Excercise-Sheet 1
Excercise 1.3

Team: Marcel Weber
David Kirk

Data-Structures

struct Vector: (should represent a 3-D Vector)

It consists of three floating point numbers where each number represents a component of the 3-D Vector.

float x,y,z;

Also the << stream operator is overloaded, so it streams our x,y,z in the given order.

Mesh:

Due to the description of the excercise it has an id, number of vertices and faces which are represented by an integer. Furthermore it is described as an class which consits of a scaling which is represented by a floating point number. The Meshs center is represented by a vector and its name by a string.

So our Mesh has the following fields:
int id, vertices, faces
float scaling
string name
Vector center

To make sure our id is unique our Mesh class has a static integer which represents the next id which could allocated to a Mesh.
static int next_id

To instatiate our Mesh it has two constructors. One with the paramenters vertices, faces, scaling, name and vector and one without parameters (so to have a standard constructor), which sets our fields to the desired values. In addition our unique id is set to the value of next_id and next_id is incremented by one.
Mesh(string name, int vertices, int faces, float scaling, Vector center)

Also the << and >> stream operator is overloaded and streams id, name, faces, vertices, scaling and center in the given order.
The comparison == operator is overloaded which compares the id (because of its uniqueness).

SceneManager:

The SceneManager consists of a single linked list and its constructors, one with the parameter of a single linked list and one withour parameters.
The single linked list contains the mehs of a scene in the order as they are added to a scene.
forward-list<Mesh> meshs
SceneManager()
SceneManager(foward-list<Mesh> meshs)

Excercise-Sheet 1
Excercise 1.3

Team: Marcel Weber
David Kirk

As we want to add and delete meshs to our scene a SceneManager has a method add_mesh(Mesh mesh) which adds a mesh and the end of our lists of meshs and a method delete_mesh(int id) wich searchs for a mesh in our list by its id and then removes it from our list.