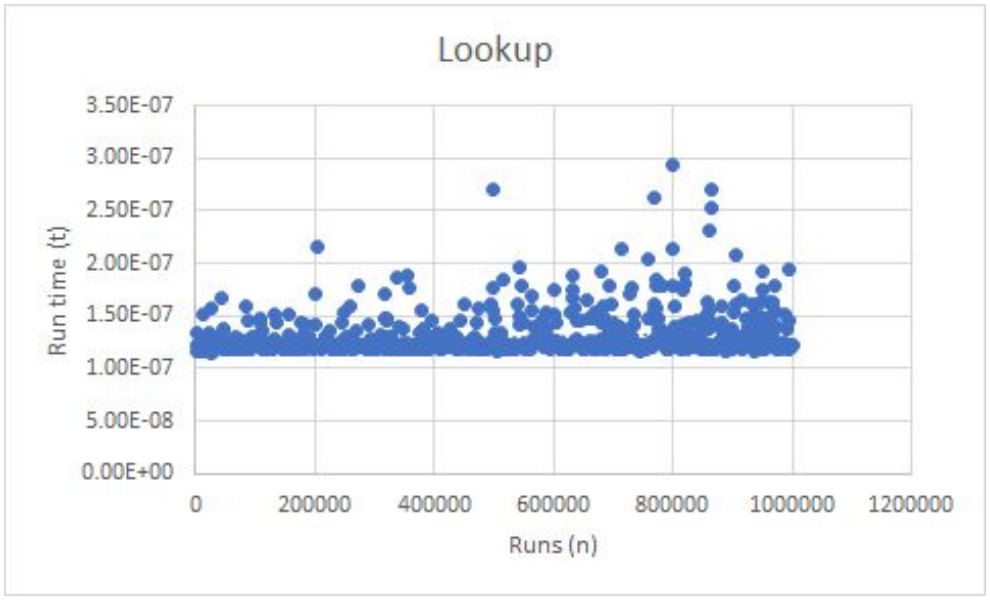
Scatter Plot Graph:



Problems regarding experiment:

The problem faced during the experiment was being unable to extract one-million values to plot the scatter graph from the console. The values obtained were not convincing until they were fixed. A problem faced was not finding the average of 1000 runs instead total time was used. This problem was efficiently solved by taking the average runtime of 1000 runs, 1000 times. Another problem faced during the experiment was being unable to obtain a perfect constant graph (due to background processes running in the system). This wasn't resolved as it was beyond our scope.

Conclusion and Fixed Scatter Plot Graph:



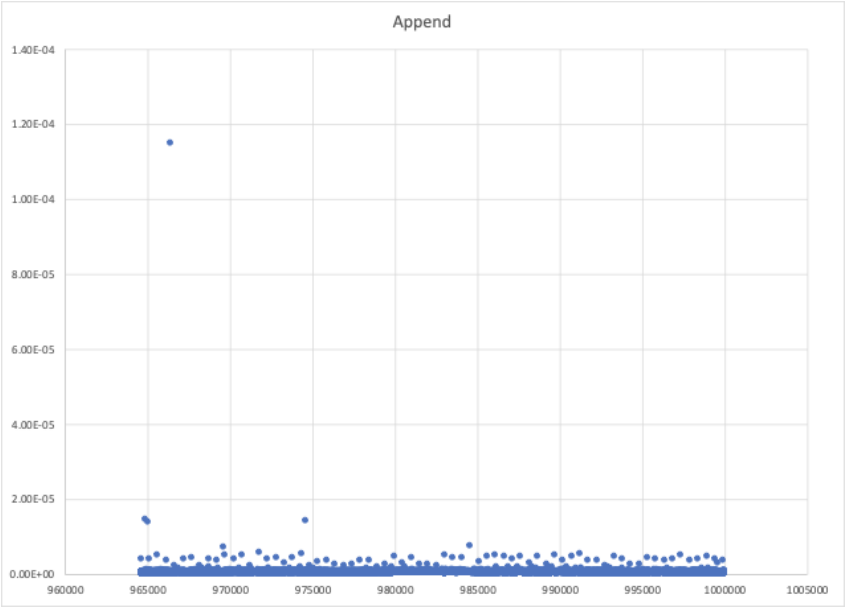
The obtained graph is consistent with runtime and the runtime complexity of the copy method is proved to be O(1).

Append():

Predictions:

The common prediction regarding the data is that the runtime complexity of append is O(1) and the graph being constant as list[i] would always have a consistent runtime.

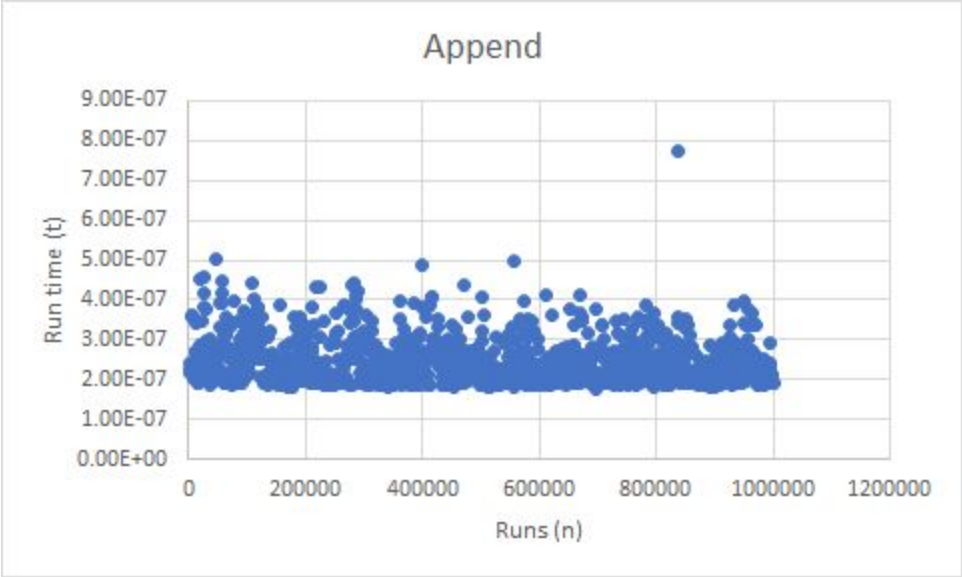
Scatter Plot Graph:



Problems regarding experiment:

The problem faced during the experiment is similar to that of lookups, being unable to extract one-million values to plot the scatter graph from the console. The values obtained were not convincing until they were fixed. A problem faced was not finding the average of 1000 runs instead total time was used. This problem was efficiently solved by taking the average runtime of 1000 runs, 1000 times. Another problem faced during the experiment was being unable to obtain a perfect constant graph (due to background processes running in the system). This wasn't resolved as it was beyond our scope.

Conclusion and Fixed Scatter Plot Graph:



The obtained graph is consistent with runtime and the runtime complexity of the copy method is proved to be O(1).