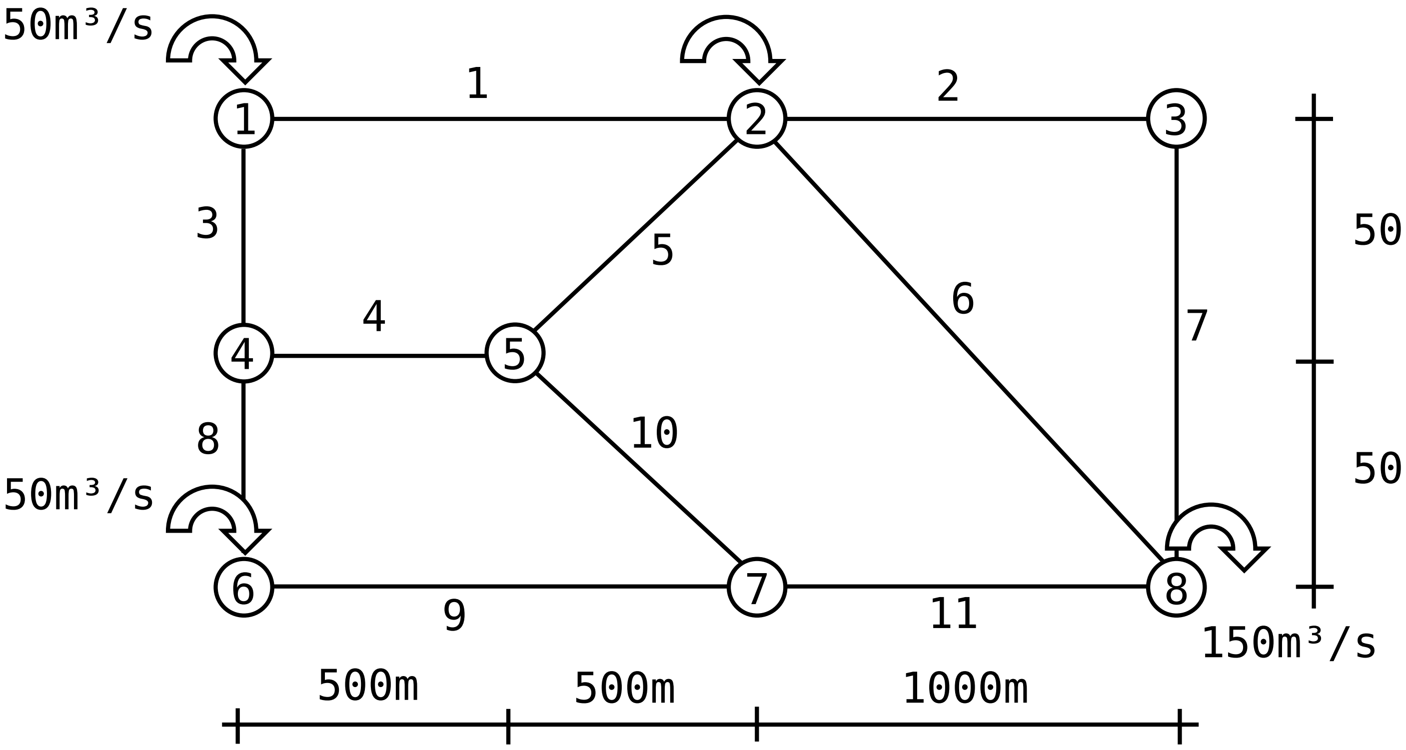
Pipe network assignment

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 pipe network consists of

- Nodes: A node is defined by its two spatial coordinates, a global id, and a flow value Q (inflow: $Q < 0$, outflow: $Q > 0$)
- Tubes: A tube has a diameter and lies between two nodes

Pipe network assignment

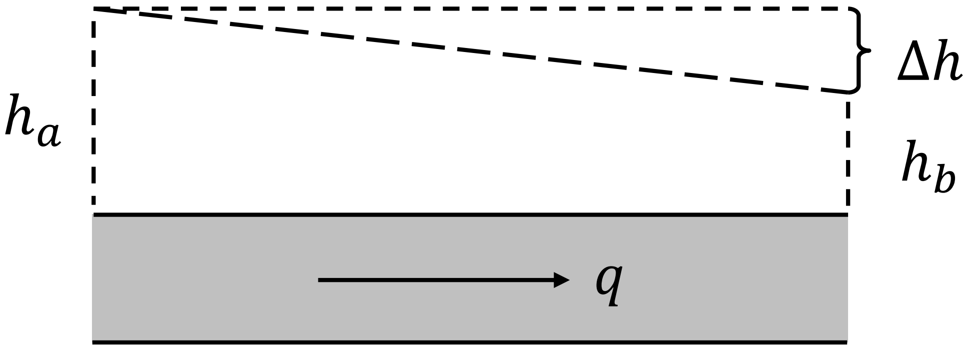
1. Write a shared library that allows to
   - read input data for nodes and tubes from a file, see section *input file*
   - use this data to set up a class structure given in the section *class structure*
   - compute the resulting fluxes according to sections *physical problem*, *example* and *algorithm*
   - output the results either on screen or in an output file
2. Add a unit test executable for your library and add the tests (`pipenetwork_test.cpp`) uploaded in Moodle. You will have to provide the catch2 header for the tests to compile (see e.g. exercise 4)
3. Write a driver executable that uses your library to print the fluxes in a pipe network given in section *input file*. Store the data in a text file `pipedata.txt` and read that file from your driver.

You are also given a simple linear algebra library on Moodle that allows you to perform computations with vectors and matrices.

The assignment is due to **January 25th**, 2023. There will be short individual **interviews.** The schedule and location will be announced later

Pipe network assignment: Physical problem

The underlying physical problem is explained by considering the example of a single tube:



The loss of hydraulic head in the tube depends on the velocity u of the flow:

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

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

$$\lambda = \frac{64}{\mathrm{Re}}, \qquad Re = \frac{u \cdot d}{\nu},$$

where $\nu = 10^{-6}$ for water. With

$$u = \frac{q}{A} = \frac{4 \cdot q}{d^2 \pi},$$

we get

$$q = \frac{\pi \cdot g \cdot d^4}{128 \cdot \nu \cdot l} \Delta h = \frac{\pi \cdot g \cdot d^4}{128 \cdot \nu \cdot l} (h_a - h_b),$$

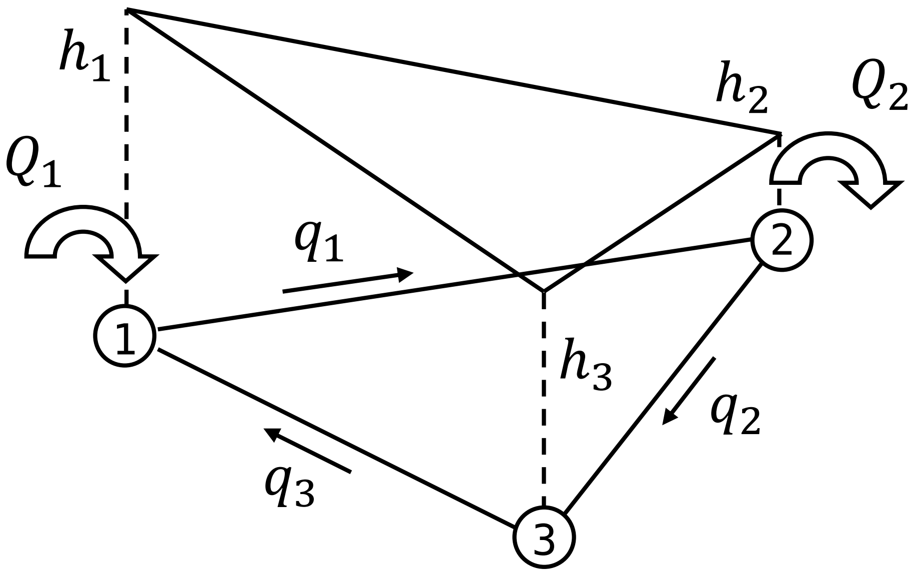
or simply $q = B(h_a - h_b)$, where

$$B = \frac{\pi \cdot g \cdot d^4}{128 \cdot \nu \cdot l}$$

represents the permeability of the tube.

Pipe network assignment: Example

To demonstrate the procedure calculating the flow distribution in a pipe network, consider the following example:



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

$$q_1 - q_3 - Q_1 = 0$$
$$-q_1 + q_2 + Q_2 = 0$$
$$-q_2 + q_3 = 0$$

which gives:

$$B_1(h_1 - h_2) - B_3(h_3 - h_1) - Q_1 = 0$$
$$-B_1(h_1 - h_2) + B_2(h_2 - h_3) + Q_2 = 0$$
$$-B_2(h_2 - h_3) + B_3(h_3 - h_1) = 0$$

or $\mathbf{Bh} = \mathbf{Q}$ in matrix form:

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

Pipe network assignment: Example

The matrix $\mathbf{B}$ represents the global permeability matrix and $\mathbf{Q}$ the load vector. The elements in the solution vector $\mathbf{h}$ are the values of the head at the nodes.

The resulting equation system 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. $h_1 \coloneqq 0$).

This can be enforced, for example, by setting the off-diagonal entries K_{i1} and K_{1i} of the coefficient matrix and the entry Q_i of the right hand side vector to 0, while setting the diagonal entry K_{11} to 1.

Now we have the following equation system:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & B_1 + B_2 & -B_2 \\ 0 & -B_2 & B_2 + B_3 \end{bmatrix} \begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} 0 \\ -Q_2 \\ 0 \end{pmatrix}$$

Just like in finite elements there are other methods to impose Dirichlet boundary conditions, however, the presented method is sufficient.

Pipe network assignment: Algorithm

The following pseudo code sketches the algorithm to compute the fluxes of the pipe network (Note, that indices start from 0 here):

```
Create zero-initialized permeability matrix B
∀ tubes i:
   ➢ get id1 and id2 of the two bounding nodes
   ➢ compute permeability factor B_i
   ➢ assemble B_i into the B matrix:
      • B(id1, id1) += B_i
      • B(id2, id2) += B_i
      • B(id1, id2) -= B_i
      • B(id2, id1) -= B_i

create load vector Q
∀ nodes i:
   Q[i] = -Q_in[i] (the given (in-/out) flow value)
```

1. Set up permeability matrix

2. Set up load vector

Pipe network assignment: Algorithm

```
for i = 1 ... number nodes:
   ➢ B(i, 0) = 0
   ➢ B(0, i) = 0
B(0,0) = 1
Q(0) = 0

solve linear equation system to get head vector h

create zero-initialized vector of fluxes q
∀ tubes i:
   ➢ extract the entries h1 and h2 at the ids of the
     two bounding nodes from h
   ➢ compute permeability factor B_i
   ➢ q[i] = B_i * (h1 - h2)

return q
```

3. Impose boundary condition

4. Solve linear system

5. Postprocess fluxes

Pipe network assignment: Input file

The input file containing the network data is divided into three parts:

1. Number of nodes
   Number of tubes
2. Node data: Each line contains one node as x-coordinate, y-coordinate and the inflow/outflow separated by a whitespace.
3. Tube data: Each line contains one tube defined by start and end node ids and its diameter

The input file is saved as text file.

The example on the right side has 8 nodes and 11 tubes with inflow at the nodes 0, 1 and 5 and outflow at the node 7. All tubes have the same diameter of 0.5 meters.

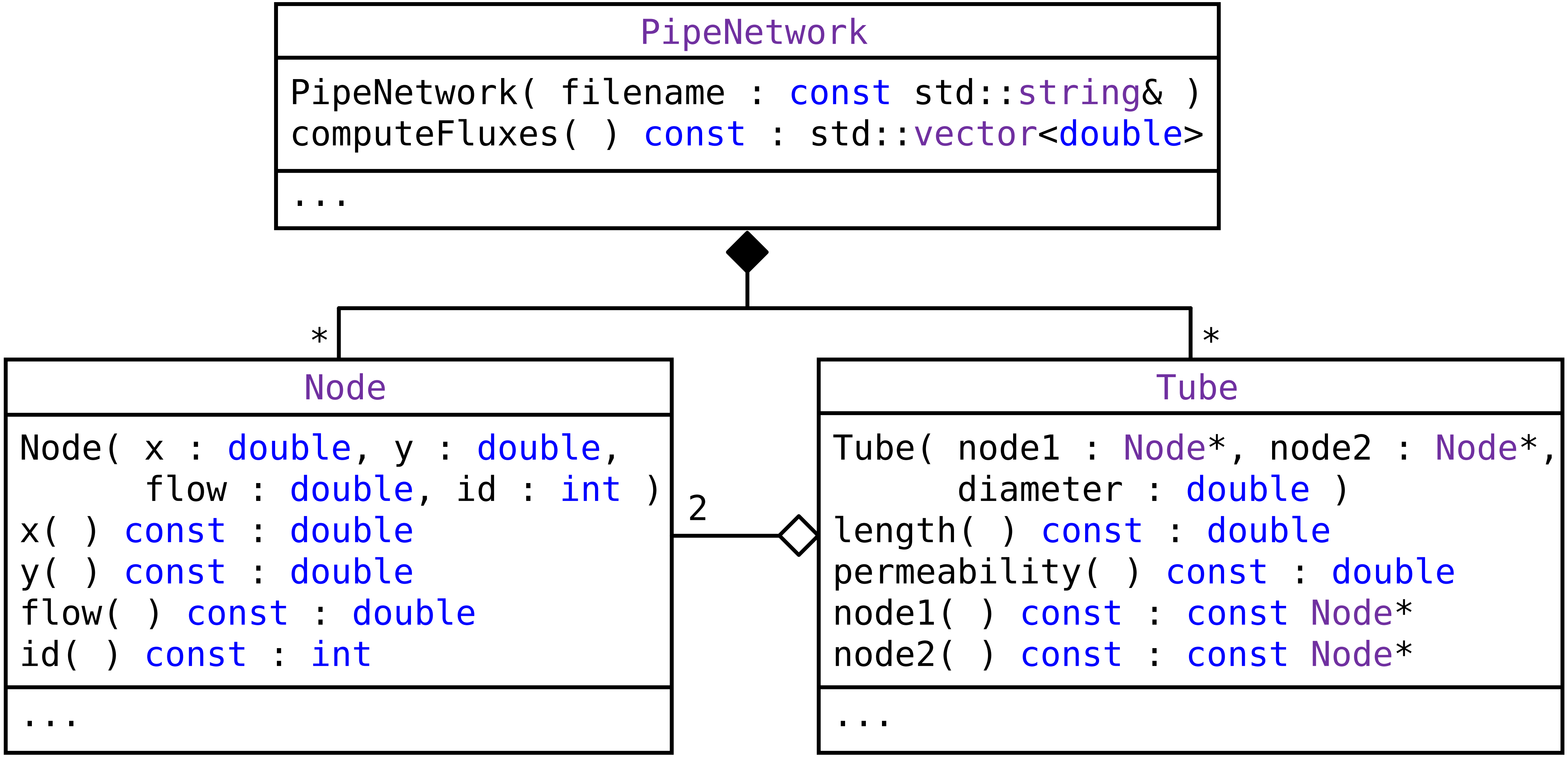
```
8
11
0       0       -50
1000    0       -50
2000    0       0
0       500     0
500     500     0
0       1000    -50
1000    1000    0
2000    1000    150
0 1     0.5
1 2     0.5
0 3     0.5
3 4     0.5
4 1     0.5
1 7     0.5
2 7     0.5
3 5     0.5
5 6     0.5
4 6     0.5
6 7     0.5
```

Pipe network assignment: Input file

The following code shows how to read a file with the name `input.txt` whose first entry is an integer number specifying the number of following floating point numbers.

```
std::ifstream infile( "input.txt" );

int numberOfValues;
infile >> numberOfValues;

std::vector<double> data;
data.reserve( numberOfValues );

for( int i = 0; i < numberOfValues; ++i )
{
    double value;
    infile >> value;

    data.push_back( value );
}
```

Pipe network assignment: Class structure



Pipe network assignment: Code

- Set up a new source project using CMake. Your source project shall not contain project files of any kind other than the CMakeLists.txt files. The process of creating a build project from your source code can be part of the interview.
- Use a separate pair of header and source files for each class you introduce and choose the same name as the class. For example: `PipeNetwork.hpp`, `PipeNetwork.cpp`, `Node.hpp`, `Node.cpp`, `Tube.hpp` and `Tube.cpp`
- Embed your library code into the (nested) namespaces `cie` and `pipenetwork`
- Choose the member variables according to ownership model shown in the UML diagram on the previous page. You may add additional members.
- Make all member variables private
- Don’t modify the unit tests, you can, however, add additional tests for code you write. Make sure that you set up your test files such that exactly one file defines `CATCH_CONFIG_MAIN`.
- Pay attention to the const qualifiers in the UML diagram, most member functions are const and some return values