

# Workshop 2: mountains

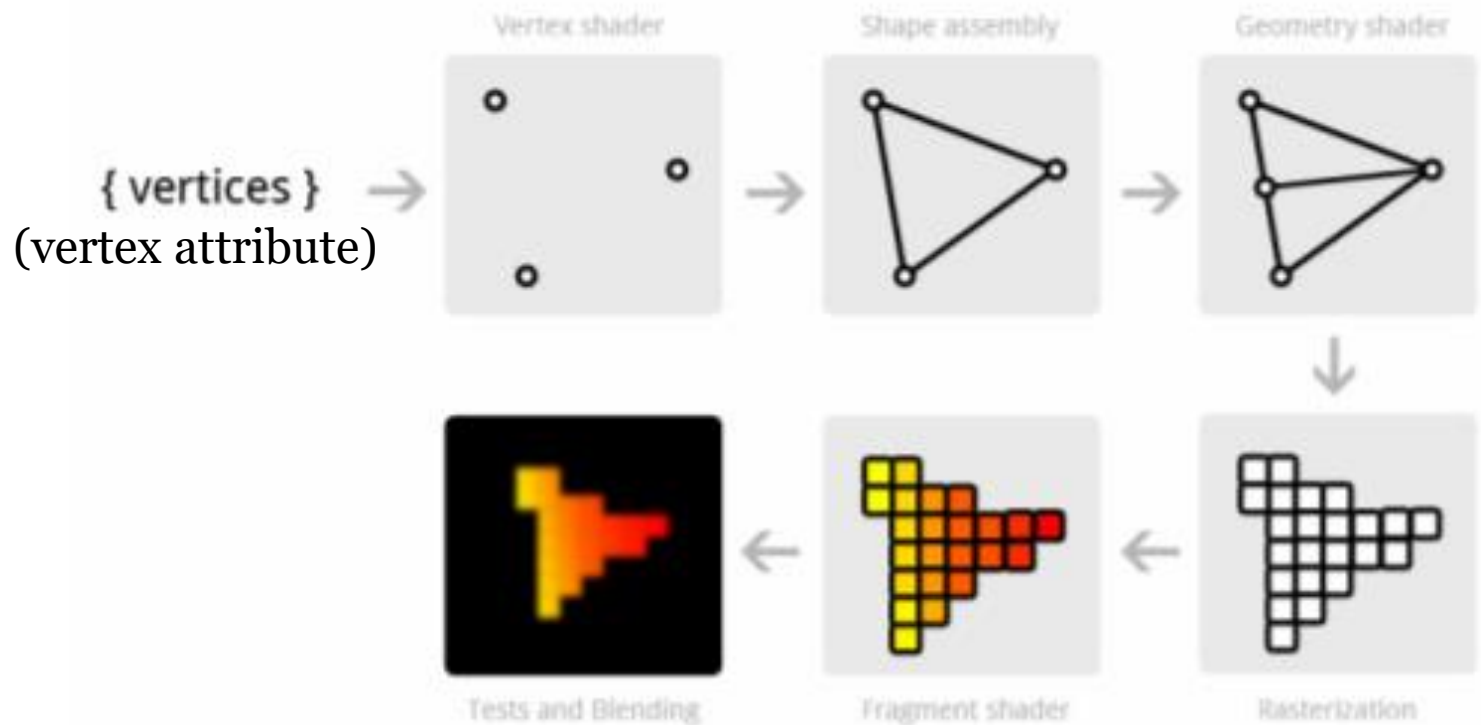
Hannah van Gemert | Leiden



**Universiteit  
Leiden**  
The Netherlands

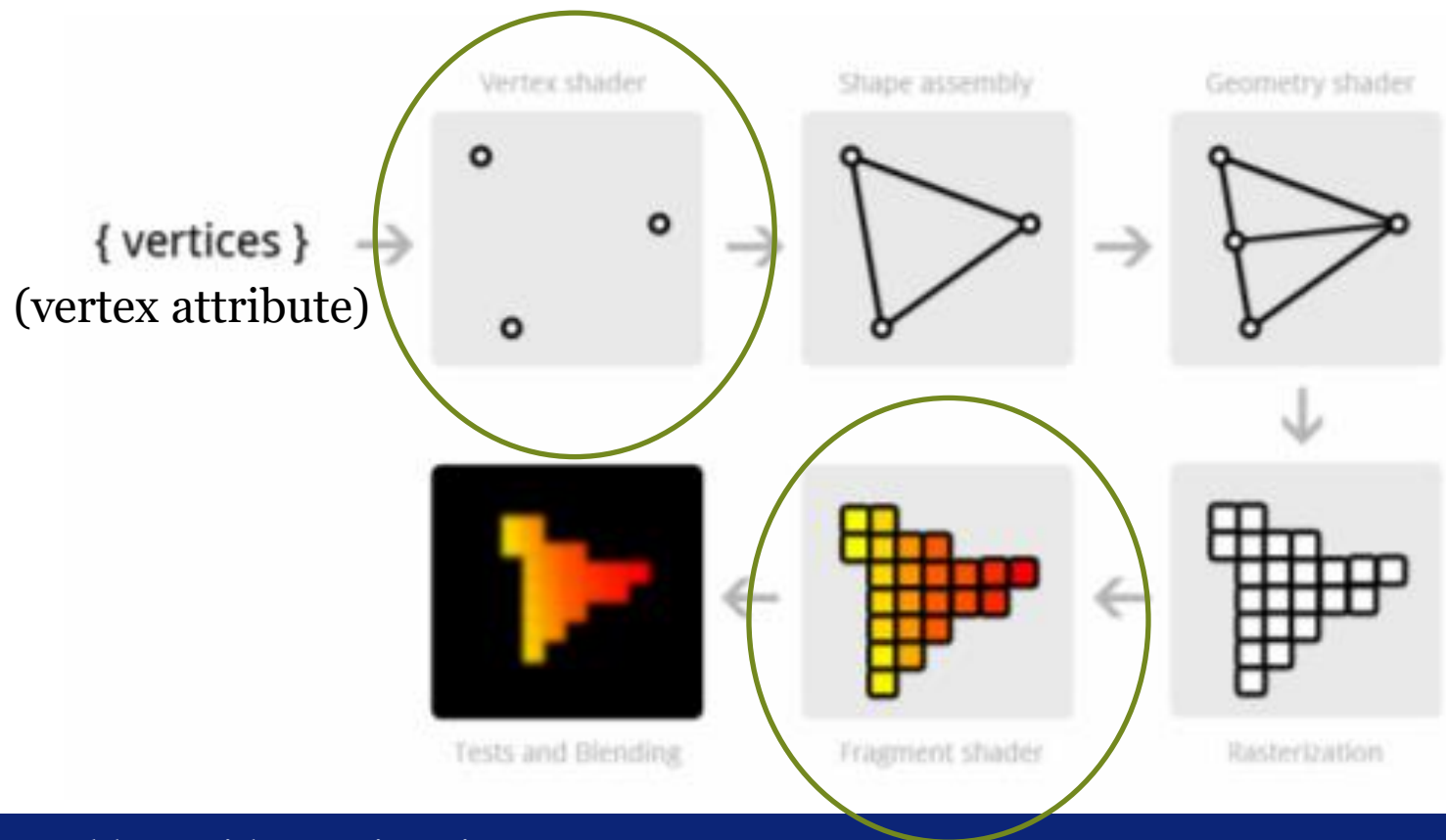
# In general, GLSL

- OpenGL Shader Language
- Manipulation of shapes in shaders
- Compiled during execution-time by the graphics driver



# In general, GLSL

- Shaders are the steps in the pipeline you can influence.
- Shader language is C-like.
- Loops and condition statements are supported.



## Vertex Shader (.vs):

- Retrieves data from the application
- Used to modify vertex positions (transformations, scaling, etc)
- Sends attributes to fragment shader. (out vertex should match fragment in)

## Fragment shader (.fs):

- Used to add colors and lighting
- Colors are a 4 vector: (r,g,b,w) where w is the white layer ratio
- Needs a vec4 output

```
#version version_number
in type in_variable_name;
in type in_variable_name;

out type out_variable_name;

uniform type uniform_name;

void main()
{
    // process input(s) and do some weird graphics stuff
    ...
    // output processed stuff to output variable
    out_variable_name = weird_stuff_we_processed;
}
```

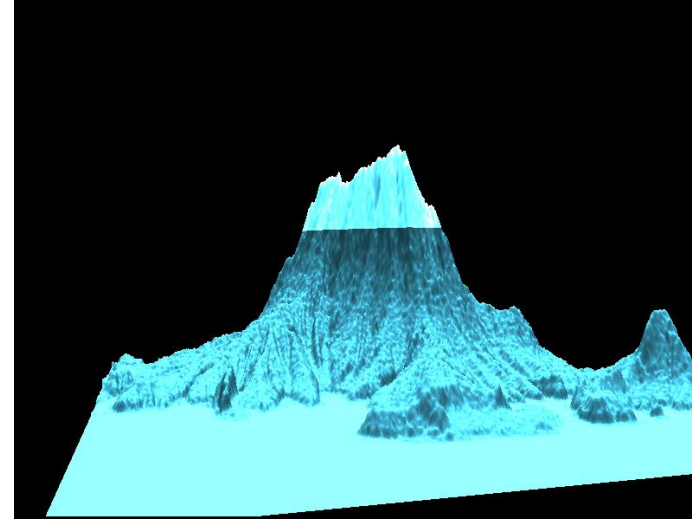
**Uniforms:** like attributes, but they are global -> access in any shader

- For this workshop you only need to edit the code from Workshop2.cpp and the fragment shader

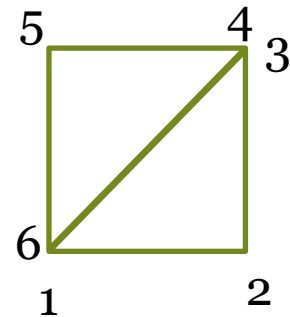
For this workshop you only need to edit the code from Workshop2.cpp and the fragment shader

## Problem 1:

- Assignment: get the heights from the heightmap in the `getTerrainvertexFunction` and link them to the correct positions indices.



- 513x513 are the dimensions of the heightmap.
- x and y are indices in the heightmap
- Positions and normal arrays have 512x512 squares, drawn by  
2 triangles=6 vertices. Dimensions: 512x512x6



For this workshop you only need to edit the code from Workshop2.cpp and the fragment shader

## Problem 2:

- Updatefunction and fragmentshader
- Wireframe: different mode of drawing. Look up `glPolygonMode()`
- You can query keystates using e.g. the boolean `inputstate.keydown[SDLK_LEFT]` (left key)
- Colors by (r,g,b,w) values

# Problem 3 (bonus):

- Rotate option: position rotates around mountain,  
target stays the same at all times
- First person, target and position changes:
  - Move back and forth
  - Look right and left
  - Look up and down
  - The geometry class has vectors and rotations

