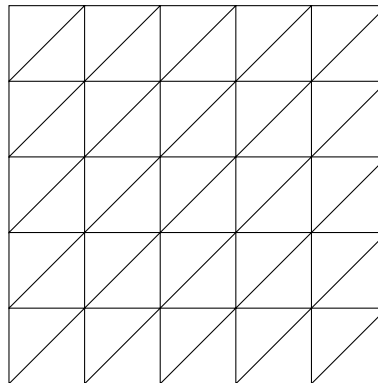EXERCISE 1 P_1 FINITE ELEMENTS ON A STRUCTURED SIMPLICIAL MESH
Consider the Poisson equation

$$-\Delta u = f \qquad\qquad \text{in } \Omega = (0,1)^2 \subset \mathbb{R}^2,$$
$$u = 0 \qquad\qquad \text{auf } \partial\Omega\ .$$

We want to solve this equation numerically with P_1 Finite Elements. The unitsquare Ω is discretized with the following *structured triangular mesh*:

Let N be the number of divisions in x- and y-direction (in the picture we have $N = 5$), thus the mesh size is $h = \frac{1}{N}$. We number the nodes starting from the origin row-wise beginning at 0 to $N^2 - 1$.
Your task is to specify one row of the stiffness matrix belonging to an interior node of the mesh.
10 Points

EXERCISE 2   COMPUTATION OF THE L^2-NORM IN DUNE
Implement a template function `l2norm()` that computes the L^2-norm of (mathematical) function. The mathematical function is represented by a class with the member function `evaluateGlobal` which has the point of evaluation as an input parameter and the result as an output parameter. The following code fragment gives and example of the implementation which is also found at the beginning of the file `uebung01.cc`.

```
template<typename GV, typename RF>
class Exp :
...
{
public:
  Exp(const GV& gv_)
    : BaseT(gv_)
  {}

  //! evaluate extended function on element
  inline void evaluateGlobal (const typename Traits::DomainType& x,
                              typename Traits::RangeType& y) const
  {
    y = exp(-(x*x));
  }

};
```

The template function `l2norm()` receives as a parameter a `GridView` and a function object. The integration should be done over all elements of the grid. A simple possibility would be to approximate the integral by the value of the function at the center of each element times the volume of each

element. A code fragment showing how to iterate over the grid and to compute the volume of the domain by calculating the volume of each element is found at the bottom of the file `uebung01.cc`.

Implement also the L^2-norm for `Discrete Grid Functions` (DGF). DGF's are functions that do not depend on a global coordinate x but on an element e and local coordinate x_e with respect to that element. In other words, they are defined element-wise. You will find in the file `uebung01.cc` a corresponding class and hints to the implementation such that you don't have iterate over the grid by yourself as in the analytical case.

Modify the boundary conditions and the right-hand side for the Poisson equation for a given analytical solution of the Poisson equation. Compute the L^2-error between the numerical and the analytical solution.

INSTALLATION OF DUNE ON YOUR OWN COMPUTER: If you have a computer with a Unix-like operating systems installed on, you can work on the exercises with your own computer. Just copy the shell script `build.sh` into a directory of your choice and run it.

```
scp psolv0XX@pool.iwr.uni-heidelberg.de:dune-parsolve-ss15/build.sh .
bash build.sh
```

For the installation you need a working internet connection. A documentation for the installation can be found at the homepage of DUNE. *10 Points*