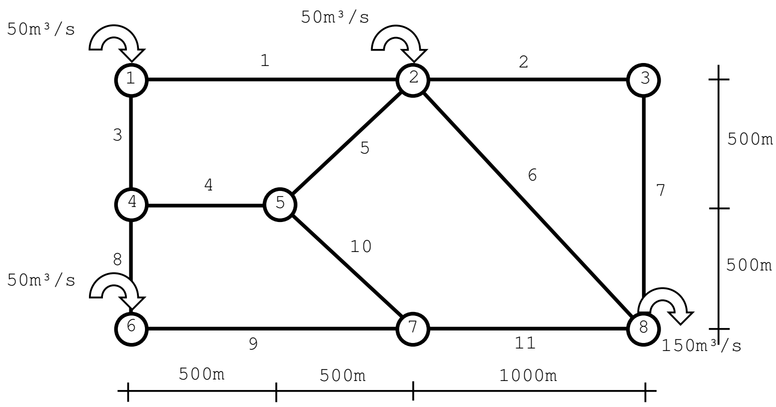
Assignment: Pipe Network

due to January 24th, 2018
including a short interview in room 3238, 9:45 am

A village in Bavaria plans to build a new water supply network. Your task is to develop a C++ console application, which calculates the flows in the tubes of such a network. The pipe network consists of nodes and tubes between the nodes.

- A node is defined by its coordinates x, y and a flow Q (inflow: Q<0; outflow: Q>0)
- A tube with diameter d lies between two nodes a, b

The picture shows a simplified model of the network:



Write a program with the following functionality:

- All input data for the nodes and the tubes is to be read from a text file and shall be saved in the class structure given on the next page (see Sections “Input File”, “Class structure”)
- The resulting fluxes shall be computed (see Sections “Physical problem”, “Pipe network”)
- Output the results either on screen or in an output file

Input file:

The input file containing the network data is divided into three parts: First, the number of nodes and tubes has to be specified, and then the nodes are given with the x-coordinate, the y-coordinate and the amount of inflow/outflow. The tubes are then given with the start node, the end node and the diameter. The input file has to be saved as a simple text file in the project directory. For the given example it will look like this:

number of nodes
number of tubes
node data: x,y,Q
tube data: node_a, node_b, diameter

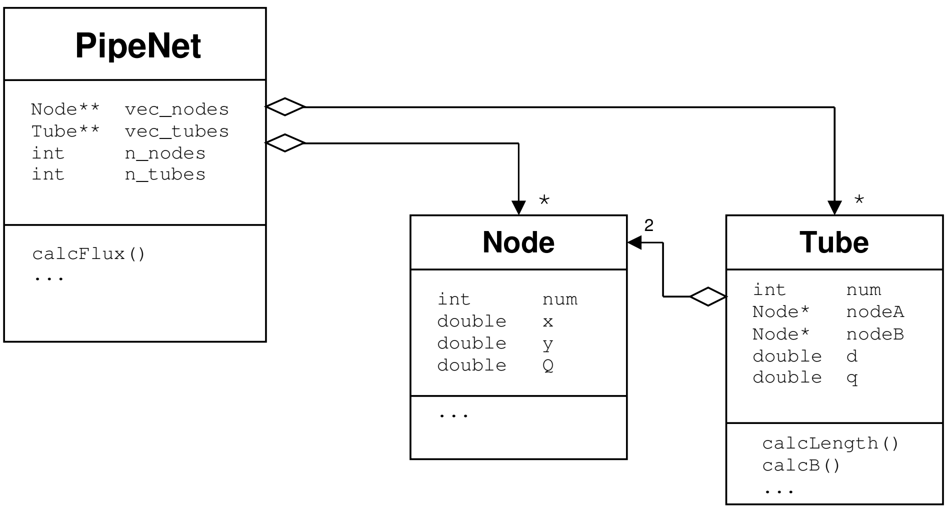
```
8
11
0      0     -50
1000   0     -50
2000   0     0
0      500   0
500    500   0
0      1000  -50
1000   1000  0
2000   1000  150
1 2    0.5
2 3    0.5
1 4    0.5
4 5    0.5
5 2    0.5
2 8    0.5
3 8    0.5
4 6    0.5
6 7    0.5
5 7    0.5
7 8    0.5
```

The example has 8 nodes and 11 tubes with inflow at the nodes 1, 2 and 6 and outflow at the node 8. All tubes have the same diameter of 0.5m.
In order to read the input file, it is recommended to use an input filestream (`ifstream`) object (defined in <fstream>). The following example shows how to read a file with the name "input.dat" whose first entry is an integer number specifying the number of the following floating point numbers.

```
int n; double *array;
ifstream infile("input.dat");              //opening the file;
infile >> n;                               //use >> operator to read from file
array = new double[n];                     //allocate memory to read data
for(int i=0;i<n;++i) infile >> array[i];
```

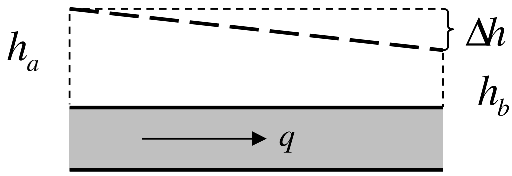
Class structure:

It is mandatory to use the following class structure in your project:



Physical problem:

The underlying physical problem can be demonstrated by considering the example of a single tube inside of which a fluid flows:



The loss of hydraulic head Δh in the tube depends on the velocity υ of the flow

$$\Delta h = \lambda \frac{l}{d} \frac{\upsilon^2}{2g},$$

where l is the length of the tube, d the diameter and $g = 9.81 m/s^2$ the acceleration due to gravity. The factor λ describes the roughness of the tube material. For laminar flow:

$$\lambda = \frac{64}{\mathrm{Re}}, \qquad \mathrm{Re} = \frac{\upsilon \cdot d}{\nu} = \frac{\upsilon \cdot d}{1 \cdot 10^{-6}}$$

With $\upsilon = \frac{q}{A} = \frac{4 \cdot q}{d^2 \pi}$ we get:

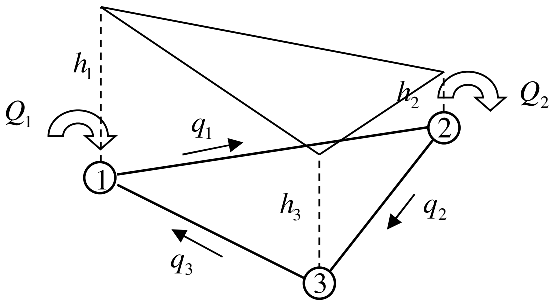
$$\Rightarrow \quad q = \frac{\pi \cdot g \cdot d^4}{128 \cdot \nu \cdot l} \Delta h_i = \underbrace{\frac{\pi \cdot g \cdot d^4}{128 \cdot \nu \cdot l}}_{B} \left(h_a - h_b\right)$$

The factor B specifies the permeability of the tube. Finally we get a linear relation between the flow and the difference in hydraulic head:

$$q = B \cdot \left(h_a - h_b\right)$$

Pipe Network:

To demonstrate the procedure calculating the flow distribution in a pipe network, an example with 3 nodes and 3 tubes is shown below:



The system must fulfill the constraint that the sum of the inflow and the outflow has to be zero in every node (note: outflow is positive):

$$\begin{aligned}
q_1 - q_3 - Q_1 &= B_1(h_1 - h_2) - B_3(h_3 - h_1) - Q_1 &&= 0 \\
-q_1 + q_2 + Q_2 &= -B_1(h_1 - h_2) + B_2(h_2 - h_3) + Q_2 &&= 0 \\
-q_2 + q_3 &= -B_2(h_2 - h_3) + B_3(h_3 - h_1) &&= 0
\end{aligned}$$

This leads to a linear equation system

$$\underline{\underline{B}}\,\underline{h} + \underline{Q} = \underline{0}$$

$$\begin{bmatrix} B_1 + B_3 & -B_1 & -B_3 \\ -B_1 & B_1 + B_2 & -B_2 \\ -B_3 & -B_2 & B_2 + B_3 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} + \begin{bmatrix} -Q_1 \\ +Q_2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

where $\underline{\underline{B}}$ is the global permeability matrix and $\underline{Q}$ denotes the load vector. The equation system shown here is singular, i.e. it has no unique solution. This is due to the fact that all formulas are based on the *difference* in hydraulic head between two nodes, but an absolute height of the system is not specified. Therefore, we have to include such a boundary condition by defining the absolute height of one node (e.g. height of first node: h_1=0). This can be enforced, for example, by setting the off-diagonal entries $B_{1,i}$ and $B_{i,1}$ of the coefficient matrix and the entry Q_1 of the right hand side vector to 0, while setting the diagonal entry $B_{1,1}$ to 1 (see algorithm).