Krisha Sharma
CSE13s
Winter 2022 Long
01/09/2022

Assignment 1 Design Document

This assignment asks you to create a plot.sh file that holds three for loops that each individually plot one of the figures from the assignment document. The plot.sh file is written in bash shell scripting, utilizing the given collatz.c file, for loops with bash scripting commands and gnuplot commands in order to produce the required plots.

Collatz Sequence Lengths Figure 1:

The objective of this project is to create a graph that displays the collatz sequence lengths. A C program that produces collatz sequences is given.

To start, all the lengths of a given collatz sequence should first be placed in a file. By getting the length of the collatz sequence and placing it into a length.dat file I will be able to easily obtain the line count and know how many lines were generated. The bash wc command will specify the line count of a file and this is needed to find the length of the collatz sequence, which is also what we need to create the y axis of the first graph. It is necessary to pipe the starting point, in this case $n, into the wc to get the line count and then write it to /tmp/length.dat that way it is able to be read by gnuplot. At this time /tmp/length.dat only has the sequence length. In order to get an x coordinate it is necessary to echo x and place everything on the same line into a length.dat file for gnuplot to read. Echo x naturally creates a new line; however, it is possible to not print the trailing new line character by adding a -n. All of this will be placed within a for loop that has a range of 1 to 10,000 as the x axis goes to 10,000 in the figure provided in the assignment 1 document.

In the gnuplot commands it is important to change the output to length.pdf, the title to sequence lengths and the plot location to length.dat with points. For figure 1, it is important to take the existing program and format the output to what was needed; that way gnuplot is able to plot out the sequence lengths derived from the collatz sequence which is given. (Eugene Section 1/7/2022)

Maximum Collatz Sequence Value Figure 2:

For this figure, I will first need to write another gnuplot command which is similar to the one given for the collatz sequence length graph, that was modified to fit what was needed for the first figure. The gnuplot commands for this figure will also be slightly different. The output will be set to maximum.pdf, the title will be “Maximum Collatz Sequence Value”, the xlabel will be “n” and the ylabel will be “value”. For this graph I will also need to specify the y range by adding “set y range [0:100000]” which will extend my y range to what it needs to be. Gnuplot will also plot the location to maximum.dat with points.

In order to create this figure, I will first need to get the maximum value of a sequence. By using a for loop to sort by the largest number and take the zeroth element we will be able to find the maximum value. If I use the sort command on the collatz file I should be able to see the maximum value of the sequence; however, there are two things to note. Firstly, it is necessary for me to pipe to less that way I can parse through the file to see the values. Furthermore it is also good to note that when the ./collatz | sort | less command is given, the values generated are shown lexicographically rather than numerically. I want to see the maximum value of the sequence, and thus I need the values to be sorted numerically. By adding -nr after sort, the numbers will sort in numerical order and they will be displayed in reverse, meaning that the maximum number will be at the top rather than at the bottom. (Eugene Section 1/7/2022)

Collatz Sequence Length Histogram Figure 3:

To start to create this figure I will first need to place all the sequence lengths into one file in order to see that there are clearly repeating numbers. In order to create a graph that displays a histogram of Collatz sequence lengths starting from $n \in \{ 2, \ldots, 10000 \}$ I will need to uniq the repeated elements and then plot them. The uniq command in bash reports or filters out repeated lines in a file and it can also precede each output line with the count of the number of times the line occurred in the input if a -c is added afterwards. By doing this to the length.dat file I will be able to see how many times each element is repeated. The command cat /tmp/length.dat | uniq -c | less will be able to show me the exact amount of times an element is repeated in the collatz sequence lengths. (Eugene Section 1/7/2022)

The gnuplot commands will also remain the same in structure for this figure; however, the contents will once again change, just like for figure 1 and 2. The gnuplot commands for this figure will have the output going to “histogram.pdf”, the title will be set to “Collatz Sequence Length Histogram”. The xlabel will be set to “length” and the ylabel will be set to “frequency”. Gnuplot will plot the location to histogram.dat using 2:1 (meaning that the x should be column 2 and the y should be column 1, thus inverting the axis) with impulses rather than points.

In order to properly create figure 1, I will first start by getting the length of the collatz sequence and place it into a file. ./collatz > /tmp/collatz.dat is a bash shell scripting command that has collatz output to a collatz.dat file. The collatz.dat file will now contain a collatz sequence. In order to get the length of the sequence in a file I will use wc which is a bash command that gives me the line count of a file. By piping the output of collatz into wc and specifying that I want the line count I will be able to get the amount of lines in the file. The next step I want to do is get the length of several sequences from different starting points. I will use a for loop to do this and set the range at 1 to 10,000 as the x axis goes to 10,000 in the figure provided in the assignment 1 document.

Echo is a Unix/Linux **command tool used for displaying lines of text or string** which are passed as arguments on the command line. This is one of the basic commands in linux and most commonly used in shell scripts.

$n specifies that whatever the value of n will be the start point.