1. Question one was more or less a hello world program that prints out “Hey there, cowboy!” Because it is less boring than hello world.

2. This program takes in an integer value through the command line and catches invalid input. Then, I ensure that the first few values of the fibonacci array are initialized correctly with a for loop, as the logic is difficult to implement otherwise. I then use another for loop to replace the numbers after index 2 with the true fibonacci values. Finally, I use a for loop and some printf logic to format it into a nice little right-aligned set of triangles.

3. This program takes in an integer value through the command line and initializes a boolean array of size n (the input value). I then initialize the first couple of values to false, because the logic is difficult to implement otherwise. I start a timer, call the Eratosthenes function, and then record the time it took for this algorithm to run. I then print out the time and the last five prime values with a clever while loop.

4. This program reads in bytes from a file and converts them from signed to unsigned. Then, I calculate the cumulative sum of the array with a for loop. After this, I apply a linear transformation to each of these numbers, start a timer to see how long it takes to print these items in seconds. They’re the prime numbers, AWESOME!

5. I will be honest, I understand that this program decrypts an affine cipher by applying the inverse operation to the cipher array, but I don’t fully understand how it works. I worked with Haley on this problem, and neither of us quite knew why 55 * I -165 is the inverse of 7 * I + 3. However, after this, I print out the results of the decryption to the terminal and am greeted with a lovely tale.