Theodore Ikehara
tikehara@ucsc.edu
12 Jan, 2022

CSE13s Winter 2022
Assignment 1: Getting Acquainted with UNIX and C
Writeup Document

Reflection: What I learned
In this assignment I have learned much about bash scripting and how gnuplot works. In this assignment I had to use an existing C program and manipulate the output to graph with gnuplot. All this work was automated and used to create three graphs. Bash scripting is very useful outside of just this assignment. Bash scripts could be used to automate many other things in linux.

Commands that I used in this assignment:

- rm -f
    - removes the file, forcefully
- sort -rn
    - sorts in reverse numerically
- uniq -c
    - finds the unique ints and counts how many times it occurs
- wc -l
    - counts the length of the output

Other miscellaneous blocks:

- ./collatz -n {num}
    - runs the collatz function starting with specified number
- >
    - overwrites the file
- >>
    - appends to the file
- |
    - runs the commands one after another, concatenating the commands

My script: How was it implemented

My bash script is simply split by 3 parts: initializing the script, getting the data to plot, and plotting the data. We start the script with #!/bin/bash and this portion is telling the computer that this is a bash script. After declaring this we can then start writing our script. We first start by making the collatz program, creating the executable and removing all the unnecessary files such as the old executable files and the .dat files in the tmp directory. After making sure that the rest of the script could be executed without any issues we proceed to a loop where we will iterate through the collatz from n ∈ {2, … , 10000}. We then check the length of collatz with length n iterating through and storing this to a temp file. This will allow us to store the data needed for the collatz sequence for the first graph (figure 1). Using the same loop we then proceed to finding the max integer of each sequence by sorting it in reverse so that the first int of the sequence is the largest number of the sequence and take the head, with the command head -n 1. This will grab the first int which is the largest value and append to another tmp file. This would create the second graph (figure 2). The third graph we create by first exiting the loop and sorting the length.dat that we created in the tmp directory. This step is necessary as we will be using the uniq command which will help us count the amount of duplicate sequences but this command only counts the duplicates of adjacent numbers. After sorting is complete we can then use the data that was just produced and use uniq -c to count all the duplicates to get the numbers and frequency of appearance. This will yield the third graph shown (figure 3). Then after all the data is collected we can then plot all the data in gnuplot with dots or boxes.

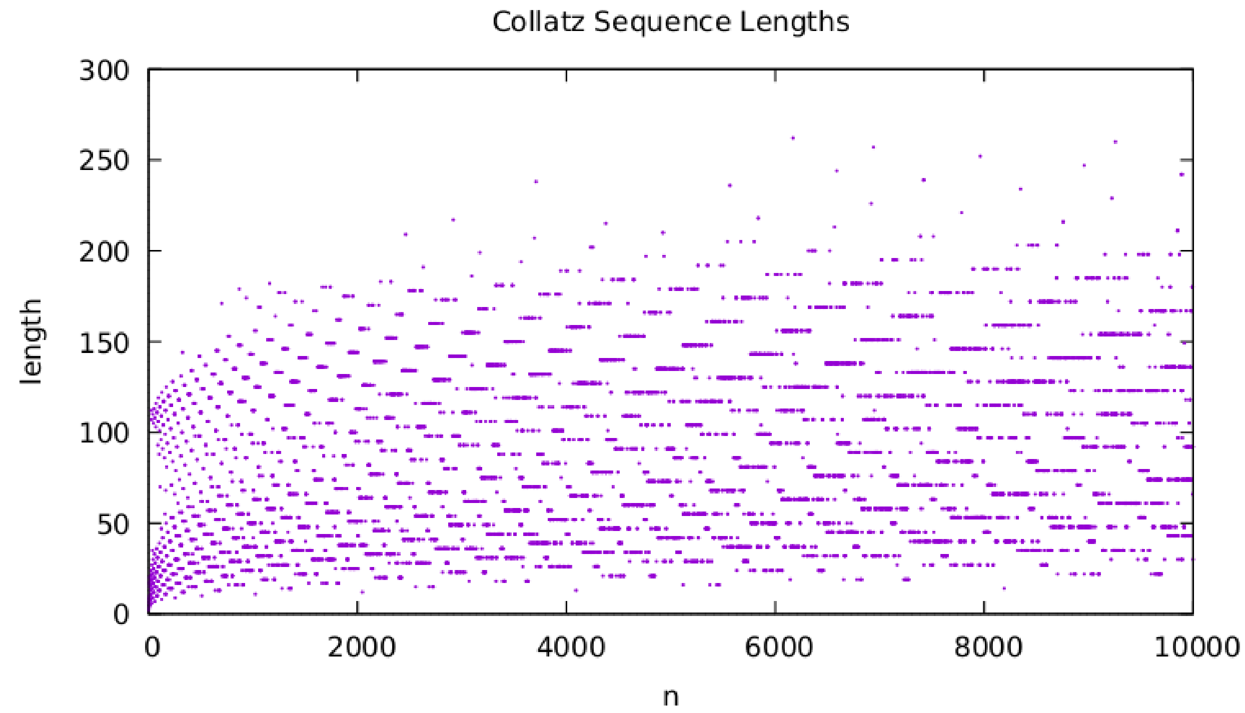

Figure 1.

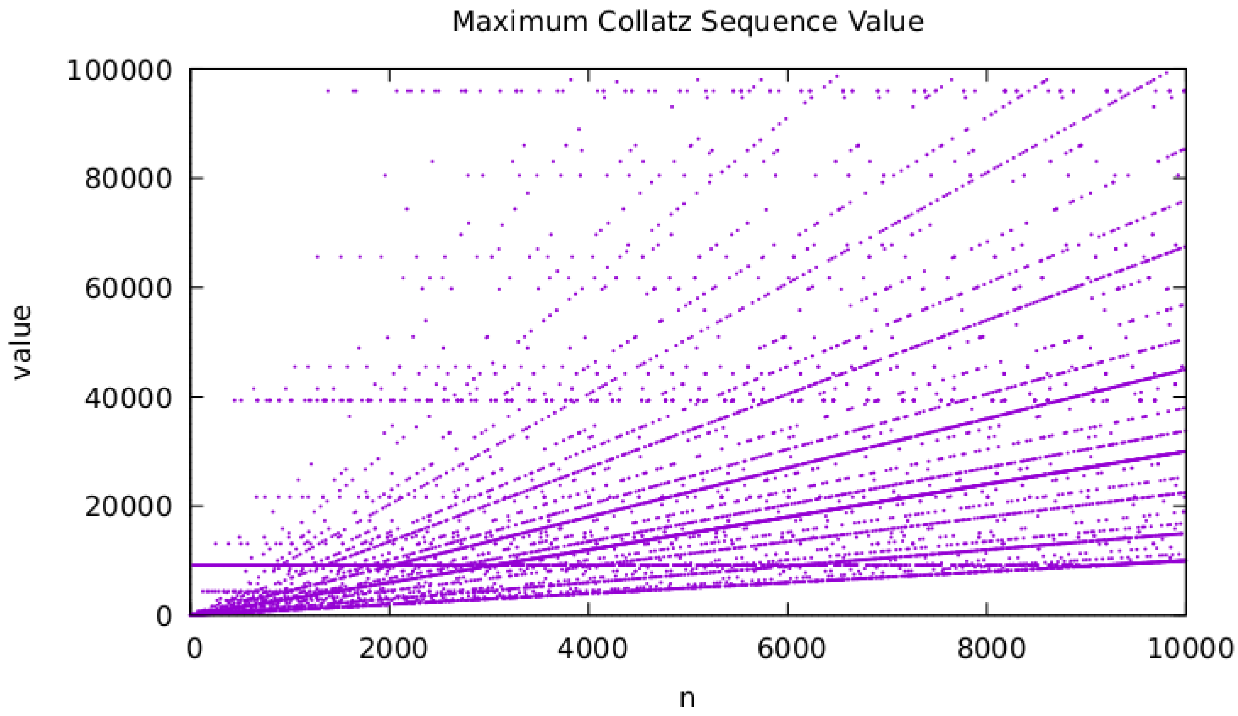

Figure 2.

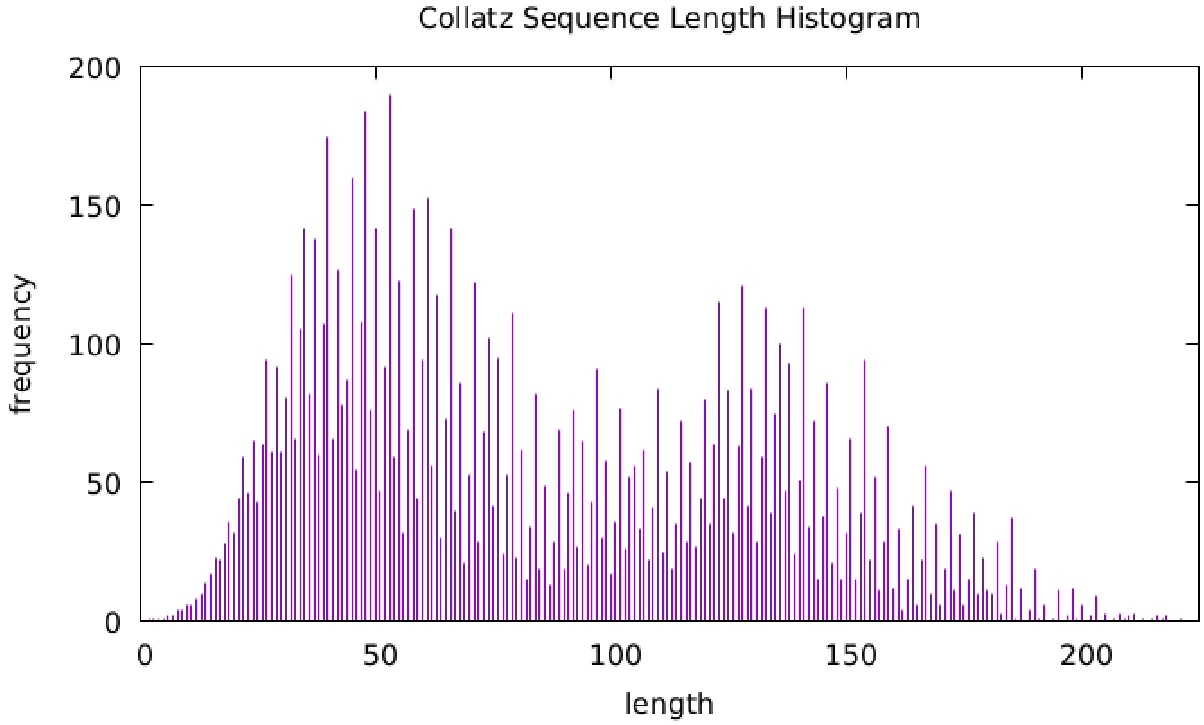

Figure 3.