جامعة نيويورك أبوظبي

NYU ABU DHABI

ENGR-UH 1000
Computer Programming for Engineers

Assignment 1-Civil Engineering Case Study
Pressure Drop of a Fluid Through a Pipe

Mohamed Eid

Fall 2024

Step 1. Problem identification:

The objective of this assignment is to develop a software that determines the pressure drop in a horizontal pipe due to friction under laminar flow conditions. This involves calculating the pressure drop by analyzing factors such as friction losses, pipe geometry, and the flow characteristics of the fluid. The solution aims to understand how friction impacts fluid transport efficiency in systems ranging from household plumbing to large-scale industrial processes. The effects of pressure drop on a piping system can be substantial, impacting both the efficiency and the operation of the system. A significant pressure drop indicates that the piping system must work harder to pump the fluid (or require a higher elevation starting point) increasing the energy consumption of the pump. This inefficiency can lead to higher operational costs and reduced overall system performance.

2. Gathering information and input/output description:

Relevant information:

Pressure drop is a phenomenon that happens when the pressure of a fluid decreases as it goes through a pipe. It is a very important concept to study because if the pressure drop of a fluid is very large, the energy to pump the fluid through the pipe becomes very high which in return increases the cost to pump the fluid through the pipe.

Additionally, the added strain on these components can lead to more frequent maintenance or replacement, raising costs further. Significant pressure drops can create unstable flow conditions, potentially causing vibrations and mechanical wear in the piping system. Over time, this can reduce the lifespan of the components, leading to more frequent failures or operational downtime.

The software provides the user with four options to calculate the pressure drop. Option (1) asks the user to enter the value of each parameter and assigns them in their respective variables. Option (2) calculates all the intermediate values that are needed for the pressure drop, asks the user the file name he/she wants to save on and writes the intermediate values and the pressure drop inside that file. Option(3) reads the content of the file the user provides from and displays it to the user. Option(4) allows the user to exit the program.

When the user selects option(1), the program asks him/her the parameters(the density(d), height at two points(h1 and h2), length of the pipe(l), the diameter of the pipe(D), volume flow of rate of the fluid(Q), and viscosity of the fluid(u)). After accepting these values from the user it assigns them in their respective variables.

When the user selects option(2), the program calculates the intermediate values using the values the user provided and other constant values like: gravity(g), and the value of pi. And after calculating the pressure drop using these intermediate values, it saves all the intermediate values and the final pressure drop in the file the user gave. And the order of this execution is:

v = (4 x Q) / (pi x D^2),
here ‘v’ is the velocity of the fluid, ‘Q’ is the volume flow of rate, and ‘D’ is diameter of the pipe

R = (d x v x D)/u
here ‘R’ is the reynolds number, ‘d’ is density, and ‘u’ is the viscosity of the fluid

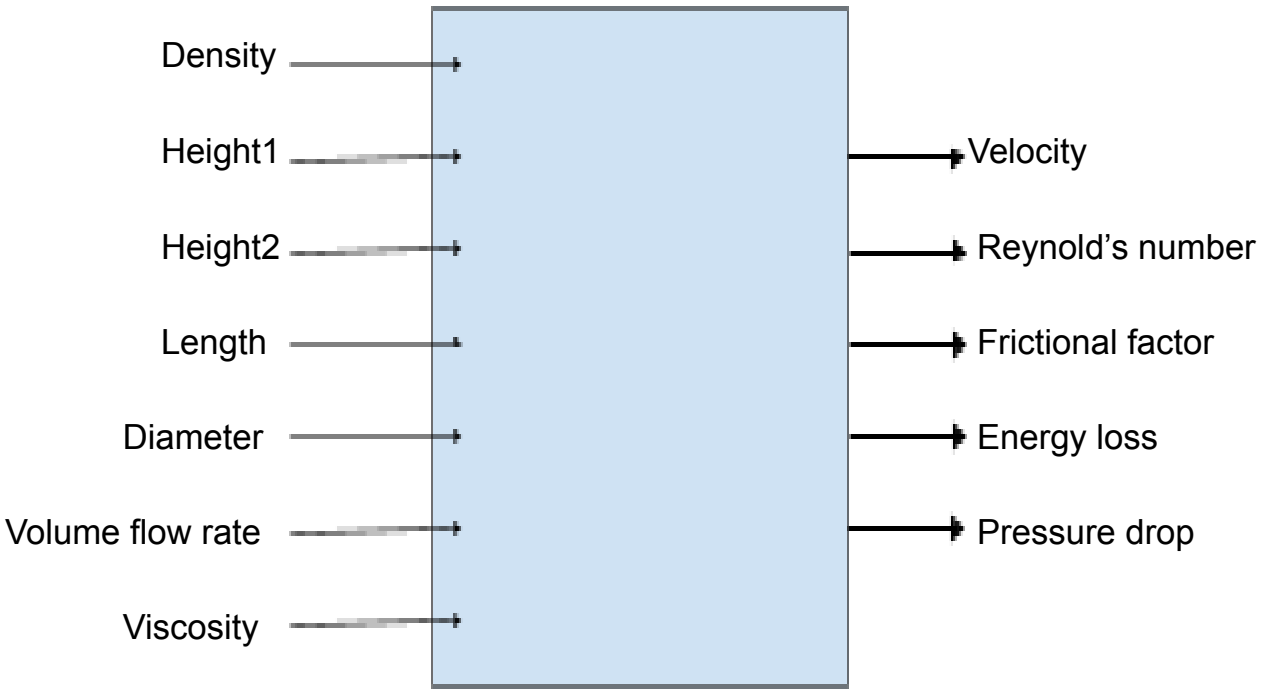
If the R is less than 2000 then f=8/R
Here ‘f’ is the friction factor of the pipe

Energy loss=(4 x f x v^2 x l) / D

Pressure drop = P1 - P2 = d x (g x hchange + Energy loss)
Here ‘g’ is gravity whose value is considered 10, and hchange is the height difference

When the user selects option(3), the program asks the user for a file name and reads all the contents inside the file the user provided.

Input/Output description:



As we can see from the diagram, the input values we need are: density of the fluid, height of the pipe at initial position from the ground, height of the pipe at another position from the ground, length of the pipe, diameter of the pipe, volume flow rate of the fluid, viscosity of the fluid. And after getting these values from the user we calculate the velocity of the fluid, the reynold's number to confirm the laminar flow of the fluid, the frictional factor of the fluid inside the pipe, the energy loss of the fluid while going through the pipe, and the final pressure drop.

3. Design of algorithm and hand-solved problems:

Design of algorithm:

//Declaring variables

Assign 0 to density
Assign 0 to height1
Assign 0 to height2
Assign 0 to heightDifference
Assign 0 to length
Assign 0 to diameter
Assign 0 to volumeFlowRate
Assign 0 to viscosity
Assign 0 to reynoldsNumber
Assign 0 to firctionFactor
Assign 0 to velocity
Assign 0 to energyLoss
Assign 0 to pressureDrop
Assign '0' to option
Declare fileLine
Declare writeFile
Declare readFile
Assign 0 to flag

//Repeating to print the menu

Repeat until option equals 4:
Print: "What are you looking to do?", newline
Print: "Enter the parameters needed to calculate the pressure", newline
Print: "Calculate the intermediate values and the pressure drop", newline
Print: "See the previously written values in a file", newline
Print: "Exit the program", newline
Read value into option

If option equal to 1:
Repeat infinitely:
If flag is greater than 1:
Declare answer
Print:"Already entered values, want to enter again, 'n' for no"
Read value into answer
If answer equals to 'n':
Break the loop

Print: "Enter height one", new line
Read value into height1
Repeat until height1 is greater than 1:
    Print: "You entered an invalid height value, try again", new line
    Read value into height1

Print: "Enter height two", new line
Read value into height2
Repeat until height2 is greater than 1:
    Print: "You entered an invalid height value, try again", new line
    Read value into height2
heightDifference=height2-height1

Print: "Enter the density of the fluid", new line
Read value into density
Repeat until density is greater than 1:
    Print: "You entered an invalid density value, try again", new line
    Read value into density

Print: "Enter the length of the pipe", new line
Read value into length
Repeat until length is greater than 1:
    Print: "You entered an invalid length value, try again", new line
    Read value into length

Print: "Enter diameter", new line
Read value into diameter
Repeat until diameter is greater than 1:
    Print: "You entered an invalid diameter value, try again", new line
    Read value into diameter

Print: "Enter volume flow rate", new line
Read value into volumeFlowRate
Repeat until volumeFlowRate is greater than 1:
    Print: "You entered an invalid volumeFlowRate value, try again"

Read value into volumeFlowRate

Print: "Enter the viscosity of the fluid", new line
Read value into viscosity
Repeat until viscosity is greater than 1:
Print: "You entered an invalid viscosity value, try again", new line
Read value into viscosity
Print: "You have entered the values successfully", new line
Break the infinite loop
Increment flag by one

Else if option equals to 2:
If flag equals to 0:
Print:"You have not the entered values, make sure you first go to option 1 first",
Else:
Declare answer
Repeat infinitely:
If flag is greater than 1:
Print:"already written the values in a file, want to write again? 'n' for no"
Read value into answer
If answer equals to 'n':
Break the loop
*Assign (4*volumeFlowRate)/(3.14*(diameter)^2) to velocity*
*Assign (density*velocity*diameter)/viscosity to reynoldsNumber*
If reynoldsNumber is greater than 2000:
Print: "The program calculates only the fluids in laminar flow",
Else:
Print: "Enter the file name you want to write the value in", new
Read value into writeFile
Assign 8/reynoldsNumber to frictionFactor
*Assign (4*frictionFactor*(velocity)^2) to energyLoss*
Assign (density(10*heightDifference+energyLoss)*
Create an output file stream called wFile and open it
If the file opening did not work:
Print: "File could not be opened", new line
Exit the program

// Write inside the file the user provided
Write "Pipe parameters info:", new line, in the file
Write "Pressure point 1 height (height1):", new line, in the file
Write height1, new line
Write "Pipe diameter:", new line, in the file
Write diameter, new line
Write "Pipe length (length):", new line, in the file
Write length, new line
Write "Volume flow rate (volumeFlowRate):", new line, in the file
Write volumeFlowRate, new line
Write "Fluid parameters info:", new line, in the file
Write "Fluid velocity (velocity):", new line, in the file
Write "Fluid density (density):", new line, in the file
Write "Fluid viscosity (viscosity):", new line, in the file
Write viscosity, new line, in the file
Write "Intermediate parameters:", new line, in the file
Write "Reynold's number (reynoldsNumber):", new line, in the file
Write reynoldsNumber, new line, in the file
Write "Pressure drop between point 1 and 2:", new line, in the file
Write pressureDrop, new line, in the file

//Print the success of the operation of writing in the file
Print "You have successfully written the values in the file", new line
Close the file

Else if option equals to '3':
Create an input file stream
Repeat infinitely:
Print: "Enter the file name you want to read from", new line
Read value into readFile
Open the file
Check if the file was not opened
Declare a variable called answer
Print: "There is no file by the name you provided", new line
Print: "Do you want to try again" n for no and any other key for yes"
Read value into answer
if answer is "n":

Break the loop

If the file was opened:

Break;

Repeat until the end of file:

Read the file line one at a time and assign them to fileLine

Print fileLine on the console

Close the file

Else if option equals to '4':

Print: "You have chosen to exit the software", new line

Print: "Thank you!", new line

Break the loop

Else any other option:

Print: "Invalid choice, make sure you select 1 to 4", new line

Hand solved-problems:

Test case-1(option 1, accepting all the values needed for the calculation of the pressure drop)

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

1

Enter height one

1

Enter height two

1

Enter the density of the fluid

1

Enter the length of the pipe

1

Enter the diameter of the pipe

1

Enter the volume of flow rate of the fluid

1

Enter the viscosity of the fluid

1

You have entered all the parameters needed to calculate the pressure drop

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

Test case-2(option 2, calculating the intermediate values and the pressure drop)

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

2

Enter the file name you want to write the values in including the file extension

Cpe.txt

The intermediate values and the final pressure drops has been successfully written in Cpe.txt

The calculated pressure difference is: 40.7643

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

Test case-3(option 3, reading the intermediate values and pressure drop from the file)

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

3

Enter the file name you want to read from including the file extension

Cpe.txt

Pipe parameters info:

Pressure point 1 height (height1): 1

Pressure point 2 height (height2): 1

Pipe diameter (diameter): 1

Pipe length (length): 1

Volume flow rate (volumeFlowRate): 1

Fluid parameters info:

Fluid velocity (velocity): 1.27389

Fluid density (density): 1

Fluid viscosity (density): 1

(Intermediate Parameters)

Reynold's number (reynoldsNumber): 1.27389

Friction factor (frictionFactor): 6.28

Energy loss per kilogram (energyLoss): 6.28

Pressure drop between point 1 and 2: 40.7643

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

Test case-4(option 4, exiting the program)

What are you looking to do?

1. Enter the parameters needed to calculate the pressure drop
2. Calculate the intermediate values and the pressure drop
3. See the previously written values in a file
4. Exit the program

4

You have chosen to exit the software

Thank you!

Step 4. Implementation

```
/*
 This is a program that calculates the drop
 of an incompressible fluid that is in a laminar flow
 click the run button to run and use it.
 */
//include essential libraries
#include <iostream> //required for cout, cin, and endl functions
#include <fstream> //required for file operation
#include <math.h> //required for pow function
```

```
#define GRAVITY 10.0
#define PI 3.14
using namespace std;
int main() {
   //declare and initialize variables
   double density(0), height1(0), height2(0), heightDifference(0), length(0),
   diameter(0), volumeFlowRate(0), viscosity(0), reynoldsNumber, frictionFactor,
   velocity, energyLoss, pressureDrop(0);
   char option('0'), fileLine;

//variables to read the file name from the user
   string writeFile, readFile;

   //a variable to track the actions of the user
   int flag(0);

   //loop until the user enters the character 4
   while(option!='4'){
      //print out the menu
      cout<<"What are you looking to do?\n";
      cout<<"1.Enter the parameters needed to calculate the pressure drop\n";
      cout<<"2.Calculate the intermediate values and the pressure drop\n";
      cout<<"3.See the previously written values in a file\n";
      cout<<"4.Exit the program\n";
      cin>>option;

      //the block that runs when the user enters character 1
      if(option=='1'){
         while(true){
            if(flag>1){//check if the user already entered the variables needed
               char answer;
               cout<<"You have already entered the values needed. Do you want to enter another set
of values? click n for no and any other key for yes\n";
               cin>>answer;
               if(answer=='n'){
                  Break;//break the loop
               }
            }

            //Accept height one from the user
            cout<<"Enter height one\n";
            cin>>height1;
            while(height1<=0){//repeat until the user gives a valid height value
               cout<<"You have entered an invalid height value, try again\n";
               cin>>height1;
            }

            //Accept the height two from the user
            cout<<"Enter height two\n";
```

```
cin>>height2;
while(height2<=0){//repeat until the user gives a valid height value
   cout<<"You have entered an invalid height value, try again\n";
   cin>>height2;
}
heightDifference=height2-height1;//calculate the height difference

//Accept the density from the user
cout<<"Enter the density of the fluid\n";
cin>>density;
while(density<=0){//repeat until the user gives a valid density value
   cout<<"You have entered an invalid density value, try again\n";
   cin>>density;
}

//Accept the length from the user
cout<<"Enter the length of the pipe\n";
cin>>length;
while(length<=0){//repeat until the user gives a valid length value
   cout<<"You have entered an invalid length value, try again\n";
   cin>>length;
}

//Accept the diameter from the user
cout<<"Enter the diameter of the pipe\n";
cin>>diameter;
while(diameter<=0){//repeat until the user gives a valid diameter value
   cout<<"You have entered an invalid diameter value, try again\n";
   cin>>diameter;

}

//Accept the volume flow rate from the user
cout<<"Enter the volume flow rate\n";
cin>>volumeFlowRate;
while(volumeFlowRate<=0){//repeat until the user gives a valid volume flow rate value
   cout<<"You have entered an invalid volume flow rate value, try again\n";
   cin>>volumeFlowRate;
}

//Accept the viscosity from the user
cout<<"Enter the viscosity of the fluid\n";
cin>>viscosity;
while(viscosity<=0){//repeat until the user gives a valid viscosity value
   cout<<"You have entered an invalid viscosity value, try again\n";
   cin>>viscosity;
}

cout<<"You have entered all the parameters needed to calculate the pressure drop\n";
```

```
            break;//option one has been executed so break the loop
        }
         flag++; //increment the flag by one which confirms the user already selected 1
        }


      //the block that runs when the user enters character 2
      else if(option=='2'){
        if(flag==0){//check if the user did not select option 1 before selecting option 2
            cout<<"You have not entered the values to calculate the pressure drop, go back to
option 1 and make sure you enter all the values needed\n";
        }
        else {
            char answer;
            while(true){
              if(flag>1){//check if the user already written the values in a file
                  cout<<"You have already entered all the values and written in "<<writeFile<<". do
you want to write the same in another file again? click n for no and any other key for yes\n";
                  cin>>answer;
                  if(answer=='n')
                    break;
              }
                  velocity = (4 * volumeFlowRate) / (PI * pow(diameter, 2));//calculate the velocity
                  reynoldsNumber = (density * velocity * diameter) / viscosity;//calculate the reynolds
number
                  if(reynoldsNumber>2000){//check if the flow is laminar
                    cout<<"The reynold's number is greater than 2000, which means the flow of the
fluid is turbulant and this program only calculates for the laminar flow\n";
                  }
                  else{
                    cout<<"Enter the file name you want to write the values in including the file
extention\n";//prompt the user to enter the file name
                    cin>>writeFile;//get the file name from the user
                    frictionFactor=8/reynoldsNumber;
                    energyLoss=(4*frictionFactor*pow(velocity, 2)*length)/diameter;
                    pressureDrop=density * (GRAVITY * heightDifference + energyLoss);
                    ofstream wFile(writeFile, ios::app);//create the output file stream and open the file
                    if(wFile.fail()){//check if the file is opened
                      cout<<writeFile<<" could not be opend\n";
                      exit(-1);
                    }
                    //write the intermediate and the final pressure drop value in the file
                    wFile<<"Pipe parameters info:\n";
                    wFile<<"\tPressure point 1 height (height1): "<<height1<<endl;
                    wFile<<"\tPressure point 2 height (height2): "<<height2<<endl;
                    wFile<<"\tPipe diameter (diameter): "<<diameter<<endl;
                    wFile<<"\tPipe length (length): "<<length<<endl;
                    wFile<<"\tVolume flow rate (volumeFlowRate): "<<volumeFlowRate<<endl;
                    wFile<<"Fluid Parameters info:\n";
```

```
                    wFile<<"\tFluid velocity (velocity): "<<velocity<<endl;
                    wFile<<"\tFluid density (density): "<<density<<endl;
                    wFile<<"\tFluid viscosity (viscosity): "<<viscosity<<endl;
                    wFile<<"\n";
                    wFile<<"\t(Intermediate Parameters)\n";
                    wFile<<"\tReynold's number (reynoldsnumber): "<<reynoldsNumber<<endl;
                    wFile<<"\tFriction factor (frictionFactor):"<<frictionFactor<<endl;
                    wFile<<"\tEnergy loss per kilogram (energyLoss): "<<frictionFactor<<endl;
                    wFile<<"Pressure drop between point 1 and 2: "<<pressureDrop;
                    wFile<<"\n";
                    wFile<<"\n";
                    cout<<"The intermediate values and the final pressure drop has been
successfully written in "<<writeFile<<endl;
                    wFile.close();
                    cout<<"The calculated pressure difference is: "<<pressureDrop<<endl;
                    flag++;//increment the flag by one which confirms the user has written the values
          in a file
                    break;
                }
              }
            }
               }

      //the block that runs when the user enters character 3
      else if(option=='3'){

               ifstream rFile;//create an input file stream
               while(true){
                  cout<<"Enter the file name you want to read from including the file
extention\n";//prompt the user to enter the file name
                  cin>>readFile;//get the file name
                  rFile.open(readFile, ios::in);//open the file
                  if(rFile.fail()){//check if the file is not opened
                     char answer;
                     cout<<readFile<<" does not exist\n";
                     cout<<"You want to try again? n for no and any other key for yes\n";
                     cin>>answer;
                     if(answer=='n'){
                        break;
                     }
                  }
                  else{
                     break;
                  }
               }
               while(!rFile.eof()){//iterate over the line inside the file
                  rFile.get(fileLine);//read the lines of the file one at a time
                  cout<<fileLine;//print the line that is read on the console
               }
```

```
            rFile.close();
        }

    //the block that runs when the user enters character 4
    else if(option=='4'){
        //confirms the user that the program is terminated
        cout<<"You have chosen to exit the software\n";
        cout<<"Thank you !\n";
        break;//exit the loop
    }

     //the block that runs when the user enters any character other than 1, 2, 3, 4
    else{
        cout<<"Invalid input, make sure you select 1 to 4\n";
    }

    }
}
```

Step 5. Software testing and verification

Test case-1(option 1, accepting all the values needed for the calculation of the pressure drop)

```
What are you looking to do?
1.Enter the parameters needed to calculate the pressure drop
2.Calculate the intermediate values and the pressure drop
3.See the previously written values in a file
4.Exit the program
1

Enter height one
1

Enter height two
1

Enter the density of the fluid
1

Enter the length of the pipe
1

Enter the diameter of the pipe
1

Enter the volume flow rate
1

Enter the viscosity of the fluid
1

You have entered all the parameters needed to calculate the pressure drop
What are you looking to do?
1.Enter the parameters needed to calculate the pressure drop
2.Calculate the intermediate values and the pressure drop
3.See the previously written values in a file
4.Exit the program
```

Test case-2(option 2, calculating the intermediate values and pressure drop)

```
What are you looking to do?
1.Enter the parameters needed to calculate the pressure drop
2.Calculate the intermediate values and the pressure drop
3.See the previously written values in a file
4.Exit the program
2

Enter the file name you want to write the values in including the file extention
Cpe.txt

The intermediate values and the final pressure drop has been successfully written in Cpe.txt
The calculated pressure difference is: 40.7643
What are you looking to do?
1.Enter the parameters needed to calculate the pressure drop
2.Calculate the intermediate values and the pressure drop
3.See the previously written values in a file
4.Exit the program
```

Test case-3(option 3, calculating the intermediate values and pressure drop)

```
The calculated pressure difference is: 40.7643
What are you looking to do?
1.Enter the parameters needed to calculate the pressure drop
2.Calculate the intermediate values and the pressure drop
3.See the previously written values in a file
4.Exit the program
3

Enter the file name you want to read from including the file extention
Cpe.txt

Pipe parameters info:
    Pressure point 1 height (height1): 1
    Pressure point 2 height (height2): 1
    Pipe diameter (diameter): 1
    Pipe length (length): 1
    Volume flow rate (volumeFlowRate): 1
Fluid Parameters info:
    Fluid velocity (velocity): 1.27389
    Fluid density (density): 1
    Fluid viscosity (viscosity): 1

    (Intermediate Parameters)
    Reynold's number (reynoldsnumber): 1.27389
    Friction loss per kilogram (energyLoss): 6.28
Pressure drop between point 1 and 2: 40.7643
```

Test case-4(option 4, exiting the program)

```
What are you looking to do?
1.Enter the parameters needed to calculate the pressure drop
2.Calculate the intermediate values and the pressure drop
3.See the previously written values in a file
4.Exit the program
4

You have chosen to exit the software
Thank you !
Program ended with exit code: 0
```

User Guide

This is a program that will help the user calculate the pressure drop of an incompressible fluid that is in a laminar flow. As the program is written in C++ programming language, to use it, the user needs a C++ compiler in his/her computer. When the user runs the program for the first time, he/she will be provided with 4 options. Selecting the first option will prompt the user to enter the necessary parameters that are needed to calculate the pressure drop. After entering the parameters, the user can select option two to write the intermediate values and final pressure drop of the fluid in a text file. However, before writing the values in a file, the user needs to enter the necessary parameters in option one. So selecting the first option is necessary before selecting option two. The user can select option three to see the values he/she has written in a text file. When the user selects option three, he/she will be prompted to enter the file name he/she wants to read from. After doing the calculation, writing in a text file, and reading from the file, the user can select option four to exit the program. The user can find this proposal and the C++ source code in the following github repository: