ASSIGNMENT 2 DESIGN DOCUMENT

Pehara Vidangamachchi

October 9th 2022

Ethan L. Miller

About:

The purpose of this program is to create a small numerical library alongside a test harness. The program will contain the implementation of sin, cos, arcsin, arccos and arctan without using any of the math functions within the C library.

Design Process for mathlib.c:

Sin Function:

This is a function that performs an operation similar to that of Sine. In order to run the program, I have to use my created math functions for power and factorial operations. Utilizing this I create a simple for loop that runs my sine function.

```
My sine function.
    Initialise all floats before running the for loop.
    Create a for loop that goes over the conditions for sine and then runs it.
        Math formula that calculates the sine function.
    Return the value from my function.
```

Cos Function

This is a function that performs an operation similar to that of Cosine. In order to execute my program, I use a similar function to my sine function and adjust the mathematical portion of it.

```
My cosine function.
    Initialise all floats before running the for loop.
    Create a for loop that goes over the conditions for cosine and then runs it.
        Math formula that calculates the cosine function.
    Return the value from my function.
```

Arcsin Function

This is a function that performs an operation similar to that of Arcsine. This is a similar function to the previous ones however for arc sine before running the for loop I have to run an if statement so that x doesn't go outside of its range.

```
My arcsine function.
    Initialise all floats before running the for loop and if statement.
    Create an if statement going over the range
        A print statement that lets the user know that x is outside the range.
    Create a for loop that goes over the conditions for cosine and then runs it.
        Math formula that calculates the cosine function.
    Return the value from my function.
```

Arcos Function

This is a function that performs an operation similar to that of Arccosine. Following the pseudo-code for arcsin, Arcos follows a similar setup. It diverts from previous versions by using a much more simplified formula that runs arcsin and pi.

```
My arccosine function.
    Initialise all floats before running the formula and if statement.
    Create an if statement going over the range
        A print statement that lets the user know that x is outside the range.
    Math formula that calculates the arccosine function.
    Return the value from my function.
```

Arctan Function

This is a function that performs an operation similar to that of Arctan. ARctan follows arcsine more closely than arccosine, their format remains the same and the only difference is the math formula.

```
My arctan function.
    Initialise all floats before running the for loop and if statement.
    Create an if statement going over the range
        A print statement that lets the user know that x is outside the range.
    Create a for loop that goes over the conditions for arctan and then runs it.
```

```
        Math formula that calculates the arctan function.
    Return the value from my function.
```

Log Function

This function runs my version of a Log function. For this, I use Newton-Raphsons method for optimizing f(exp) which gets me a simplified formula. This then allows me to create a for loop that runs through my log program.

```
My log function.
    Initialise all floats before running the for loop.
    Use $y = x$ for readability
    Create a for loop that goes over the conditions for the log and then runs it.
        Math formula that calculates the log function.
    Return the value from my function.
```

Factorial function

I created this function to replace a missing math component for the mathematical portion of each of my functions. This acts as a program that executes a process similar to having a factorial.

```
My factorial function
    Intialzing all floats before running my for loop.
        Math formula that calculates the factorial number.
    Return the value from this function.
```

Power Function

This is another function I created to replace its math counterpart of a power function. I follow a similar process to my last math function which creates a working power function.

```
My factorial function
    Intialzing all floats before running my for loop.
        Math formula that calculates the factorial number.
    Return the value from this function.
```

Exponential function

This was given through the assignment document, I simply implemented it into my code so I could calculate the exponential function.

The mathlib tests are a program that tests that all the math functions are functional.

Design process for mathlib-test:

Main Code:

In order to create my main program which iterates through my functions against the math.h function I had to create the main argument and declare it as a float, following the directions on the assignment to use argument value. Which was then followed by initializing more floats and creating a for loop and a while loop. I then utilized the same segment of code to create a running operation so that when compiled and executed it would return with my output and the math.h output.

#define my options h,a,s,c,S,C,T,l which are the variables I use to run each function.

The main float program executes my program.

Initializing the floats within my program

While loop that uses getopt and executes the command that allows the variables to work when typed in.

The same code is used for all of my math functions:

Declaring which case signifies which operation is run.

Check the results and print the following formatting that was required for the assignment.

Run a for loop that tests the given input and adds to it with enough iterations.

Use my function value and run my math function

Use the math function and run the test input.

Print the line of code given in the assignment which prints the values from both segments of code.

Break and continue to the next portion.

This repeats until all the functions have been executed and everything is returned in the end.