Jackson Warren

CSCI 330

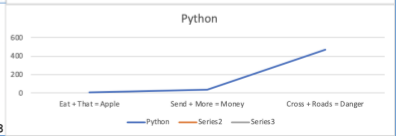
Prof. Seif

10/4/21

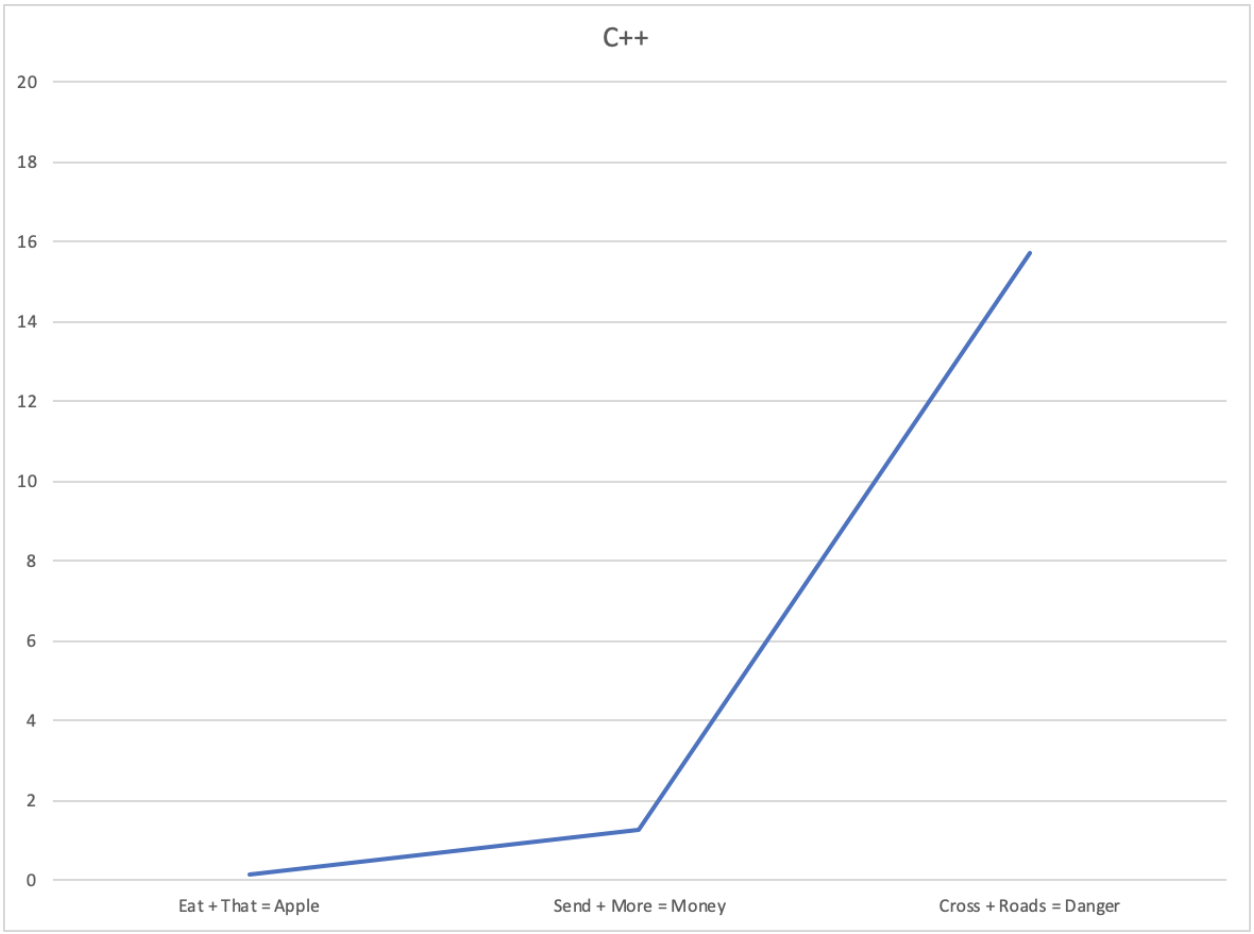
Project 2

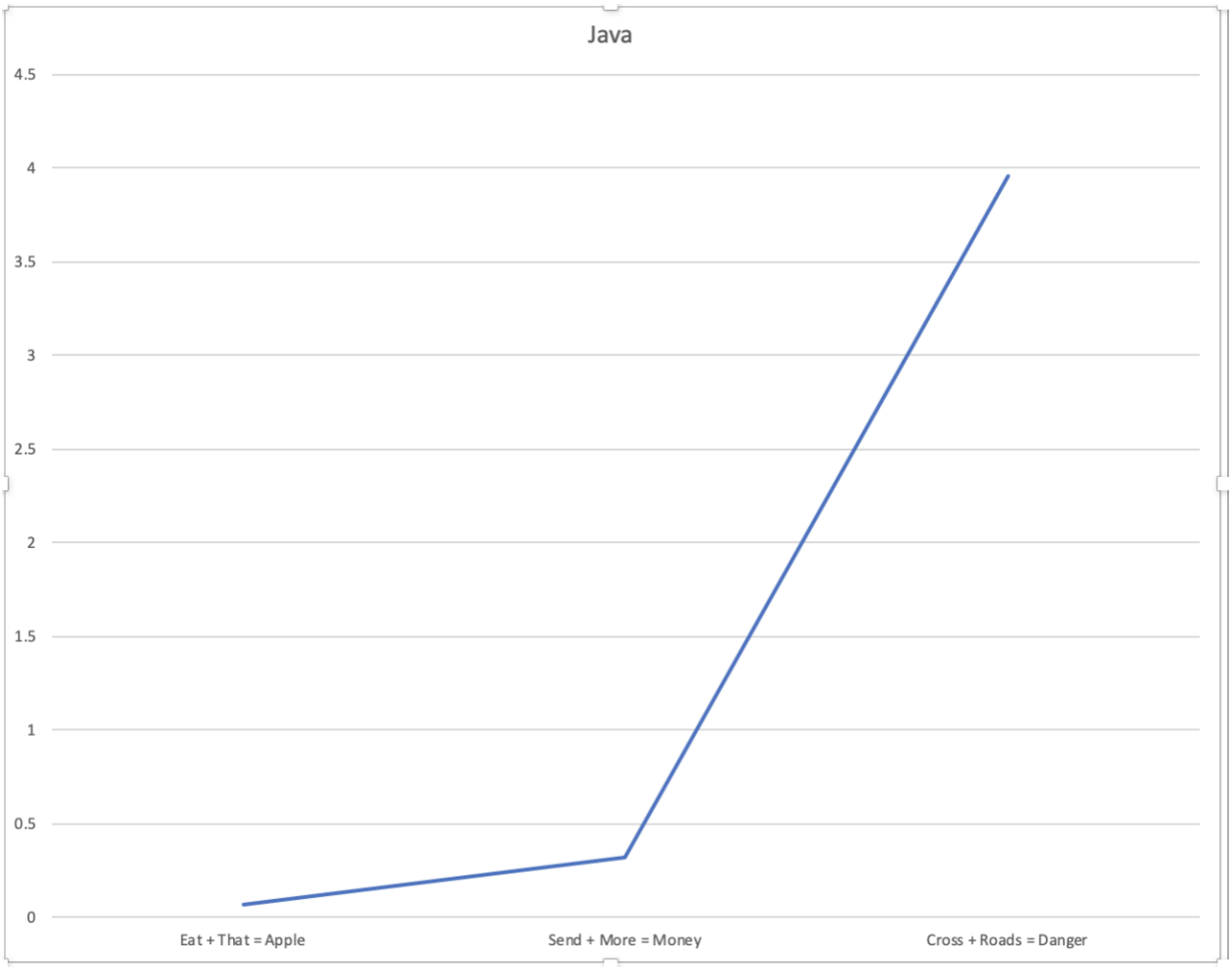
Notes for coding methodology: I used the exact same tactics as Project 1, following the algorithms discussed in class closely, as well as keeping the format and methodologies as close as possible as I transitioned between the three languages.

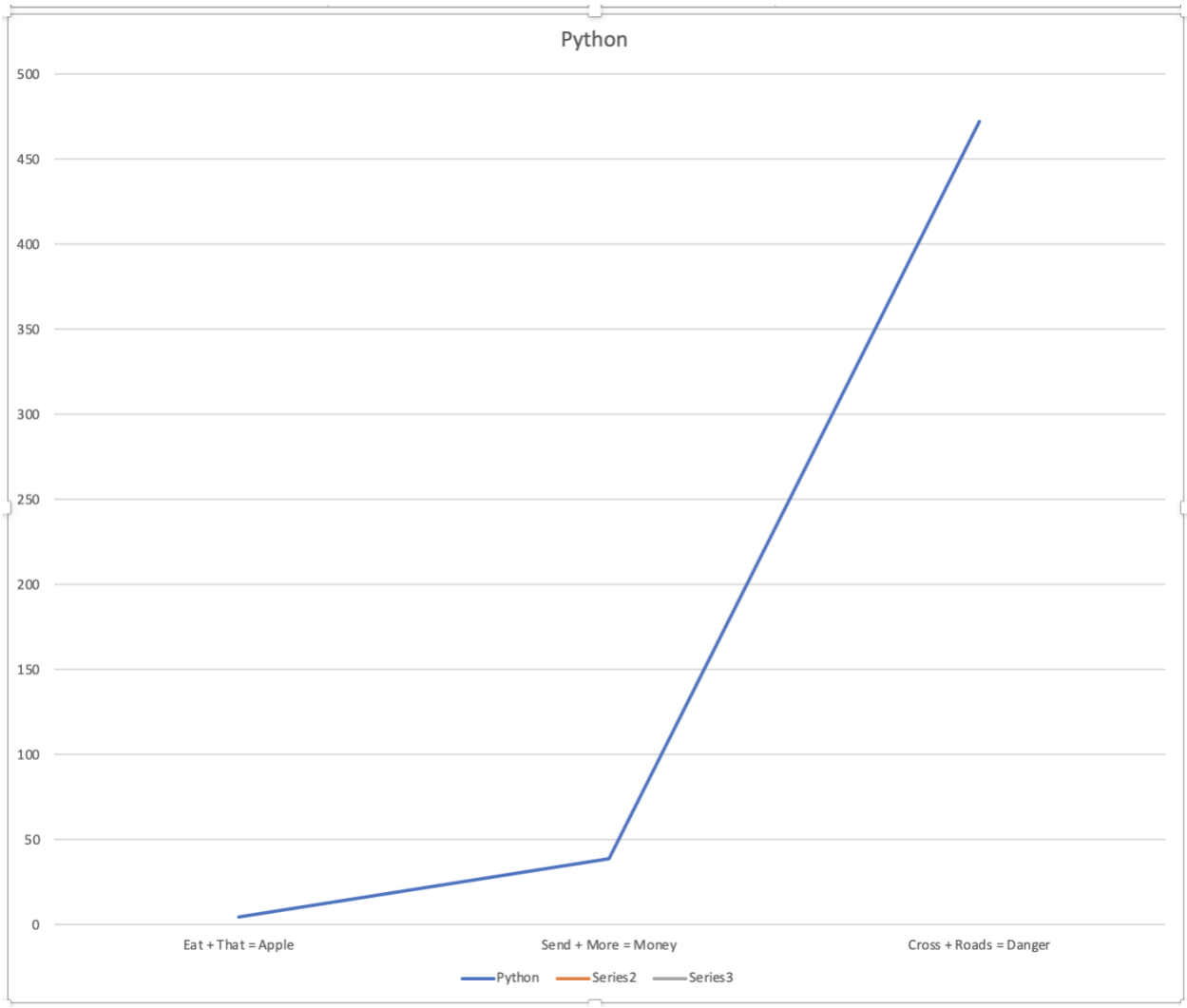
Chart of elapsed times:

Language	Eat + That = Apple	Send + More = Money	Cross + Roads = Danger	Graph
C++	0.136378	1.27442	15.7153	
Java	0.067567	0.324719	3.952204	
Python	4.6841475	39.12002	471.6943683	

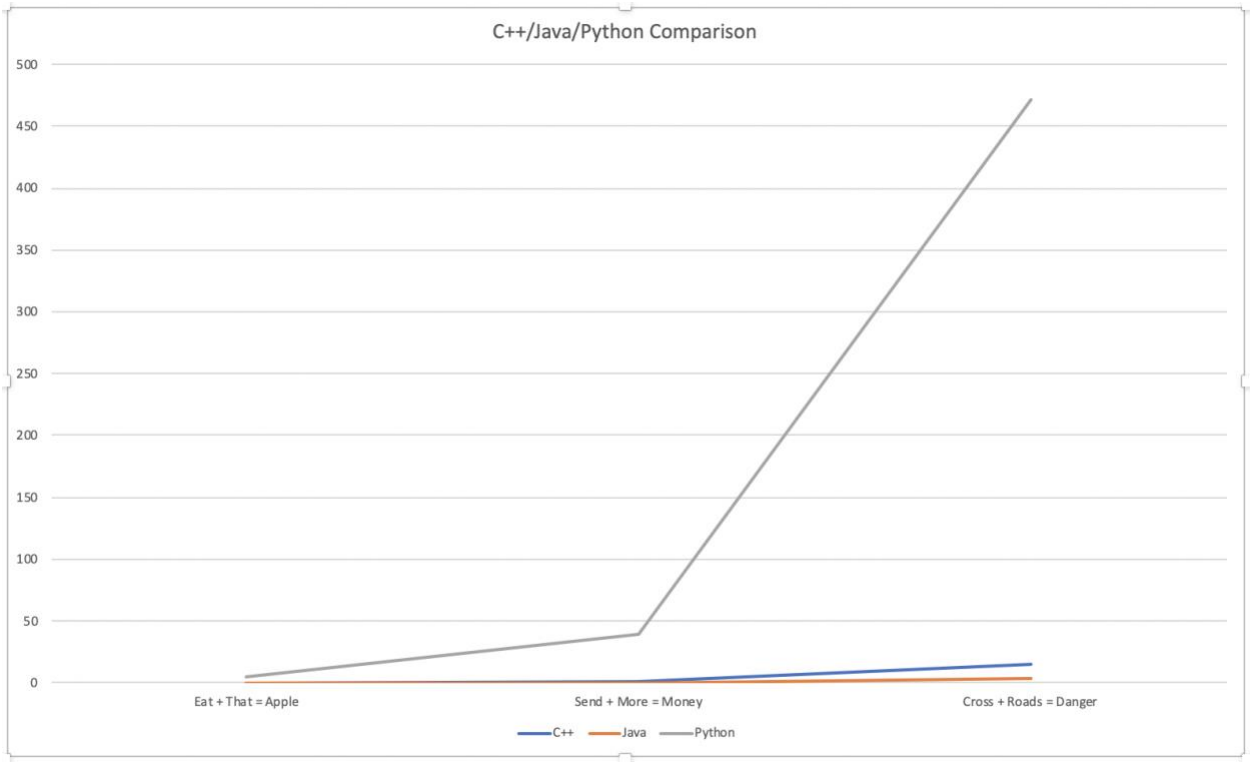
Larger Graphs:




Java
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Eat + That = Apple
Send + More = Money
Cross + Roads = Danger


Python
500
450
400
350
300
250
200
150
100
50
0
Eat + That = Apple
Send + More = Money
Cross + Roads = Danger
Python
Series2
Series3

Comparison:



Analysis:

Analyzing these graphs, we can see that each language, when being compared in relation to itself, all increase in time to execute at similar rates. The rate of increase (in seconds) between the three different operations is very similar for each language. The only difference appears to be the time it takes to execute in different languages, python being significantly slower than the other two. When compared on the same graph together, we see that both Java and C++ obviously have much faster run times, as well as much lower rates of increase.

Based off my results, Java has the best run times, operating moderately better than C++, as shown in the comparison graph. Java also had a slightly lower rate of growth when compared to C++. Python was massively slower than the other two languages, especially as the number of operations increased. This gap in performance only became more apparent through

each program. Python was also the most different from the other two programs syntactically, which may very well mean that python syntax is the reason for its slower operation.