0.1 question 1

The PISO algorithm implementation in openFoam:

1)define equation for U

```
fvVectorMatrix UEqn
(
          fvm::ddt(U)
     + fvm::div(phi, U)
     - fvm::laplacian(nu, U)
);
```

2)solve the momentum predictor:

```
solve (UEqn == -fvc::grad(p));
```

3)Calculate a_p coefficient and calculate U

```
volScalarField rUA = 1.0/UEqn().A();
 U = rUA*UEqn().H();
```

4)Calculate the flux

```
phi = (fvc::interpolate(U) & mesh.Sf())
       + fvc::ddtPhiCorr(rUA, U, phi);
 adjustPhi(phi, U, p);
```

5)Solve the pressure equation

```
fvScalarMatrix pEqn
 (
    fvm::laplacian(rUA, p) == fvc::div(phi)
 );
 pEqn.setReference(pRefCell, pRefValue);
 pEqn.solve();
```

6)Correct the flux

```
if (nonOrth == nNonOrthCorr)
 {
    phi -= pEqn.flux();
 }
```

7) Correct the flux

```
 if (nonOrth == nNonOrthCorr)
 {
    phi -= pEqn.flux();
 }
```

8)calculate continuity error

```
# include "continuityErrs.H"
```

9)Perform the momentum corrector step

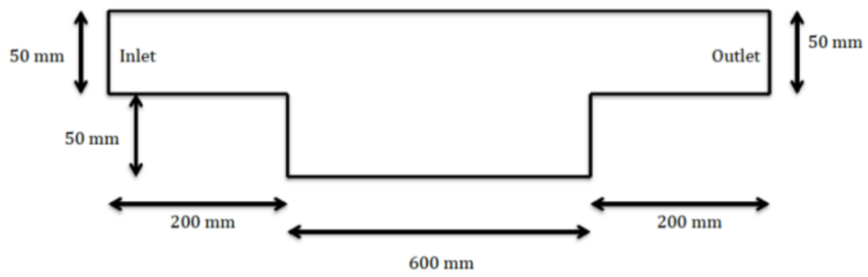
```
U -= rUA*fvc::grad(p);
 U.correctBoundaryConditions();
```

10)repeat from 3) for the prescribed number of PISO corrector step

0.2 question 2

consider the domain given by

Figure 1: Computation domain



We want to solve this flow using the pisoFoam solver for LES model. The governing equations are given by the LES continuity equation and the LES momentum equations:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \bar{\vec{u}}) = 0 \tag{1}$$

$$\frac{\partial (\rho \bar{\vec{u}})}{\partial t} + \nabla \cdot (\rho \bar{u} \bar{\vec{u}}) = -\frac{\partial \bar{p}}{\partial t} + \mu \nabla \cdot (\nabla \bar{u}) - \nabla \cdot (\rho \overline{u \vec{u}}) - \nabla \cdot (\bar{u} \bar{\vec{u}}) \tag{2}$$

$$\frac{\partial (\rho \bar{\vec{v}})}{\partial t} + \nabla \cdot (\rho \bar{v} \bar{\vec{v}}) = -\frac{\partial \bar{p}}{\partial t} + \mu \nabla \cdot (\nabla \bar{v}) - \nabla \cdot (\rho \overline{v \vec{v}}) - \nabla \cdot (\bar{v} \bar{\vec{v}}) \tag{3}$$

Boundary condition:
Constant velocity at the inlet $u = 10m/s$. Solid wall

The mean velocity and pressure are given by :

Figure 2: average pressure

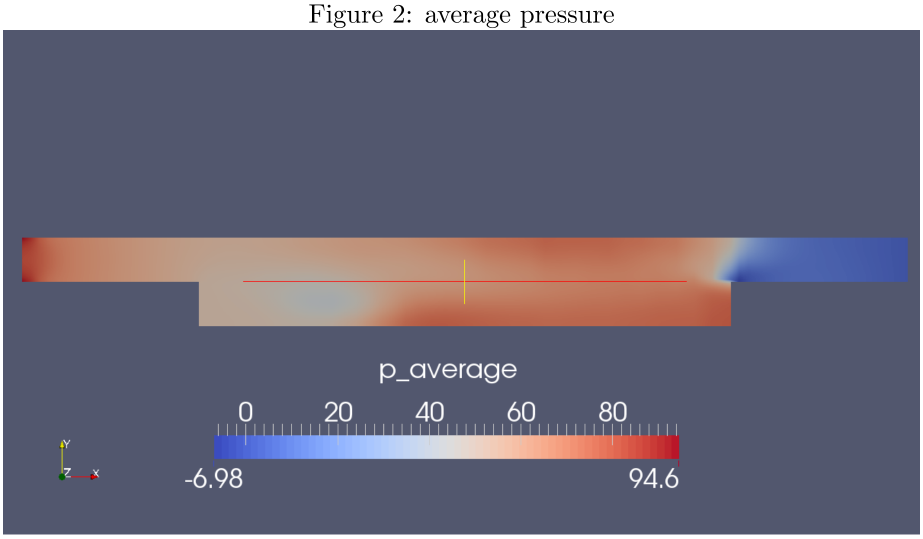
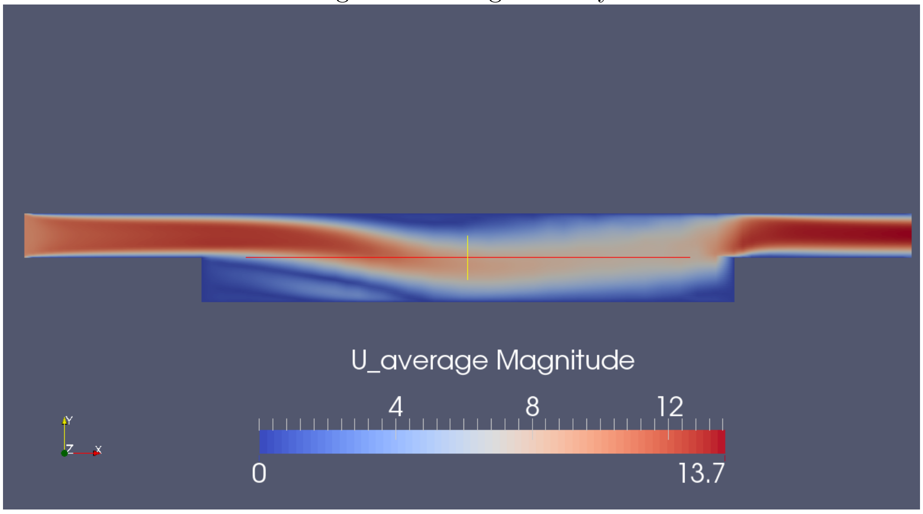


Figure 3: average velocity



The size of the mesh will drive the quality of the solution.

0.3 question 3

We use pimplefoam to solve the same flow. The k-ϵ model equations are given by the:

$$\frac{\partial(\rho k)}{\partial t} + \nabla \cdot (\rho k \vec{U}) = \nabla \cdot (\frac{\mu_t}{\sigma_k} \nabla k) + 2\mu_t S_{ij}.S_{ij} - \rho\epsilon \tag{4}$$

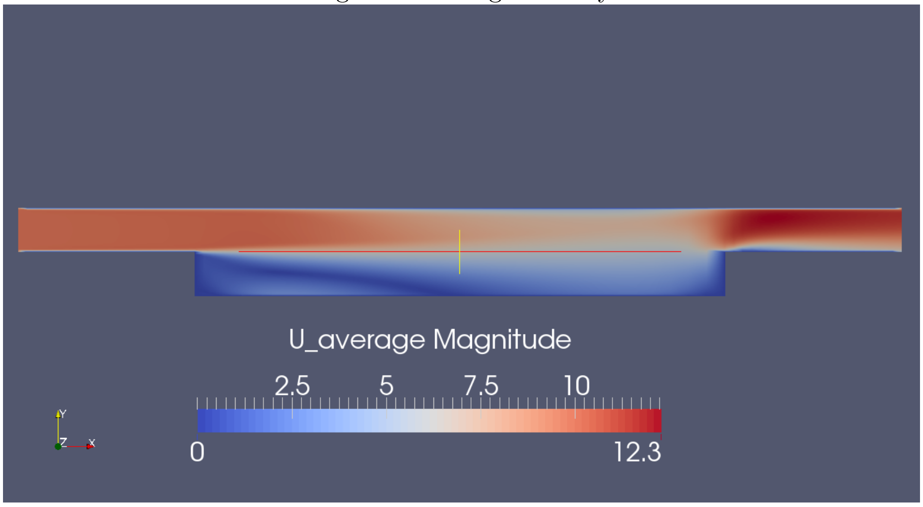
$$\frac{\partial(\rho \epsilon)}{\partial t} + \nabla \cdot (\rho \epsilon \vec{U}) = \nabla \cdot (\frac{\mu_t}{\sigma_\epsilon} \nabla \epsilon) + C_{1\epsilon}\frac{\epsilon}{k} 2\mu_t S_{ij}.S_{ij} - C_{2\epsilon}\rho\frac{\epsilon^2}{k} \tag{5}$$

with

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon} \tag{6}$$

the boundary condition are:
the velocity in the x direction at the inlet is $u = 10m/s$ and at the wall we have solid wall with no slip-condition.

Figure 4: average velocity



0.4 question 4

pimplefoam uses large time step. It is more suitable for RANS because RANS can use coarser mesh than LES which uses more finer mesh. In RANS you use larger time step then in LES.

0.5 question 6

LES is not usually used as a two D model because turbulence is a 3D phenomenon.