Part I
Worksheet 3

1 Backlog

1. **Tool to create scenarios (domain + obstacles)**

   (a) Group disussion:

   what "tool" are we using to create domains? (consider: should be easy to "integrate" and flexible enough to build "complex" scenarios)

   Programmatically... Matlab/Python - just create a boolean matrix, and tell which "pixels" are obstacles, matrix can also be plotted easily, etc. There are also scaling functions to make it larger, etc.

   Graphically... "Paint"-like drawing and then reading via a script?

   (b) Group discussion:

   Do we use "read_pgm"? (see page 12 (3) `https://en.wikipedia.org/wiki/Netpbm_format`)

2. **Adapt parameter file, read parameter and check for correctness**

   (a) Variable length in all three directions (x_length, y_length, z_length); adapt signature of functions; see page 12 (1)

   (b) substitute "wallVelocity" by an array of parameter, for inflow condition; see page 12 (2)

3. **Adapt initializeFields according to the current problem we are simulating; see page 12 (3)**

   (a) extent signature of initializeFields with "*char problem", which contains the name of scenario

   (b) set flagField according to the current geometry (created in step 1)

   (c) using of "read_pmg" can be used to initialize flagField — Decide !!

   (d) Check for correctness, is there is any illegal boundary (too thin)?

4. **Write treatBoundary again**

   (a) implement other boundary conditions (Free-Slip, Inflow, Outflow); see page 12 (2)

   (b) rewrite iteration such that arbitrary geometries are handled

   (c) try to improve performance (that can, for example, involve additional steps in the initialization of the simulation)

5. **Write WriteVTKOutput again**

   (a) Currently it is restricted to the previous worksheet

(b) One possible way would be to use "read_pgm" and store the column and line number to help in creating the coordinates corresponding to different types of cells. Then iterate over the flag field and print out only the fluid cells Decide!!

6. **Scenarios**

   (a) Create scenarios given in worksheet (page 13)

   (b) Save results and paraview visualization files in a special folder, (Note: page 13, "Please bring your computer[...] to speed up the review process")

   (c) Create some own scenarios/tests

   (d) "Use the features provided by ParaView excessively to analyze the flows." → generate outputs and save them for discussion/understanding/oral exam

7. **Tests for plausibility**

   (a) Find given given examples and see if our results match

   (b) Test small trivial examples

   (c) Test "free-slip" with half the domain against the normal simulation for example of "Plane shear force" (see section 2.4(b) page 12)

2 Consider from worksheet!

- Changes of parameters or boundary conditions should no longer require modifications in the source code, nor recompilation of the program.
- Demonstrate, that the program works properly by providing solutions for the examples shown in the last section.
- Implement the possibility to perform computations of the problems that are defined by setting the boundary pressure. Check your program with the plane shear flow and the flow over a step.
- With the help of ParaView, visualize the Karman vortex street and the flow over a step. Use the features provided by ParaView excessively to analyze the flow