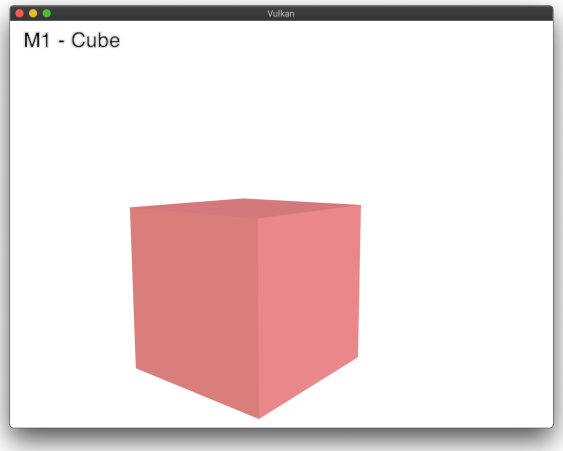
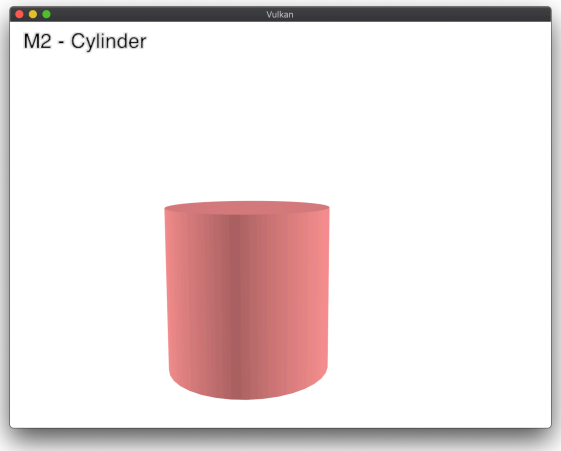
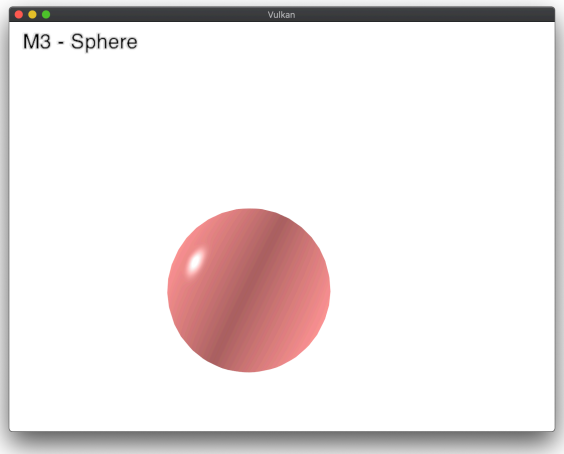
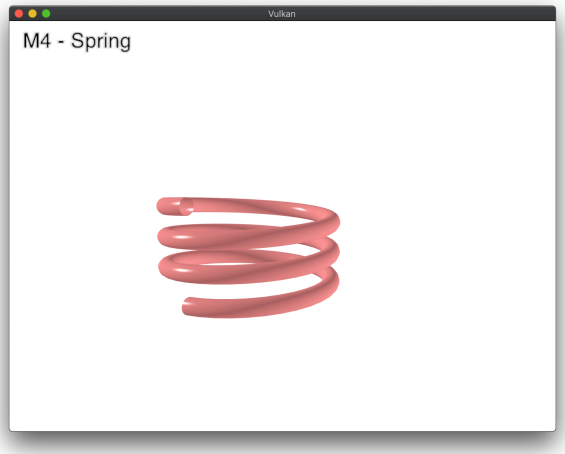
A18 – Smooth primitives

The Vulkan application whose source code is contained in file `Assignment18.cpp`, needs to show the models of a cube, a cylinder, a sphere and a spring, building them with the code contained `models.cpp`. In particular, the user implements the procedure `makeModels()`, that creates the primitive using indexed triangle lists. Vectors `M1_vertices` to `M4_vertices` should contain the components of the vertices of the primitives, while vectors `M1_indices` to `M4_indices`, should contain the indices for building the triangles. Below you can find a sample result for the four objects. The main difference with respect to Assignment09 is that in this case vertices are characterized by both the *position* and the *normal vector direction*, contained in the `Vertex` data structure. Expected results are the following:









You can move the view using the same keys as in *Assigment0*:

ESC – quit the application		SPACE BAR – move to the next projection | **X**: change view mode				
Q: roll left	**W**: forward	**E**: roll right	**R**: up		↑: look up	
A: left	**S**: backward	**D**: right	**F**: down	←: look left	↓: look down	→: look right

Pressing **X** you can change the view mode: the wireframe mode also shows the direction of the normal vector and can be usefull for debugging your code.