Pipe Friction - The Friction Factor **oms7891**

Omar Mohamed Atia Mohamed Shehab

Professor Mohamed Eid

Computer Programming for Engineers

December 12, 2022

Step 1: Problem Identification and Statement:

- The goal of this program is to calculate the friction factor of fluid flow through pipes and tubes. The friction factor is dependent on the fluid properties, such as **density and viscosity**, as well as the characteristics of the pipe or tube, such as **its roughness and diameter**. The roughness of the pipe, for example, can affect the friction factor by **increasing the surface area** that the fluid must flow over. The friction factor is also influenced by the **Reynolds number**, which is a measure of the f**luid's inertia and viscous forces.** In general, a higher friction factor indicates a higher resistance to flow and a lower flow rate. Accurate determination of the friction factor is essential for designing efficient fluid systems.
- There are two equations that we use in order to calculate the friction factor, one is an equation that calculates the exact value of the friction factor **(colebrook equation)** and the other calculates an approximate value for the friction factor and Reynolds number is the common between both of them and it has to be greater than 4000 for the program to process properly.
- Bisection method is also one of the important functions that helps in calculating the zeros of a function based on the lower and upper bounds **(0.008, 0.08).** The bisection method involves iteratively narrowing down the range of possible values for f until a sufficiently accurate value is found. This method is considered to provide an approximate value for the friction factor.
- Then, we study the effects of the conduit diameter, the pipe roughness, the fluid density, the conduit roughness, and the dynamic viscosity on the friction factor. This is done by utilizing the bisection method instead of getting the value from the user.

Step 2: Gathering Information:

- **Colebrook equation(1):**

The Colebrook equation is a mathematical equation that is used to determine the friction factor in fluid dynamics. It is given by the following expression:

$$\frac{1}{\sqrt{f}} = -2log_{10}\left(\frac{\varepsilon}{3.7D_h} + \frac{2.51}{Re\sqrt{f}}\right)$$

Where

f = friction factor,

e = the roughness (m),

D = conduit diameter (m),

and Re = the **Reynolds number**, defined by the following equation

$$\mathrm{Re} = \frac{\rho VD}{\mu}$$

Where

p = the fluid's density [kg/m3],

V = its velocity [m/s], and

μ = dynamic viscosity [N s/m2].

Those variables have to be chosen carefully in a specific range in order to be able to get correct results. The range of the roughness is 0.0001 to 3. The range of the fluid density is 0.5 to 2000. The range of the dynamic viscosity is 10^-6 to 300 Also, the Re should be greater than 4000 for the application to run.

- **The Bisection Method for Finding Roots:**

Finding the value of friction factor (f) from Equation **(1)** using analytical means is far from trivial. Numerical methods can provide very good results by using a root finding method, that is, finding the root of the function **g(f)** defined as

$$g(f) = \frac{1}{\sqrt{f}} + 2\log\left(\frac{\varepsilon}{3.7D} + \frac{2.51}{Re\sqrt{f}}\right)$$

The bisection method is a numerical method for finding the root of an equation. It is a simple and robust method that can be used to find approximate solutions to equations that are difficult to solve analytically. The bisection method works by iteratively narrowing down the range of possible values for the root until a sufficiently accurate value is found. To use the bisection method, the equation to be solved must be written in the form f(x) = 0, where f(x) is a continuous function and x is the variable to be solved for. The initial range of possible values for x is then divided in half, and the sign of the function at the two ends of the range is compared. If the function has opposite signs at the two ends of the range, then the root must lie within that range and the bisection method can be applied. The midpoint of the range is then calculated, and the sign of the function at the midpoint is compared to the sign at one of the ends of the range. The range is then narrowed down to the half that contains the root, and the process is repeated until a sufficiently accurate value for the root is found. If we supposed that the lower bound is xl and the upper bound is xu and the mid point of those values is xm

Then xm = (xl + xu)/2

Then we check if **f(xl)*f(xu) is greater than zero**

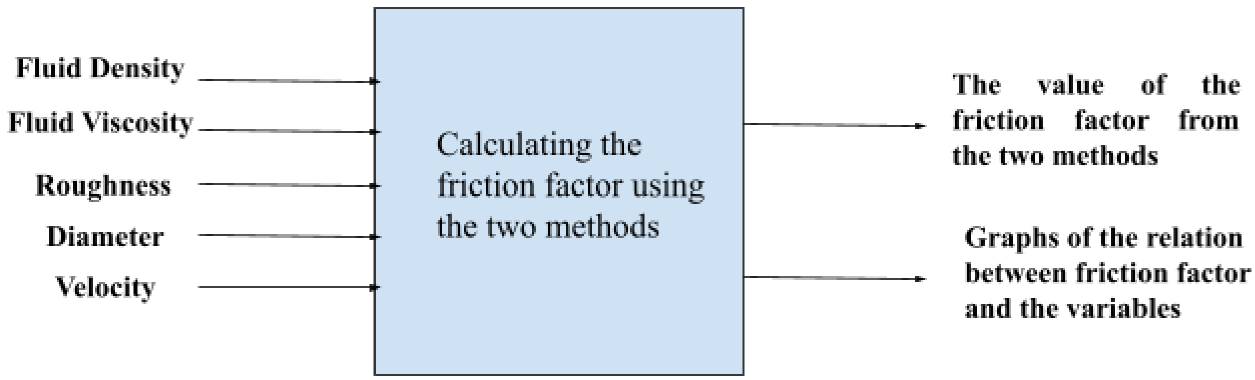
Then we check if **f(xl)*f(xm) < 0**, we will assign **xu = xr**

Then we check if **f(xl)*f(xm) > 0,** we will assign **xl = xr**

Then we check if **|f(xl)*f(xm)| < 0.01**, then the **root will be equal to xr** and the **loop will terminate**.

- **I/O Menu:**

  In the program there is a menu that is displayed when we run the program and it has many operations that include calculating the roots of the bisection method and the analytical friction factor. Then we get graphs representing the relation between the friction factor and the fluid density, fluid viscosity, roughness, and diameter.



Step 3: Test cases and Algorithms:

A. Test cases:

- **Option 1:**

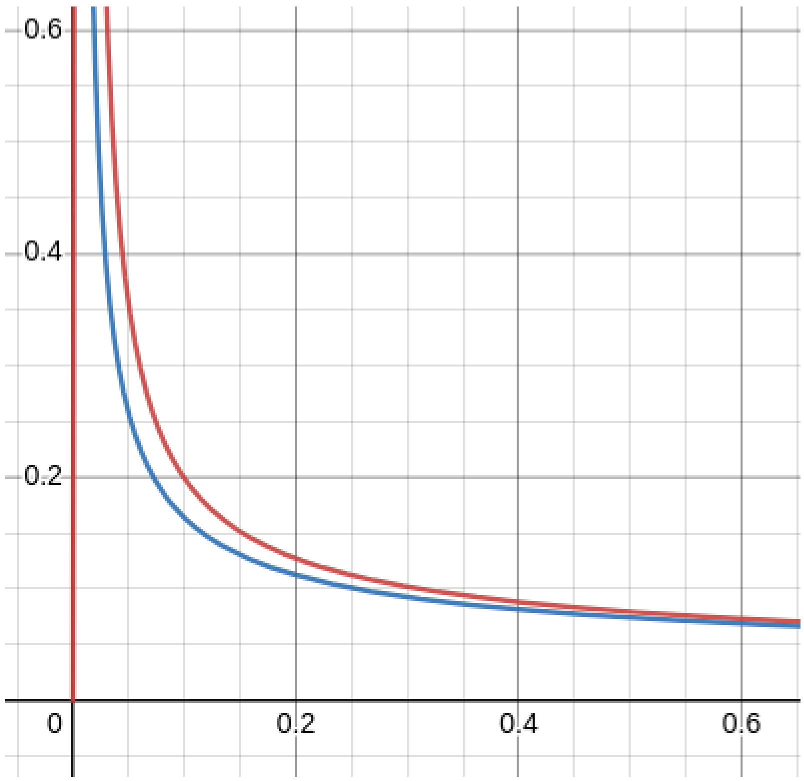
Fluid density	Fluid viscosity	Velocity	Diameter	Epsilon	Analytical friction factor	Roots of numerical function
1000	1	2	5	0.01	0.030965	0.030887

- **Option 2:**

Fluid density	Fluid viscosity	Velocity	Epsilon	Diameter	Analytical friction factor	Roots of numerical function
1000	1	2	0.01	1 2 3 4	0.0486 0.0383 0.0338 0.031	0.0473 0.0376 0.0333 0.0306

				5 6 7 8 9 10	0.0291 0.0277 0.0265 0.0256 0.0248 0.0242	0.0288 0.0275 0.0263 0.0254 0.0247 0.024

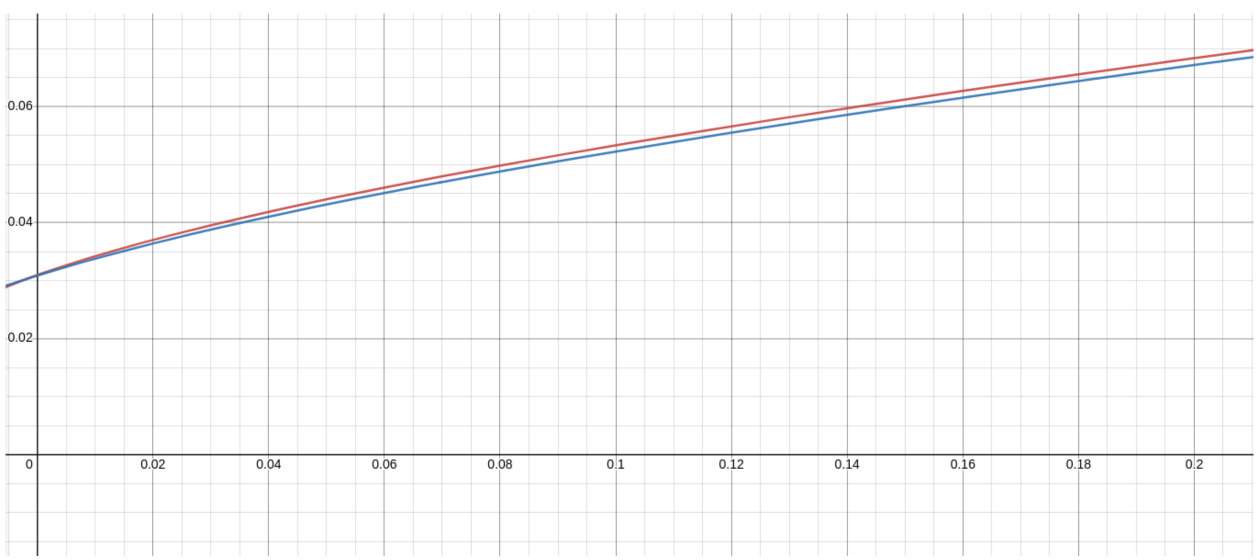
Expected graph:



- **Option 3:**

Fluid density	Fluid viscosity	Velocity	Diameter	Epsilon	Analytical friction factor	Roots of numerical function
1000	1	2	5	0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08 0.09	0.0342 0.037 0.0395 0.0419 0.044 0.0461 0.048 0.0498 0.0516	0.0338 0.0364 0.0388 0.041 0.0431 0.0451 0.047 0.0489 0.0506

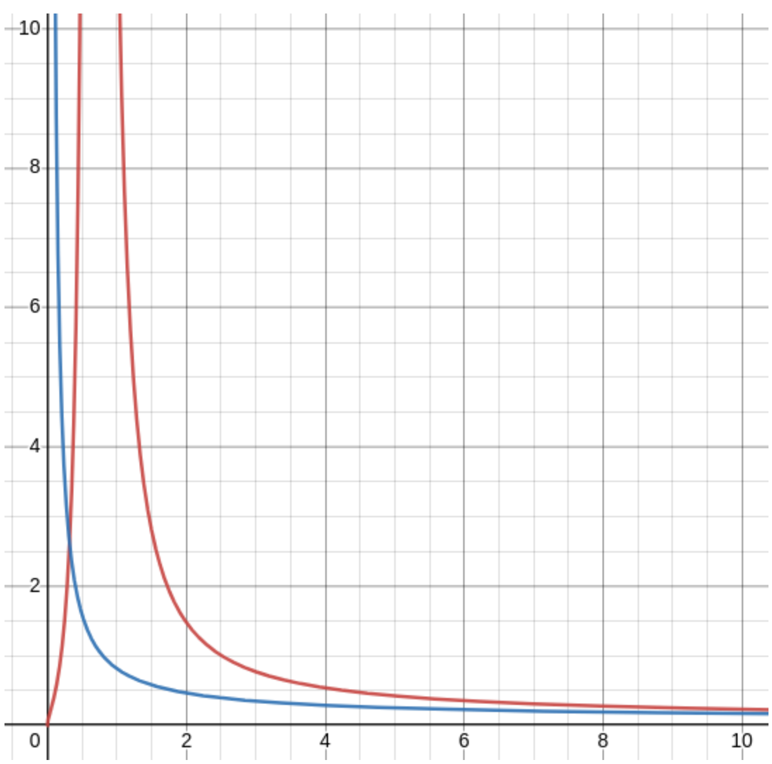
Expected graph:



- **Option 4:**

Fluid viscosity	**Velocity**	**Diameter**	**Epsilon**	**Fluid density**	**Analytical friction factor**	**Roots of numerical function**
1	2	5	0.01	1000 1100 1200 1300 1400 1500 1600 1700 1800 1900 2000	0.031 0.0302 0.0295 0.0288 0.0283 0.0278 0.0273 0.0269 0.0265 0.0261 0.0258	0.0309 0.0301 0.0294 0.0288 0.0283 0.0278 0.0274 0.0269 0.0266 0.0262 0.0259

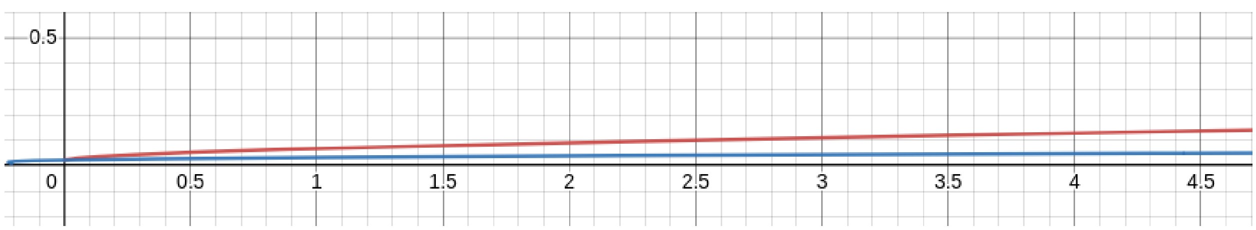
Expected graph



- **Option 5:**

Fluid density	Velocity	Diameter	Epsilon	Fluid viscosity	Analytical friction factor	Roots of numerical function
1000	2	5	0.01	0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9	0.0179 0.0208 0.0228 0.0244 0.0258 0.027 0.0281 0.0291 0.0301	0.018 0.0209 0.0229 0.0245 0.0259 0.0271 0.0281 0.0291 0.03

Expected graph:



B. Algorithms:

Pseudo code of NumericalF in the header file:

```
Declare function(numnericalF) = NumericalF with arguments Re,
diameter, roughness,frictionfactor
     Assign numnericalF to ((1./sqrt(frictionfactor)) +
2.*log10(roughness./(3.7.*diameter)+
2.51./(Re.*sqrt(frictionfactor))));
End
```

Pseudo code of analyticalF in the header file:

```
Declare function(AnalyticalF) = analyticalF with arguments r,
Re, diameter
     Assign AnalyticalF to
1.325./(log((r/(3.7.*diameter))+(5.74./(Re.^0.9)))).^2;
```

```
end
```

Pseudo code of bisectionCalc in the header file:

```
Declare function (bisection) = bisectionCalc with arguments
lowerBound,upperBound,Re,D,r
    Assign mid to (lowerBound + upperBound)/2;
    while abs(NumericalF(Re, D, r,lowerBound).*NumericalF(Re, D,
r,mid)) >= 0.00001 Repeat
        if (NumericalF(Re, D, r,lowerBound)*NumericalF(Re, D,
r,mid)) < 0
            Assign upperBound to mid;
        else
            Assign lowerBound to mid;
        end
        Assign mid to (lowerBound + upperBound)/2;
    end
    Assign bisection to mid;
end
```

Pseudo code of main file:

```
Assign check to true
While check is true
     Print "Choose a number: "
     Print "1- Displays the numerical and analytical values for
the friction factor."
     Print "2- Conduct the analysis of how the friction factor
varies with the conduit diameter."
     Print "3- Conduct the analysis of how the friction factor
varies with the pipe roughness."
     Print "4- Conduct the analysis of how the friction factor
varies with the fluid density."
     Print "5- Conduct the analysis of how the friction factor
varies with the dynamic viscosity."
```

```
    Print "6- Exit"
    Print “Enter an option: ” then read number into choice
     If choice equals 1
          Print “please enter the fluid density between 0.5 to
2000: ” then read number into fluidDensity
     While (fluidDensity < 0.5 or fluidDensity > 2000)
          Print "Invalid, please enter the fluid density between
          0.5 to 2000: " and read number into  fluidDensity
     End
          Print “please enter the fluid viscosity between 0.5 to
     2000: ” then read number into fluidVisco
     While (fluidVisco < 0.5 or fluidVisco > 2000)
          Print "Invalid, please enter the fluid viscosity
          between 0.5 to 2000: " and read number into
          fluidVisco
     End
          Print “Please enter the diameter” and read value into
     D
          Print “Please enter the velocity” and read value into
     V
          Print “enter epsilon between 0.0001 to 3: ” and read
     value into epsilon
          Assign epsilon to epsilon/1000
          While (epsilon < 1e-6 or epsilon > 0.003) repeat
               Print "Invalid, please enter epsilon between
          0.0001 to 3: " and read number into  fluidVisco
          End
          Assign Re to (fluidDensity*V*D)/fluidVisco;
          if (Re <= 4000)
               Print “Re should be greater than 4000” newline
               Break
          End
```

```
        Assign f0 to analyticalF(epsilon,Re,D);
        Assign root to bisectionCalc
    (0.008,0.08,Re,D,epsilon);
        Print “analytical friction: “ float value f0 newline
        Print “bisection: “ float value root newline
    Elseif choice equals 2
        Print “please enter the fluid density between 0.5 to
2000: ” then read number into fluidDensity
        While (fluidDensity < 0.5 or fluidDensity > 2000)
            Print "Invalid, please enter the fluid density
        between 0.5 to 2000: " and read number into
        fluidDensity
        End
        Print “please enter the fluid viscosity between 0.5 to
2000: ” then read number into fluidVisco
        While (fluidVisco < 0.5 or fluidVisco > 2000)
            Print "Invalid, please enter the fluid viscosity
        between 0.5 to 2000: " and read number into
        fluidVisco
        End
        Print “Please enter the velocity” and read value into
V
        Print “enter epsilon between 0.0001 to 3: ” and read
    value into epsilon
        Assign epsilon to epsilon/1000
        While (epsilon < 1e-6 or epsilon > 0.003) repeat
            Print "Invalid, please enter epsilon between
        0.0001 to 3: " and read number into  fluidVisco
        End
        Print “enter the lower value for the conduit diameter:
" and read value into lower
```

```
        Print "enter the upper value for the conduit diameter: " and read value into upper
    Print "enter the size value for the conduit diameter: " and read value into size
        Assign D as array of from lower to upper and the values are increased by size within the array
        Declare Analytical_F as array of zeros of size length(D)*1
        Declare f as array of zeros of size length(D)*1
        For i equal 1 to length of D
            Assign Re to (fluidDensity*V*D(i))/fluidVisco;
            Assign Analytical_F(i) equals analyticalF(epsilon,Re,D(i))
            Assign f(i) to bisectionCalc(0.008,0.08,Re,D(i),epsilon)
        End
        if (Re <= 4000)
            Print "Re should be greater than 4000, try again"
        newline
            Break
        End
        plot(D, f, 'b')
        xlabel is Conduit Diameter (D)
        ylabel is Friction Factor
        Title is Friction factor vs. Conduit Diameter
        Hold on
        plot(D, Analytical_F, '--r')
        legend('Numerical' , 'Analytical')
        Hold off
        Save image as 'FrictionFactor vs Conduit Diameter.png'
```

Elseif choice equals 3

```
        Print "please enter the fluid density between 0.5 to 2000: " then read number into fluidDensity
        While (fluidDensity < 0.5 or fluidDensity > 2000)
            Print "Invalid, please enter the fluid density between 0.5 to 2000: " and read number into fluidDensity
        End
        Print "please enter the fluid viscosity between 0.5 to 2000: " then read number into fluidVisco
        While (fluidVisco < 0.5 or fluidVisco > 2000)
            Print "Invalid, please enter the fluid viscosity between 0.5 to 2000: " and read number into fluidVisco
        End
        Print "Please enter the diameter" and read value into D
        Print "Please enter the velocity" and read value into V
        Print "enter the lower value for the Roughness: " and read value into lower
        Print "enter the upper value for the Roughness: " and read value into upper
    Print "enter the size value for the Roughness: " and read value into size
        Assign epsilon as array of from lower to upper and the values are increased by size within the array
        Declare Analytical_F as array of zeros of size length(epsilon)*1
        Declare R as array of zeros of size length(epsilon)*1
        Assign Re to (fluidDensity*V*D)/fluidVisco;
        For i equal 1 to length of epsilon
```

```
          Assign Analytical_F(i) equals
analyticalF(epsilon(i),Re,D)
          Assign R(i) to
bisectionCalc(0.008,0.08,Re,D,epsilon(i))
     End
     if (Re <= 4000)
          Print "Re should be greater than 4000, try again"
     newline
          Break
     End
     plot(epsilon, R, 'b')
     xlabel is Roughness (R)
     ylabel is Friction Factor
     Title is Friction factor vs. Roughness
     Hold on
     plot(epsilon, Analytical_F, '--r')
     legend('Numerical' , 'Analytical')
     Hold off
     Save image as 'FrictionFactor vs Roughness.png'
Elseif choice equals 4
     Print "please enter the fluid viscosity between 0.5 to
2000: " then read number into fluidVisco
     While (fluidVisco < 0.5 or fluidVisco > 2000)
          Print "Invalid, please enter the fluid viscosity
     between 0.5 to 2000: " and read number into
     fluidVisco
     End
     Print "Please enter the diameter" and read value into
D
     Print "Please enter the velocity" and read value into
V
```

```
          Print “enter epsilon between 0.0001 to 3: ” and read
value into epsilon
          Assign epsilon to epsilon/1000
          While (epsilon < 1e-6 or epsilon > 0.003) repeat
               Print "Invalid, please enter epsilon between
          0.0001 to 3: " and read number into  fluidVisco
          End
          Print “enter the lower value for the fluid density: "
and read value into lower
          Print “enter the upper value for the fluid density: "
and read value into upper
      Print “enter the size value for the fluid density: " and
read value into size

          Assign fd as array of from lower to upper and the
values are increased by size within the array
          Declare Analytical_F as array of zeros of size
     numel(fd)*1
          Declare R as array of zeros of size length(fd)*1
          For i equal 1 to length of fd
               Assign Re to fd(i).*V.*D./fluidVisco
               Assign Analytical_F(i) equals
analyticalF(epsilon,Re,D)
               Assign R(i) to
bisectionCalc(0.008,0.08,Re,D,epsilon)
          End
          if (Re <= 4000)
               Print “Re should be greater than 4000, try again”
          newline
               Break
          End
          plot(fd, R,'b')
```

```
        Xlabel is 'Fluid Density'
        Ylabel is 'Friction Factor'
        Title is 'Friction Factor vs. Fluid Density'
        hold on;
        plot(fd, Analytical_F,'--r')
        legend('Numerical', 'Analytical')
        hold off;
      Save image as 'FrictionFactor vs Fluid Density.png'
    Elseif choice equals 5
        Print “please enter the fluid density between 0.5 to
2000: ” then read number into fluidDensity
        While (fluidDensity < 0.5 or fluidDensity > 2000)
            Print "Invalid, please enter the fluid density
        between 0.5 to 2000: " and read number into
        fluidDensity
        End
        Print “Please enter the diameter” and read value into
    D
        Print “Please enter the velocity” and read value into
    V
        Print “enter epsilon between 0.0001 to 3: ” and read
    value into epsilon
        Assign epsilon to epsilon/1000
        While (epsilon < 1e-6 or epsilon > 0.003) repeat
            Print "Invalid, please enter epsilon between
        0.0001 to 3: " and read number into  fluidVisco
        End
        Print “enter the lower value for the fluid viscosity:
    " and read value into lower
        Print “enter the upper value for the fluid viscosity:
" and read value into upper
```

```
          Print "enter the size value for the fluid viscosity: "
and read value into size
          Assign fv as array of from lower to upper and the
values are increased by size within the array
          Declare Analytical_F as array of zeros of size
     numel(fv)*1
          Declare v as array of zeros of size numel(fv)*1
          Assign Re to (fluidDensity*V*D)/fluidVisco;
          For i equal 1 to length of fv
               Assign Re to fluidDensity.*V.*D./fv(i)
               Assign Analytical_F(i) equals
          analyticalF(epsilon,Re,D)
               Assign v(i) to
          bisectionCalc(0.008,0.08,Re,D,epsilon)
          End
          if (Re <= 4000)
               Print "Re should be greater than 4000" newline
               Break
          End
          plot(fv, v,'b');
          hold on;
          plot(fv, Analytical_F,'--r');
          Xlabel is 'Fluid Density'
          Ylabel is 'Friction Factor'
          Title is 'Friction Factor vs. Fluid Density'
          legend('Numerical', 'Analytical')
          hold off;
          Save image as 'FrictionFactor vs Fluid Density.png'
     elseif choice equals 6
                  Assign check to false
     else
        Print "Not a valid number please try again: "
```

```
    end
End
```

Step 4: Code

```
function [numnericalF] = NumericalF(Re, diameter, roughness,frictionfactor)
      numnericalF = ((1./sqrt(frictionfactor)) +
2.*log10(roughness./(3.7.*diameter) + 2.51./(Re.*sqrt(frictionfactor))));
end

function [AnalyticalF] = analyticalF(r,Re,diameter)
   AnalyticalF = 1.325./(log((r/(3.7.*diameter))+(5.74./(Re.^0.9)))).^2;
end

function [bisection] = bisectionCalc(lowerBound,upperBound,Re,D,r)
   mid = (lowerBound + upperBound)/2;
     while  abs(NumericalF(Re,  D,  r,lowerBound).*NumericalF(Re,  D,  r,mid))  >=
0.00001

       if (NumericalF(Re, D, r,lowerBound)*NumericalF(Re, D, r,mid)) < 0
           upperBound = mid;
       else
           lowerBound = mid;
       end
       mid = (lowerBound + upperBound)/2;
   end
   bisection = mid;
end


%--------------------------
% Omar Mohamed Atia Shehab
% Date:   December 12, 2022
%        Assignment4
% Pipe Friction - The Friction Factor
check = true;
while check
   disp("Choose a number: ");
```

```
    disp("1- Displays the numerical and analytical values for the friction factor.");
    disp("2- Conduct the analysis of how the friction factor varies with the conduit diameter.");
    disp("3- Conduct the analysis of how the friction factor varies with the pipe roughness.");
    disp("4- Conduct the analysis of how the friction factor varies with the fluid density.");
    disp("5- Conduct the analysis of how the friction factor varies with the dynamic viscosity.");
    disp("6- Exit");
    choice = input("Enter an option: ");
    if choice == 1
        fluidDensity = input("please enter the fluid density between 0.5 to 2000: ");
        while (fluidDensity < 0.5 || fluidDensity > 2000)
                fluidDensity = input("Invalid, please enter the fluid density between 0.5 to 2000: ");
        end
        fluidVisco = input("please enter the fluid viscosity between 10^-6 to 300: ");
        while (fluidVisco < 1e-6 || fluidVisco > 300)
                fluidVisco = input("Invalid, please enter the fluid viscosity between 10^-6 to 300: ");
        end
        D = input("please enter the diameter: ");
        V = input("please enter the velocity: ");
        epsilon = input("enter epsilon between 0.0001 to 3: ");
        epsilon = epsilon/1000;
        while (epsilon < 1e-6 || epsilon > 0.003)
            epsilon = input("invalid, enter epsilon between 0.0001 to 3: ");
        end
        Re = (fluidDensity*V*D)/fluidVisco;
        if (Re <= 4000)
            fprintf("Re should be greater than 4000 \n");
            break;
        end
        f0 = analyticalF(epsilon,Re,D);
```

```
        root = bisectionCalc(0.008,0.08,Re,D,epsilon);
        fprintf("analytical friction: %f\n",f0);
        fprintf("bisection: %f\n",root);
    elseif choice == 2
         fluidDensity = input("please enter the fluid density between 0.5 to 
2000: ");
        while (fluidDensity < 0.5 || fluidDensity > 2000)
                fluidDensity = input("Invalid, please enter the fluid density 
between 0.5 to 2000: ");
        end
        fluidVisco = input("please enter the fluid viscosity: ");
        while (fluidVisco < 1e-6 || fluidVisco > 300)
                fluidVisco = input("Invalid, please enter the fluid viscosity 
between 10^-6 to 300: ");
        end
        V = input("please enter the velocity: ");
        epsilon = input("enter epsilon between 0.0001 to 3: ");
        epsilon = epsilon/1000;
        while (epsilon < 1e-6 || epsilon > 0.003)
            epsilon = input("invalid, enter epsilon between 0.0001 to 3: ");
        end
        lower = input("enter the lower value for the conduit diameter: ");
        upper = input("enter the upper value for the conduit diameter: ");
        size = input("enter the size value for the conduit diameter: ");
        D = lower:size:upper;
        Analytical_F = zeros(length(D),1);
        f = zeros(length(D),1);
        for i=1:length(D)
            Re= (fluidDensity.*V.*D(i))./fluidVisco;
            Analytical_F(i) = analyticalF(epsilon,Re,D(i));
            f(i) = bisectionCalc(0.008,0.08,Re,D(i),epsilon);
        end
        if (Re <= 4000)
            fprintf('Re should be greater than 4000, try again \n');
            break;
        end
        plot(D, f,'b');
        xlabel('Conduit Diameter (D)');
```

```
        ylabel('Friction Factor');
        title('Friction Factor vs. Conduit Diameter');
        hold on;
        plot(D, Analytical_F,'--r');
        legend('Numerical', 'Analytical');
        hold off;
        saveas(gcf, 'FrictionFactor vs Conduit Diameter.png');
    elseif choice == 3
         fluidDensity = input("please enter the fluid density between 0.5 to 
2000: ");
        while (fluidDensity < 0.5 || fluidDensity > 2000)
                fluidDensity = input("Invalid, please enter the fluid density 
between 0.5 to 2000: ");
        end
         fluidVisco = input("please enter the fluid viscosity between 10^-6 to 
300: ");
        while (fluidVisco < 1e-6 || fluidVisco > 300)
                fluidVisco = input("Invalid, please enter the fluid viscosity 
between 10^-6 to 300: ");
        end
        D = input("please enter the diameter: ");
        V = input("please enter the velocity: ");
        lower = input("enter the lower value for the Roughness: ");
        upper = input("enter the upper value for the Roughness: ");
        size = input("enter the size value for the Roughness: ");
        epsilon = lower:size:upper;
        Analytical_F = zeros(numel(epsilon),1);
        R = zeros(numel(epsilon),1);
        Re = fluidDensity.*V.*D./fluidVisco;

        for i=1:numel(epsilon)
            Analytical_F(i) = analyticalF(epsilon(i),Re,D);
            R(i) = bisectionCalc(0.008,0.08,Re,D,epsilon(i));
        end
        if (Re <= 4000)
            fprintf('Re should be greater than 4000, try again \n');
            break;
        end
```

```
        plot(epsilon, R,'b');
        xlabel('Roughness (R)');
        ylabel('Friction Factor');
        title('Friction Factor vs. Roughness');
        hold on;
        plot(epsilon, Analytical_F,'--r');
        legend('Numerical','Analytical');
        hold off;
        saveas(gcf, 'FrictionFactor vs Roughness.png');
    elseif choice == 4
         fluidVisco = input("please enter the fluid viscosity between 10^-6 to
300: ");
        while (fluidVisco < 1e-6 || fluidVisco > 300)
                fluidVisco = input("Invalid, please enter the fluid viscosity
between 10^-6 to 300: ");
        end
        D = input("please enter the diameter: ");
        V = input("please enter the velocity: ");
        epsilon = input("enter epsilon between 0.0001 to 3: ");
        epsilon = epsilon/1000;
        while (epsilon < 1e-6 || epsilon > 0.003)
            epsilon = input("invalid, enter epsilon between 0.0001 to 3: ");
        end
        lower = input("enter the lower value for the fluid density: ");
        upper = input("enter the upper value for the fluid density: ");
        size = input("enter the size value for the fluid density: ");
        fd = lower:size:upper;
        Analytical_F = zeros(numel(fd),1);
        R = zeros(numel(fd),1);
        for i = 1:numel(fd)
            Re = fd(i).*V.*D./fluidVisco;
            Analytical_F(i) = analyticalF(epsilon, Re, D);
            R(i) = bisectionCalc(0.008, 0.08, Re, D, epsilon);
        end
        if (Re <= 4000)
            fprintf('Re should be greater than 4000, try again \n');
            break;
        end
```

```
        plot(fd, R,'b');
        xlabel('Fluid Density');
        ylabel('Friction Factor');
        title('Friction Factor vs. Fluid Density');
        hold on;
        plot(fd, Analytical_F,'--r');
        legend('Numerical', 'Analytical');
        hold off;
        saveas(gcf, 'FrictionFactor vs Fluid Density.png');
    elseif choice == 5
         fluidDensity = input("please enter the fluid density between 0.5 to
2000: ");
        while (fluidDensity < 0.5 || fluidDensity > 2000)
                fluidDensity = input("Invalid, please enter the fluid density
between 0.5 to 2000: ");
        end
        D = input("please enter the diameter: ");
        V = input("please enter the velocity: ");
        epsilon = input("enter epsilon between 0.0001 to 3: ");
        epsilon = epsilon/1000;
        while (epsilon < 1e-6 || epsilon > 0.003)
            epsilon = input("invalid, enter epsilon between 0.0001 to 3: ");
        end
        lower = input("enter the lower value for the fluid viscosity: ");
        upper = input("enter the upper value for the fluid viscosity: ");
        size = input("enter the size value for the fluid viscosity: ");
        fv = lower:size:upper;
        Analytical_F = zeros(numel(fv),1);
        v = zeros(numel(fv),1);
        for i = 1:numel(fv)
            Re = fluidDensity.*V.*D./fv(i);
            Analytical_F(i) = analyticalF(epsilon, Re, D);
            v(i) = bisectionCalc(0.008, 0.08, Re, D, epsilon);
        end
        if (Re <= 4000)
            fprintf('Re should be greater than 4000, try again \n');
            break;
        end
```

```
        plot(fv, v,'b');
        hold on;
        plot(fv, Analytical_F,'--r');
        xlabel('Fluid Density');
        ylabel('Friction Factor');
        title('Friction Factor vs. Fluid Density');
        legend('Numerical', 'Analytical');
        hold off;
        saveas(gcf, 'FrictionFactor vs Fluid Density.png');
    elseif choice == 6
        check = false;
    else
        disp("Not a valid number please try again: ");
    end
End
```

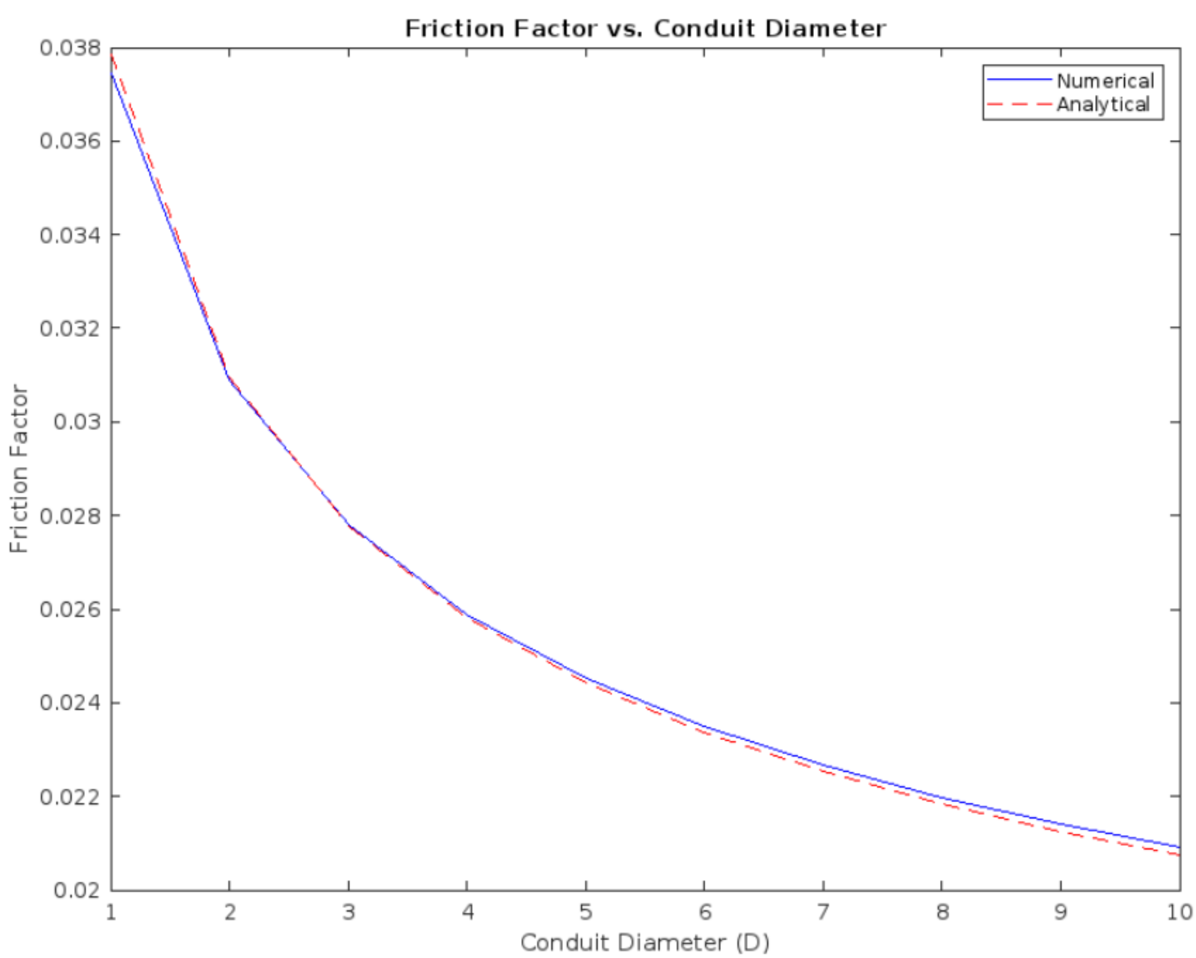
Step 5: Test and verification:

- **Test case 1:**

```
Choose a number:
1- Displays the numerical and analytical values for the friction factor.
2- Conduct the analysis of how the friction factor varies with the conduit diameter.
3- Conduct the analysis of how the friction factor varies with the pipe roughness.
4- Conduct the analysis of how the friction factor varies with the fluid density.
5- Conduct the analysis of how the friction factor varies with the dynamic viscosity.
6- Exit
Enter an option:
1
please enter the fluid density between 0.5 to 2000:
1000
please enter the fluid viscosity between 10^-6 to 300:
1
please enter the diameter:
5
please enter the velocity:
4
enter epsilon between 0.0001 to 3:
0.01
analytical friction: 0.025810
```

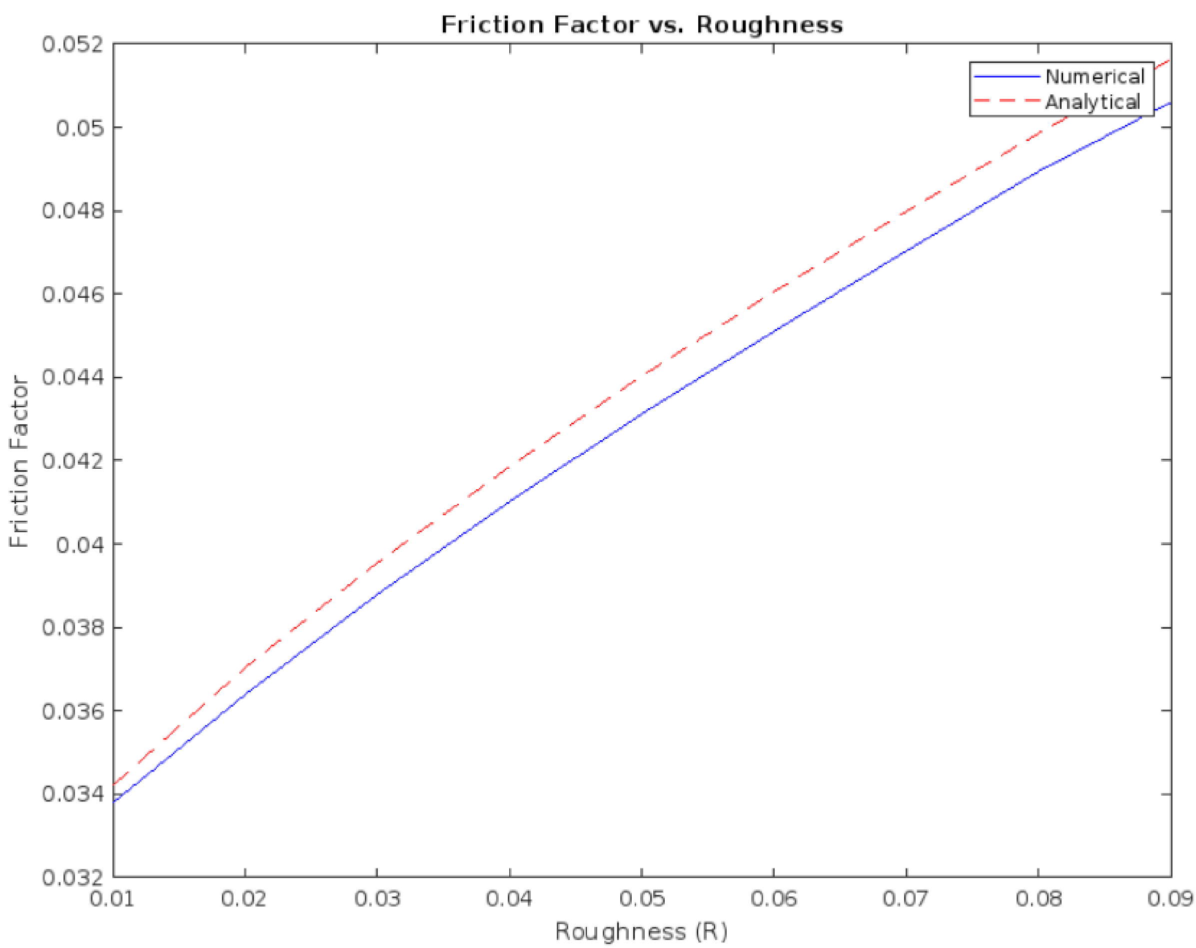
- **Test case 2:**

```
Enter an option:
2
please enter the fluid density between 0.5 to 2000:
1000
please enter the fluid viscosity:
1
please enter the velocity:
5
enter epsilon between 0.0001 to 3:
0.01
enter the lower value for the conduit diameter:
1
enter the upper value for the conduit diameter:
10
enter the size value for the conduit diameter:
1
```



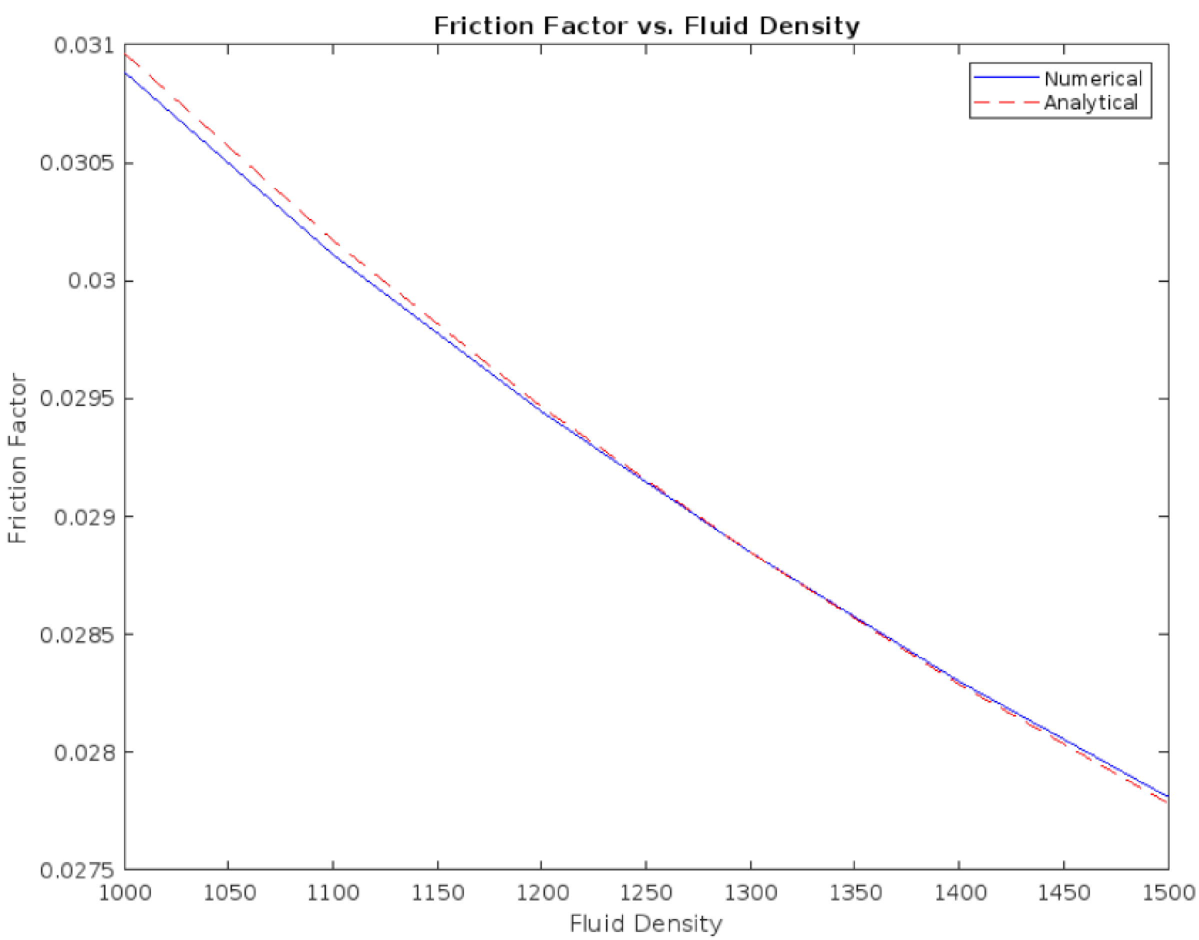
- **Test case 3:**

```
Enter an option:
>> 3
please enter the fluid density between 0.5 to 2000:
>> 1000
please enter the fluid viscosity between 10^-6 to 300:
>> 1
please enter the diameter:
>> 5
please enter the velocity:
>> 2
enter the lower value for the Roughness:
>> 0.01
enter the upper value for the Roughness:
>> 0.09
enter the size value for the Roughness:
>> 0.01
```



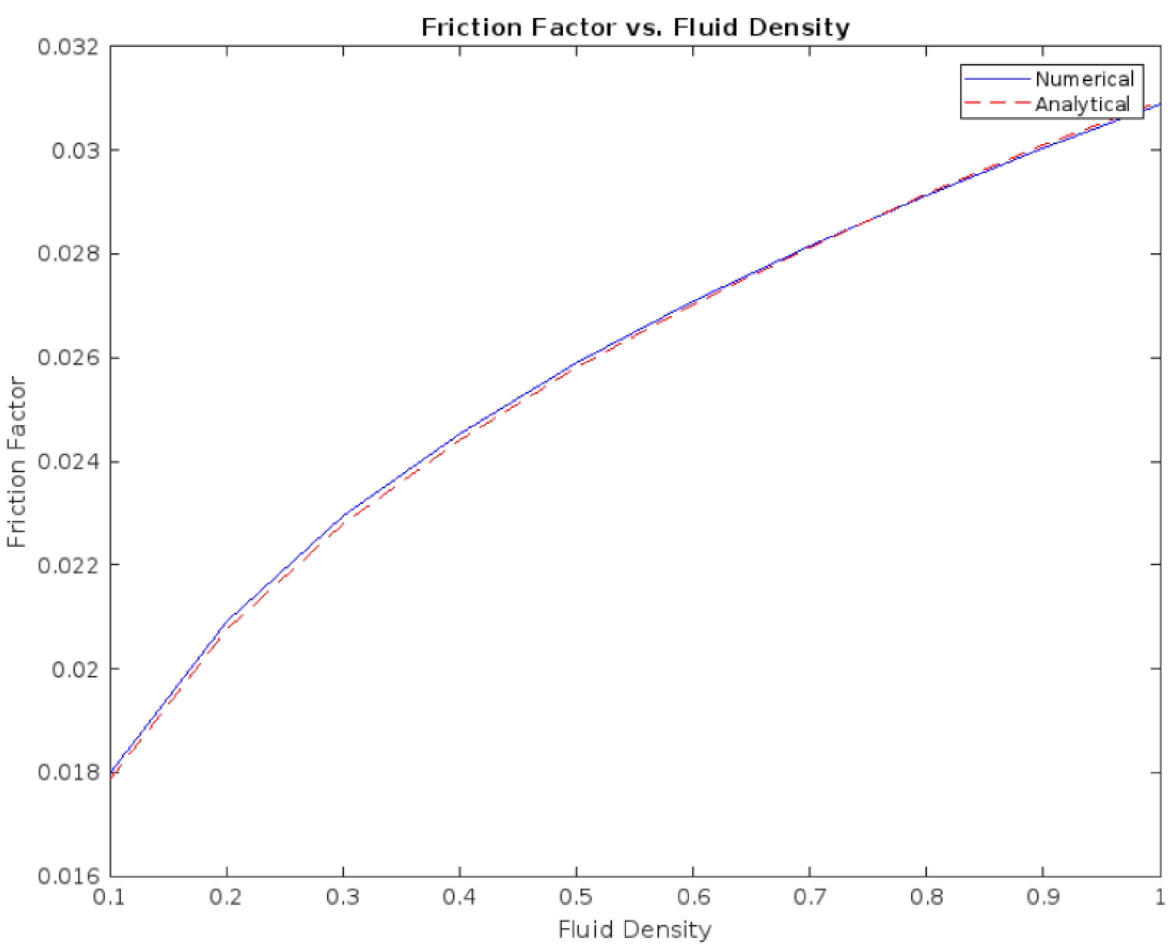
- **Test case 4:**

```
Enter an option:
>> 4
please enter the fluid viscosity between 10^-6 to 300:
>> 1
please enter the diameter:
>> 5
please enter the velocity:
>> 2
enter epsilon between 0.0001 to 3:
>> 0.01
enter the lower value for the fluid density:
>> 1000
enter the upper value for the fluid density:
>> 1500
enter the size value for the fluid density:
>> 100
```



- **Test case 5:**

```
Enter an option:
>> 5
please enter the fluid density between 0.5 to 2000:
>> 1000
please enter the diameter:
>> 5
please enter the velocity:
>> 2
enter epsilon between 0.0001 to 3:
>> 0.01
enter the lower value for the fluid viscosity:
>> 0.1
enter the upper value for the fluid viscosity:
>> 1
enter the size value for the fluid viscosity:
>> 0.1
```



Test case 6: exit the program

- **Verification:**
  - The range of the roughness is **0.0001 to 3.**
  - The range of the fluid density is **0.5 to 2000**.
  - The range of the dynamic viscosity is **10^-6 to 300**
  - The Re should be greater than **4000** for the application to run.
  - The user has to choose options from 1 to 6.