TD12: Height maps and Mesh visualization

In this TP, the objective is to practice a bit with triangular mesh data-structures and 3D visualization using `DGtal` viewer. The final objective is to render height maps as 3D meshes with colorimetric information.

1 Preliminaries

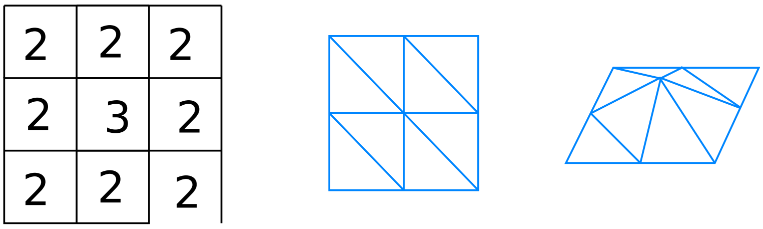
Visualization will be performed by `DGtal`. If you use the compiled library ("dcoeurjo" account), the viewer is enabled by default. If you use you own `DGtal` install, make sure that you have compiled the library with `WITH_QGLVIEWER` flag enables (e.g. `cmake .. -DWITH_QGLVIEWER=true`). You would need to have Qt and QGLViewer installed in your system.

Please also checkout the last release of the `DGtalSkel` folder. The file `image2mesh.cpp` gives examples of the `Viewer3D` usage.

First, compile this example and when executing it, you should see an OpenGL window with three triangles and two cubes. For this TP, you just need to know how to display triangles:

```
Z3i::RealPoint p1(1.0,0.0,0.0),
  p2(0.0,1.0,0.0),
  p3(0.0,0.0,1.0);
viewer.addTriangle(p1,p2,p3);
viewer  << Viewer3D<>::updateDisplay;
```

The idea is to construct a mesh from and image I such that the mesh vertices are given by 3D points $(i, j, I(i, j))$ for each point (i, j) in the image domain. As illustrated in Fig. 1, the mesh is constructed from a regular pattern from alternate triangles with respect to evenness of the x-coordinate for instance: $\{(i, j, I(i, j)), (i+1, j, I(i+1, j)), (i, j+1, I(i, j+1))\}$ or $\{(i, j, I(i, j)), (i+1, j-1, I(i+1, j-1)), (i+1, j, I(i+1, j))\}$



2	2	2
2	3	2
2	2	2

Figure 1: Image to height map mesh.

Please note that `image2mesh.cpp` also illustrates the loading of a PGM grayscale image.

2 Mesh data-structure and visualization

Question 1 From a PGM image, construct a Face-Vertex data structure: the Vertex array contains all point coordinates and the Face array contains the set of faces (triple of vertex indices).

3 Normal map rendering and curvature estimation