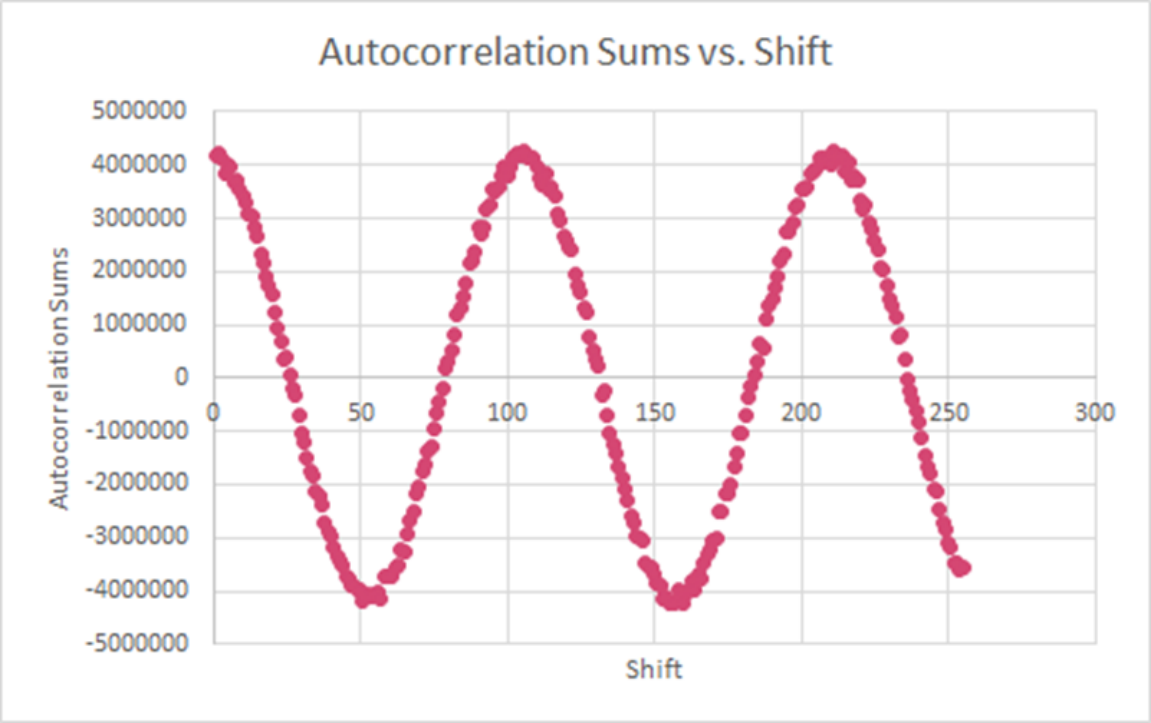
Gretel Rajamoney
rajamong@oregonstate.edu
Project #7B

Project Questions:

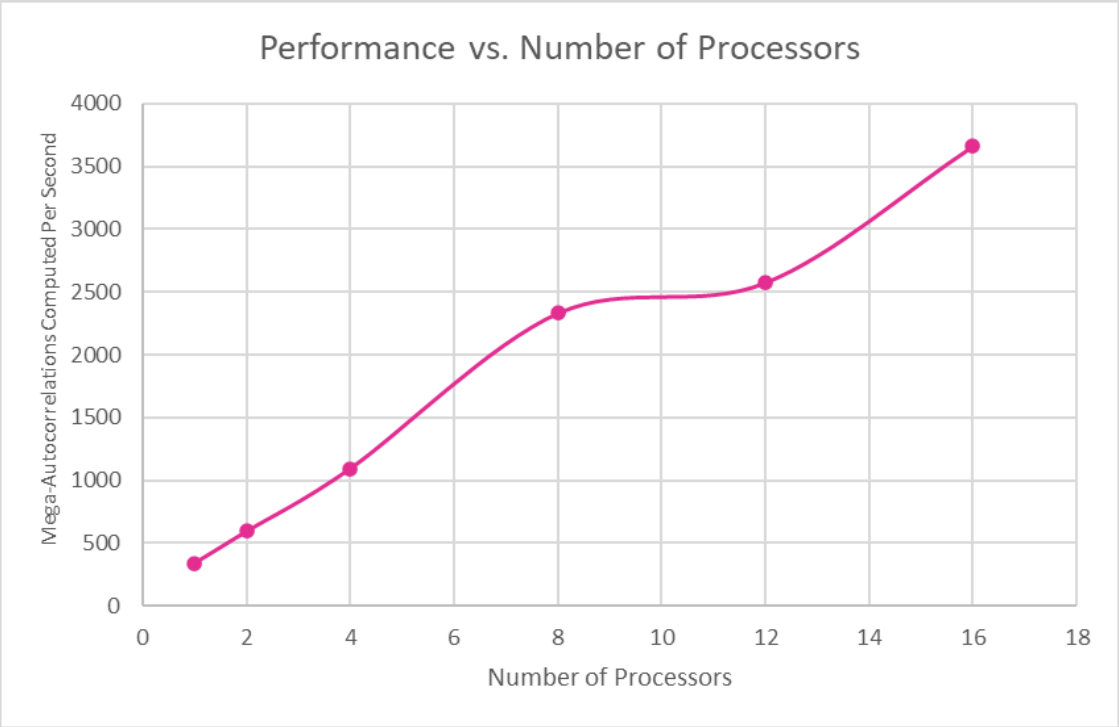
0. What machine did you run this on?
   = I ran my program on my Windows machine on Visual Studio Code utilizing the engineering server rabbit.engr.oregonstate.edu. To run my program in the terminal, I inputted the following lines of code:
   ```
   chmod u+x proj07.sh
   sh proj07.sh >& proj07.csv
   ```

1. Show the Sums[1] … Sums[255] vs. shift scatterplot?



2. State what the secret sine-wave period is, i.e, what change in shift gets you one complete sine wave?
   = The secret sine-wave period that I am analyzing from the data that has been visualized on the scatterplot from the previous question is between the shift range of approximately 26 to 131. I got these shift values through finding the autocorrelation sum values within the plot.csv file that shifted from positive to negative. Since 131 subtracted by 26 is equal to 105, the secret sine-wave period is calculated to be approximately 105 shifts.

3. Show your graph of Performance vs. Number of Processors used?



4. What patterns are you seeing in the performance graph?
   = In the graph representing the Performance versus Number of Processors, there appears to be a direct correlation between the number of processors used and the performance of the program. As seen within the graph, as the number of processors used increases, the line connecting all of the plotted data points increases as well. This positive correlation between the two variables is present throughout the entirety of the graph, telling us that a higher number of processors used leads to a higher overall performance.

5. Why do you think the performances work this way?
   = In the graph representing the Performance versus Number of Processors, there is a positive direct correlation between the number of processors used and the performance of the program. This pattern is prevalent within the graph because as we are adding more items to be worked on by our program, the higher the number of calculations needing to be calculated, the more resources that can be utilized to calculate them. Essentially our program is capable of performing using a greater number of processors since it is able to efficiently utilize its resources to handle them, thus resulting in a higher overall program performance.