Very high-level design Pseudo Code (Before I completed any code)

In my bash script:

- Use the monte_carlo executable to generate data
- Use -n option to define how many points are made
- Use grep and or awk to remove unnecessary columns and rows
- Save results into data files for gnuplot
- Use here-document in script to create a gnuplot command file
- Run Gnuplot to graph the data sets

Different shell codes generate figure 2 and figure 3 since they require different data points.

More Detailed Design (After Code has been implemented)

In my plot.sh script I had

- Two parts for figure 2 and figure 3
- The default number of points for figure 2 was set to 1000 points
- Check if the script is called with an argument it takes as the number of points for figure 2
- Use monte_carlo with an argument to generate the data
- Use grep -v commands to remove the header row
- Use awk to select the required columns for the data; store results in all.data
- Use grep -E with a regular expression to separate what data is in or out of the circle; store data in either inside_circle.data or outside_circle.data
- Use heredoc to run gnuplot to plot figure 2
    - In the circle graph, blue points
    - Outside the circle graph, red points
    - Use a formula to create a circular arc
- For figure 3, the script runs monte_carlo five times
- To get different random seeding at time delay was implemented
- Use awk to select columns and calculate the difference between the estimated pi and a preset value for pi
- Stores the five results into five data files
- Use heredoc to run gnuplot figure 3
    - Set grid
    - Set logarithmic scale base 4