Ethan Henry
CSE 13S
Dr. Long
Jan. 15, 2020

Design.pdf for asgn2

Pre-Lab Part 1:

1. For n! <= abs(x^n)
   multiply x/n to x^(n-1)/(n-1)!
   add by total
2. printf("%lf", plug into Exp function)

Pre-Lab Part 2:

1. getopt() returns distinct entities to the shell script, which is then able to parse the information accordingly.
2. I'd say no, as the code isn't terribly hard to understand, and adding in enum or bool would just be extra work for little reward
3. main
   {
   Scan number entered and save it as num
   Switch (argument)
   Case argument
   {
   printf(Exp(num));
   }
   Case argument
   {
   printf(Cos(num));
   }
   Case argument
   {
   printf(Sin(num));
   }
   Case argument
   {
   printf(Tan(num));
   }
   Case argument
   {
   printf(All(num));
   }
   }

Description of Code:

- Start with the main() function, which takes in the arguments and puts it through a switch
    - Said switch calls different functions depending on argument
- Each function corresponds to a different mathematical function
- Exp() is calculated as stated above
    - For loop that calculates summation
    - Make sure between [0, 10]
    - Make sure step is 0.1
- Cos(), Sin(), and Tan() use the Pade formula
    - Use Horner standard form of Pade approximation to approximate
    - Make sure to restrict it to [-2pi, 2pi] for Sin and Cos
    - Make sure step size is pi/16
    - Make sure to restrict Tan to [-(pi/2 - .001), (pi/2 - .001)]
- All() calls upon all of the functions