

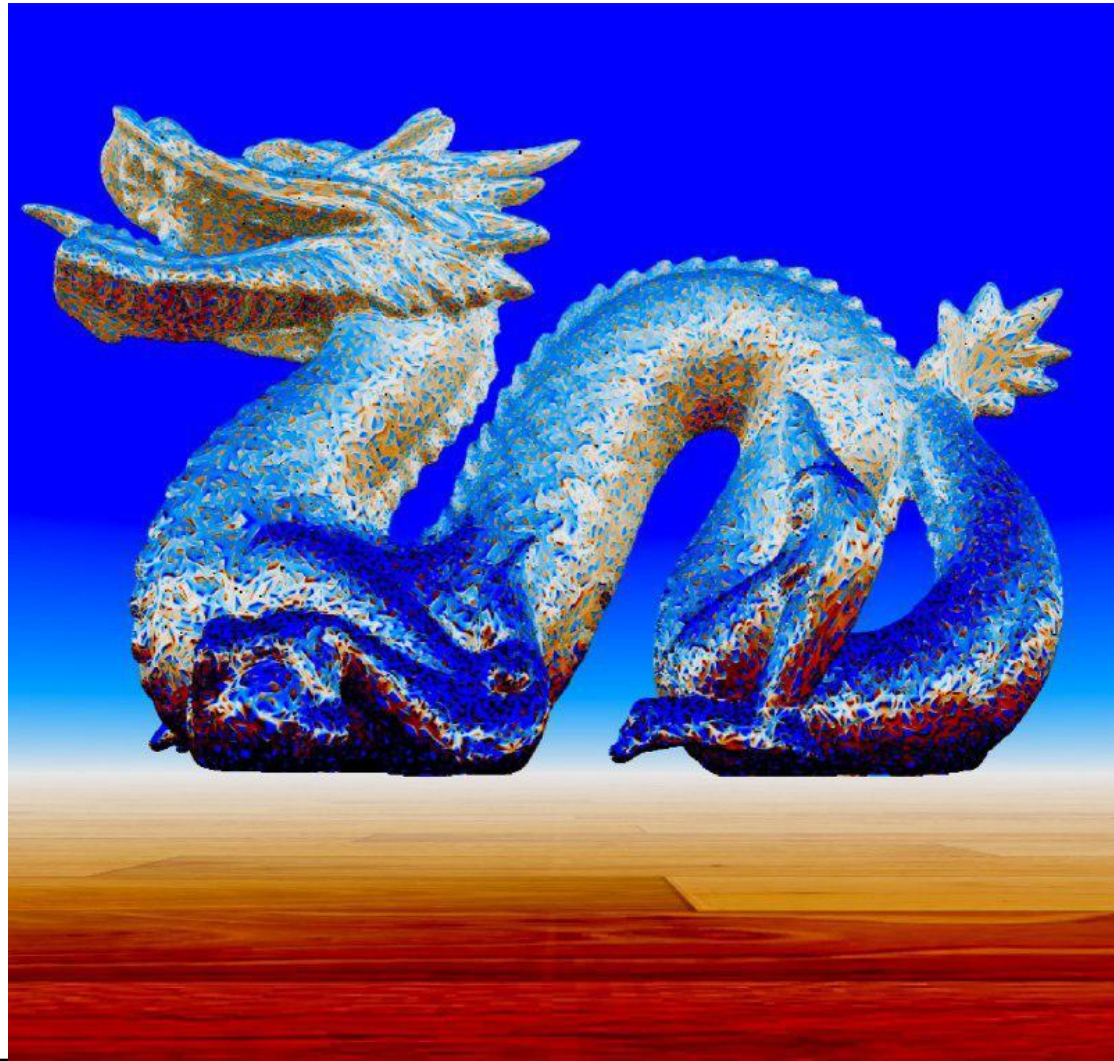
Presentation Raytracer

Group members:

- Rutger van den Berg 4060156
- Ymte Jan Broekhuizen 4246586
- Qu Chen 1256025
- Arend Jan de Graaff 4012534
- Max Groeneboom 4169298
- Rick van Hattem 1297295
- Ewoud van der Heide 1534033
- Leon Hoek 4021606
- Vince Kasanpawiro 4028880

Competition Image

Our product



Work Distribution

Distribution by team member

- Rutger: Debugging Linux, reflection, 3d models.
- Ymte Jan: Octree, multithreading, RealTime RayTracing, shadows, motion, ground textures, Phong shaders.
- Qu: Debugging Visual Studio, Ray-Intersection, presentation.
- Arend Jan: Windows operations, 3d models, presentation.
- Max: Ray-Intersection, 3d models, shadows, FPS counter
- Rick: Multithreading, 3d models, debugging MacOS, hosted git, timer, rewriting, background raytracing, FPS counter, RealTime RayTracing
- Ewoud: Colouring, 3d models, Debugging Linux
- Leon: Octree, debugging Linux, 3d models
- Vince: 3d models, presentation, debugging Visual Studio

Presentation

The actual topics

Start of Project

Raytracing

Normals

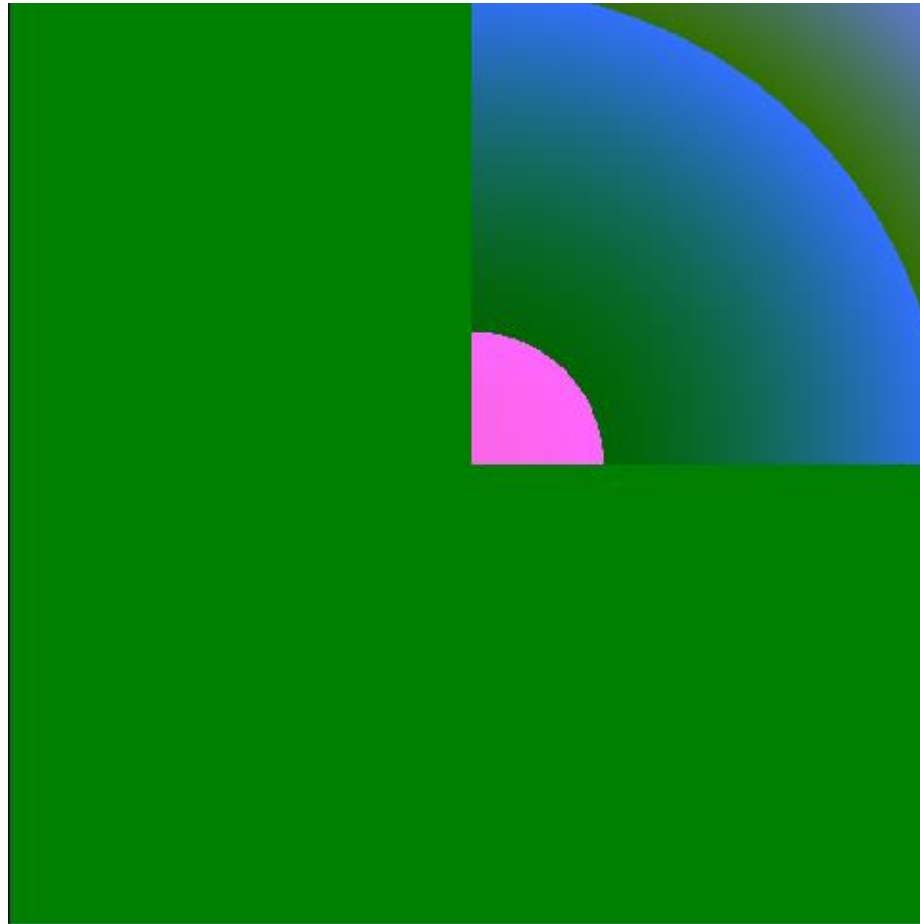
Duration

Extras

Demo

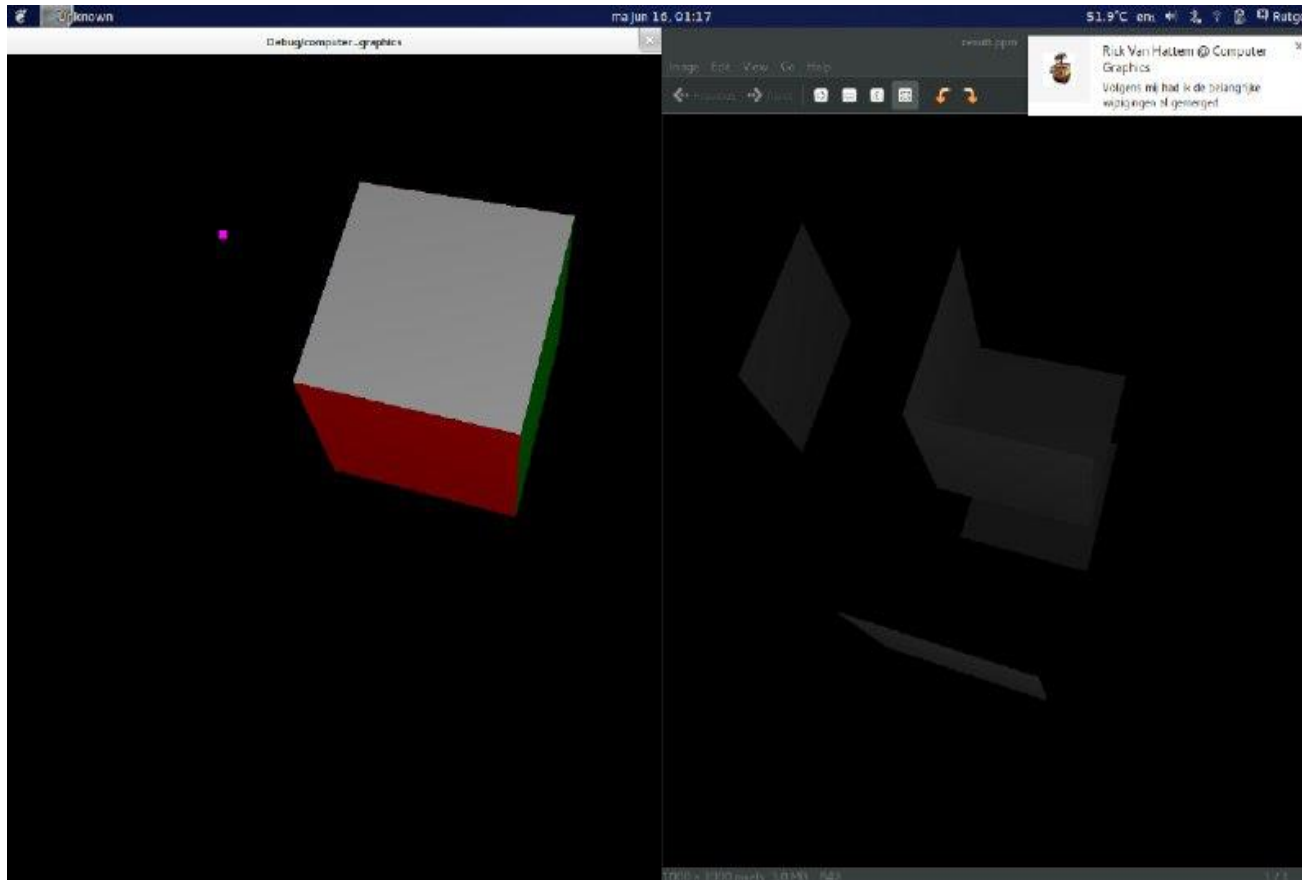
Psychedelic Art

Start of project



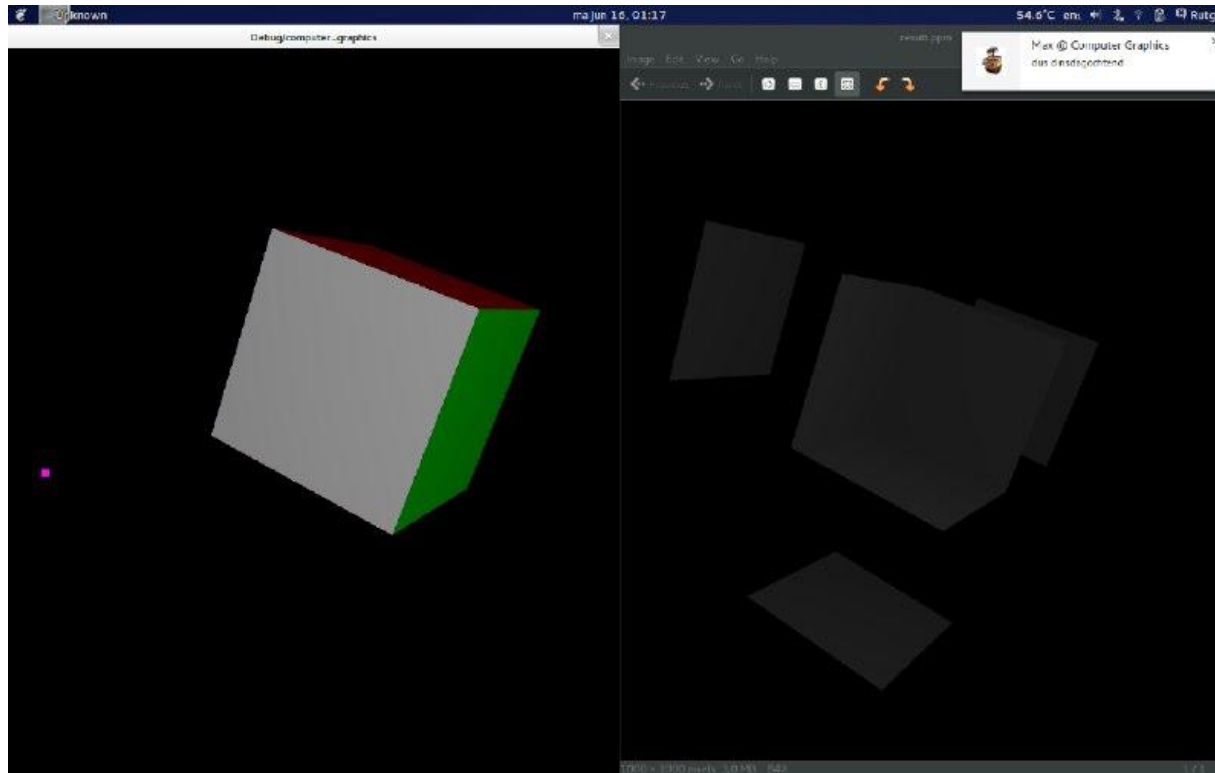
Cube Rendering

Start of project



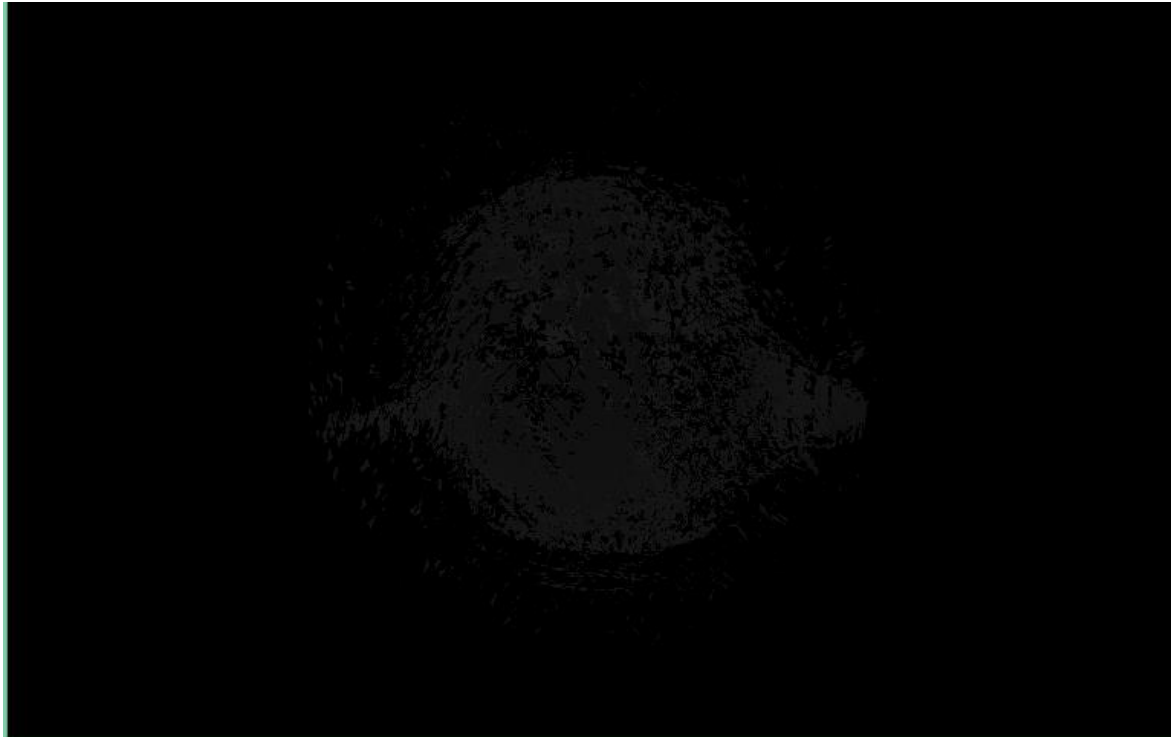
Cube Rendering

Start of project



Monkey Rendering

Start of project



Presentation

The actual topics

Start of Project

Raytracing

Normals

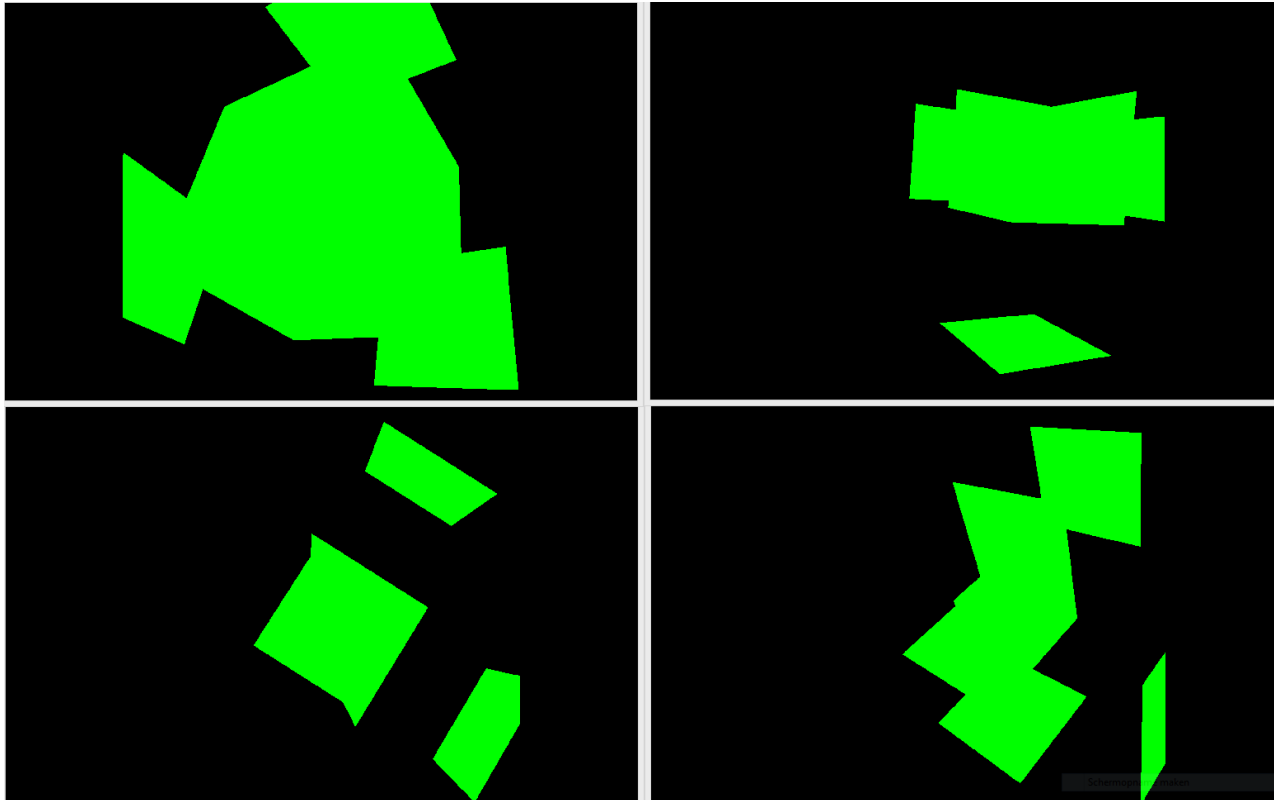
Duration

Extras

Demo

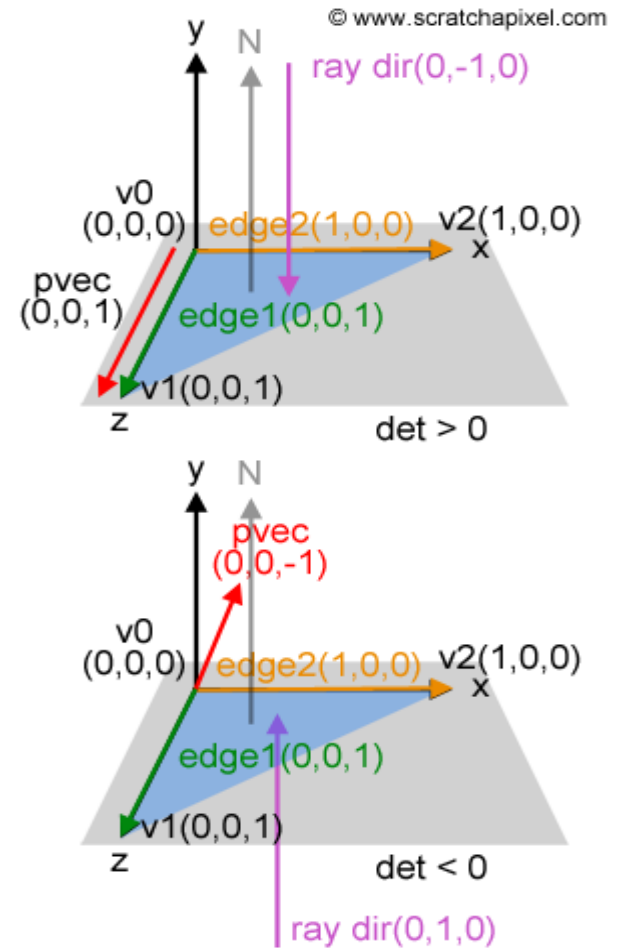
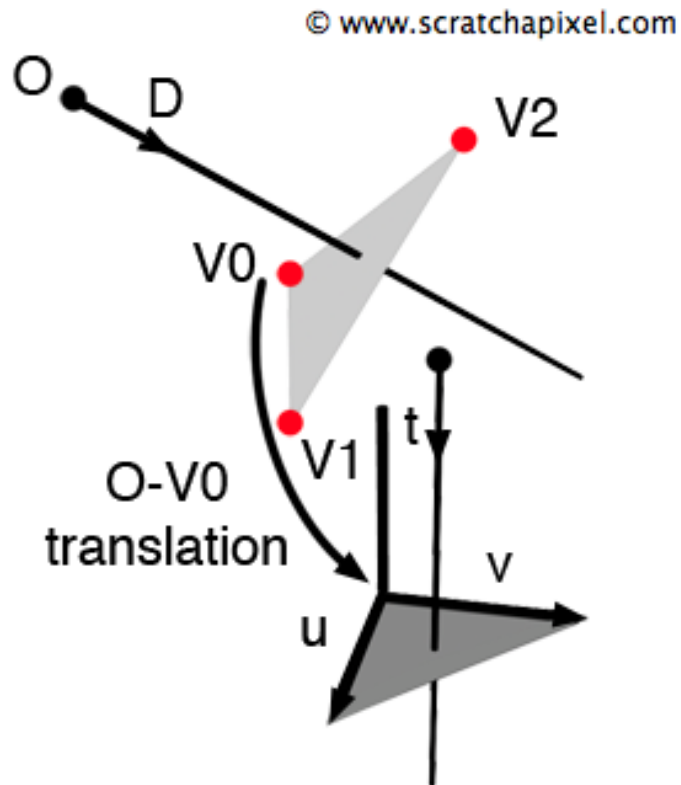
Geometric Ray Tracing

Raytracing



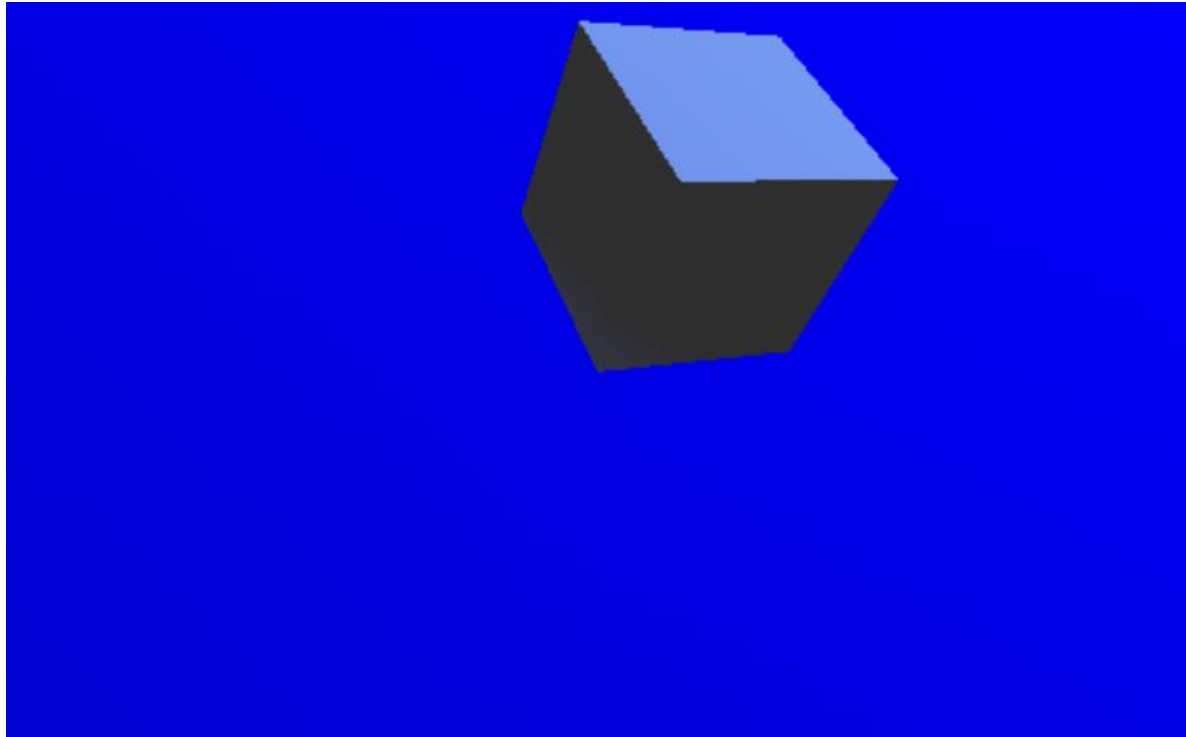
Solution

Raytracing - Möller Trumbore intersection method



Result

Raytracing - intersection working



Presentation

The actual topics

Start of Project

Raytracing

Normals

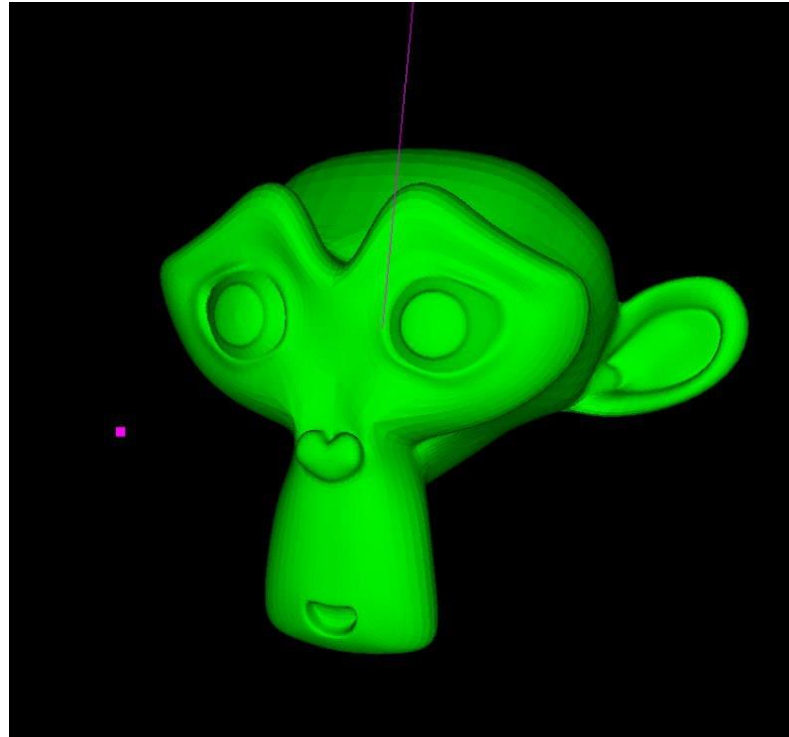
Duration

Extras

Demo

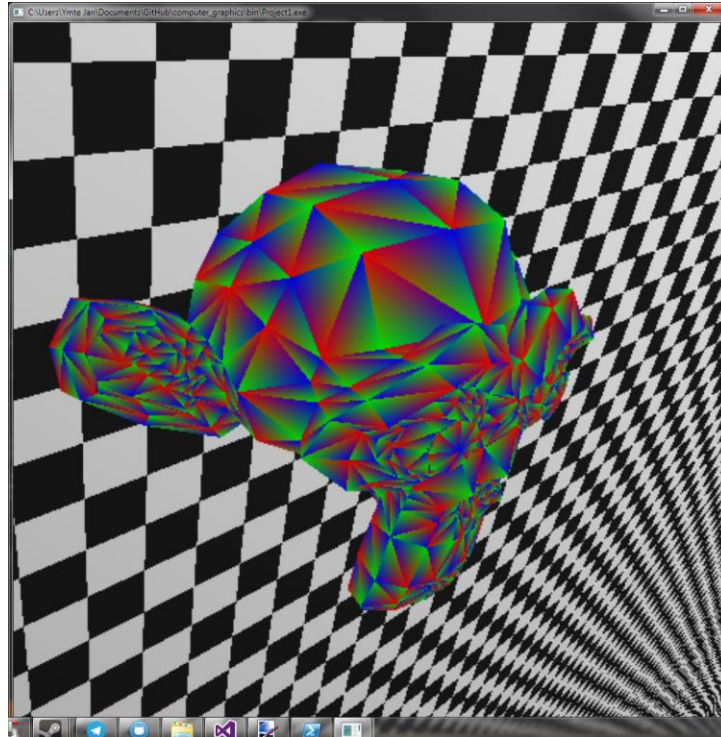
New Mesh

Normals - Blender's monkey



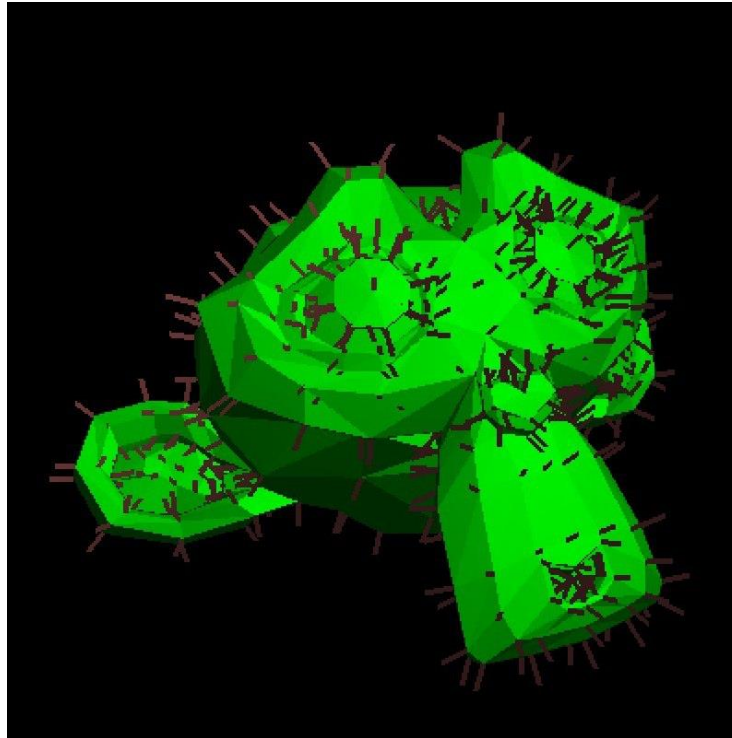
Blinn-Phong

Normals - used as colour



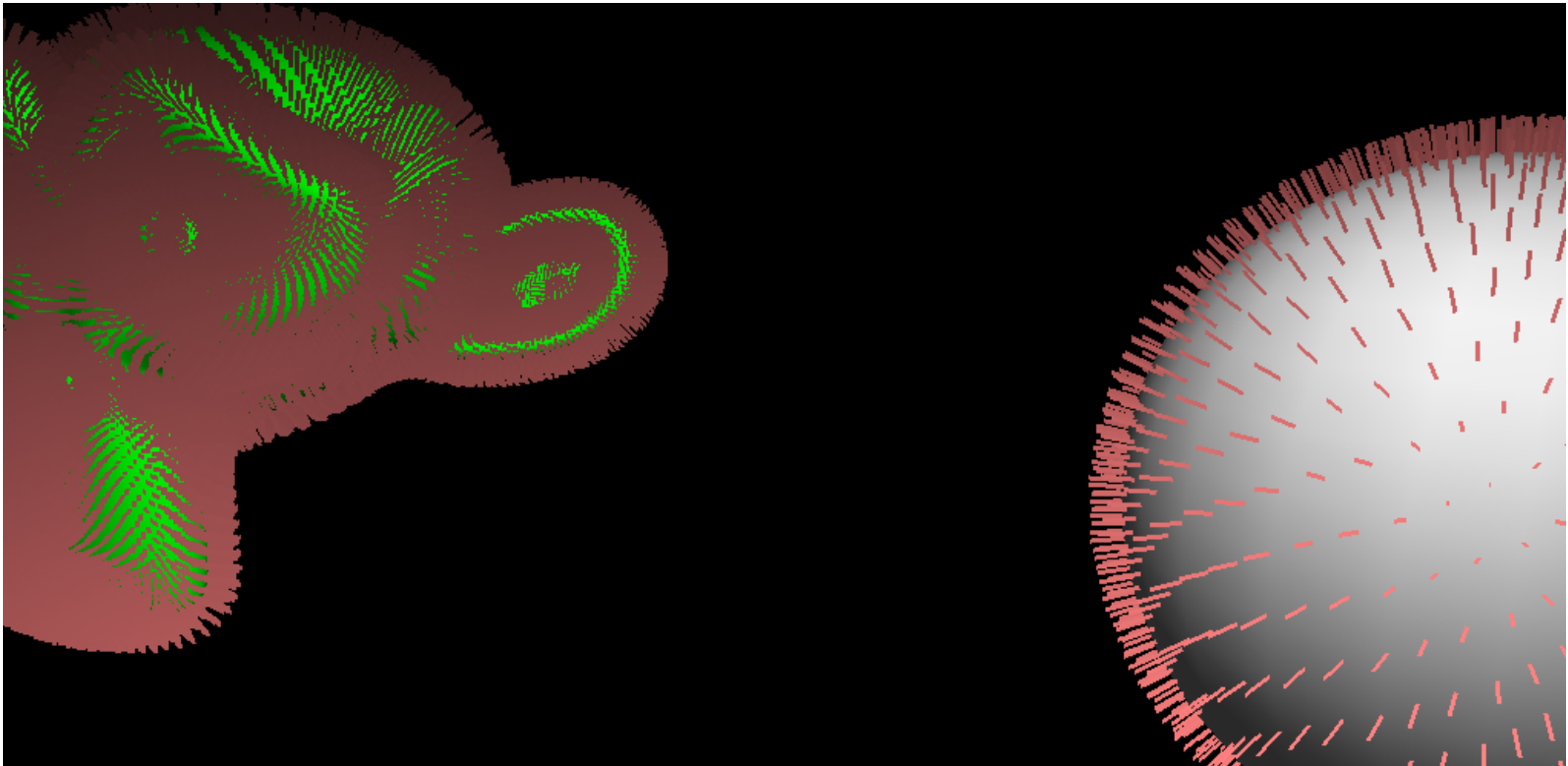
Actual normals

Normals – Using simpleMonkey



Actual normals

Normals – Applied to other models



Presentation

The actual topics

Start of Project

Raytracing

Normals

Duration

Extras

Demo

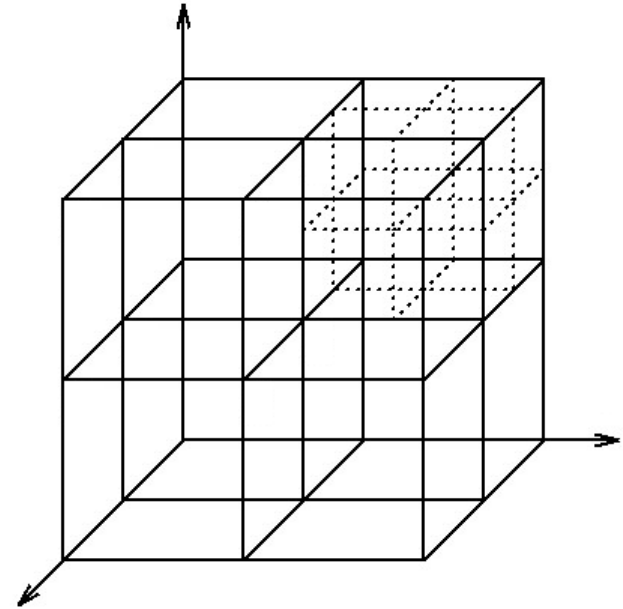
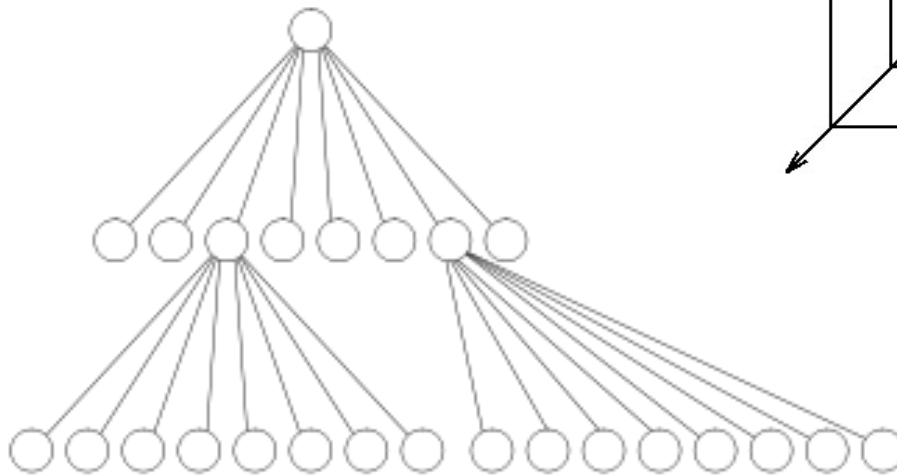
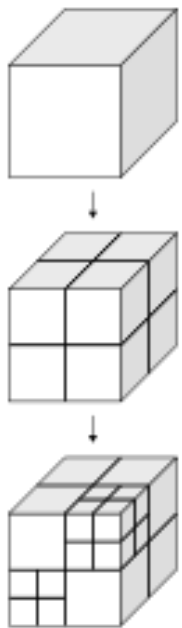
Challenges

Duration

```
Mesh file name: (<0: monkey, 1: cube, 2: dodgeColorTest, 3: simple_monkey>
You can omit the mesh/ path and the .obj extension.
0
Load material file mesh/cube.mtl
5 materials loaded.
key 98 pressed at 484,268
b pressed! The mouse was in location 484,268!
key 98 pressed at 611,202
b pressed! The mouse was in location 611,202!
key 114 pressed at 591,228
Raytracing
Rendering took 441 ms
Wrote raytrace output to result.bmp
r pressed! The mouse was in location 591,228!
-
```

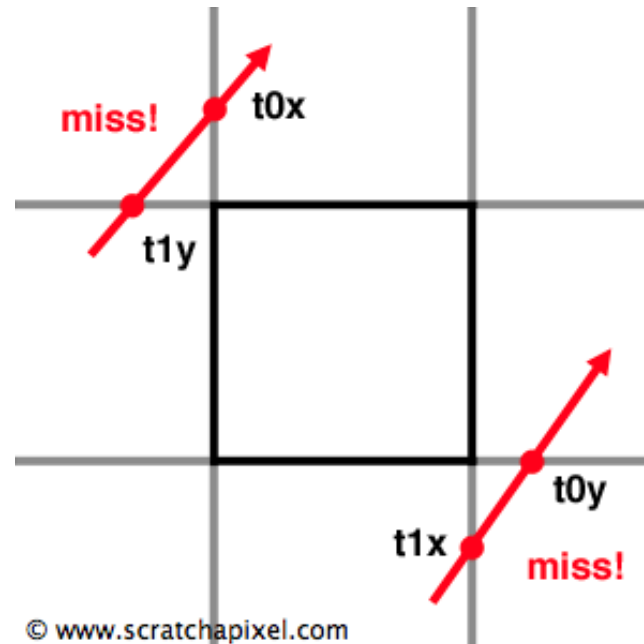
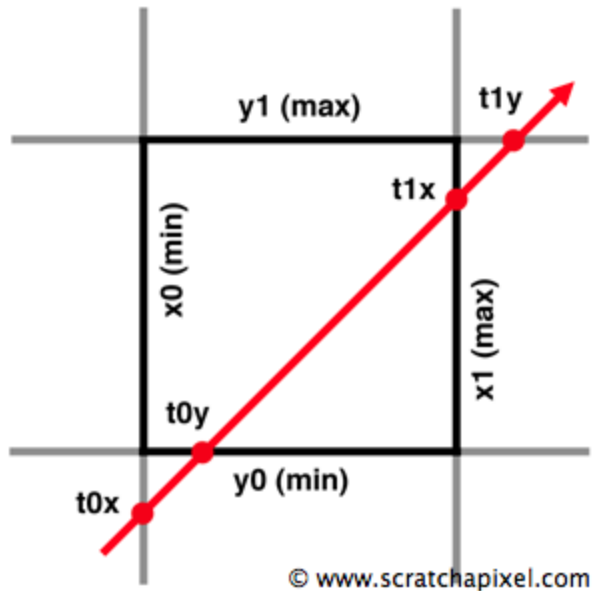
Octrees

Duration – possible solution



Octrees

Duration – possible solution



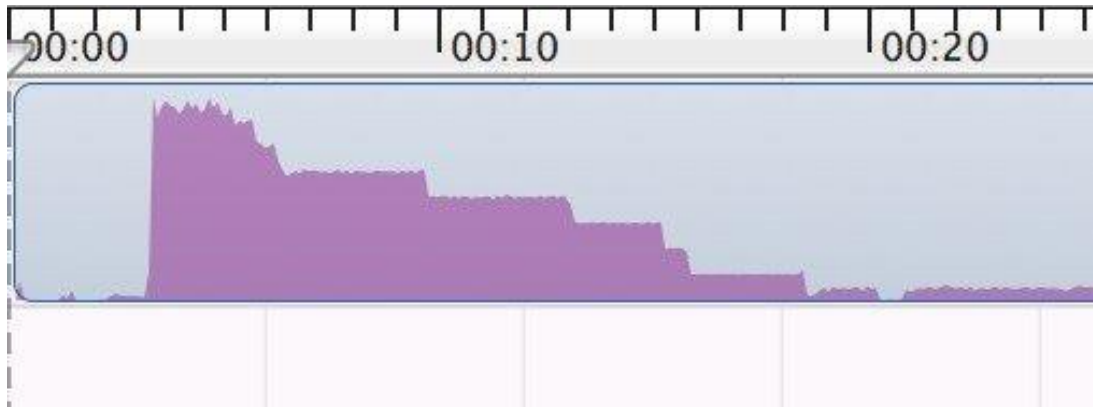
Octrees

Duration – result

```
Left button: turn in XY,  
Right button: translate in XY,  
Middle button: move along Z.  
Load material file mesh/cube.mtl  
5 materials loaded.  
Building tree!  
min: (0.0,0.0,0.0)  
max: (1.0,1.0,1.0)  
key 114 pressed at 324,298  
r pressed! The mouse was in location 324,298!  
Raytracing  
Rendering took 167 ms cpu seconds and 42 ms wall time  
Uploading to GPU took 4 ms  
Wrote raytrace output to result.bmp  
_
```

Multithreading

Duration – possible solution



```
25   ThreadPool(size_t);
26   template<class F, class ... Args>
27   auto enqueue(F&& f, Args&&... args)
28   -> std::future<typename std::result_of<F(Args...)>::type>;
29   ~ThreadPool();
30   inline bool running(){return !stop_};
31   inline void stop(){stop_ = true};
32 private:
33   // need to keep track of threads so we can join them
34   std::vector<std::thread> workers;
35   // the task queue
36   std::queue<std::function<void()>> > tasks;
37
38   // synchronization
39   std::mutex queue_mutex;
40   std::condition_variable condition;
41   bool stop_;
42 };
43
```

Multithreading

Duration – result, live rendering at 20+ fps

```
Preview 128x128@1 msaa using 8 threads
Raytrace 1024x1024@1 msaa using 8 threads
Window 1024x1024
Load material file mesh/cube.mtl
Unable to open ../textures/buffy256.ppm
Unable to open ../textures/angel256.ppm
Unable to open ../textures/willow256.ppm
Unable to open ../textures/xan
5 materials loaded.
Calculating normals...
Making triangles...
Done!
building tree with size 3
Rendering took 0.017 seconds
```


Presentation

The actual topics

Start of Project

Raytracing

Normals

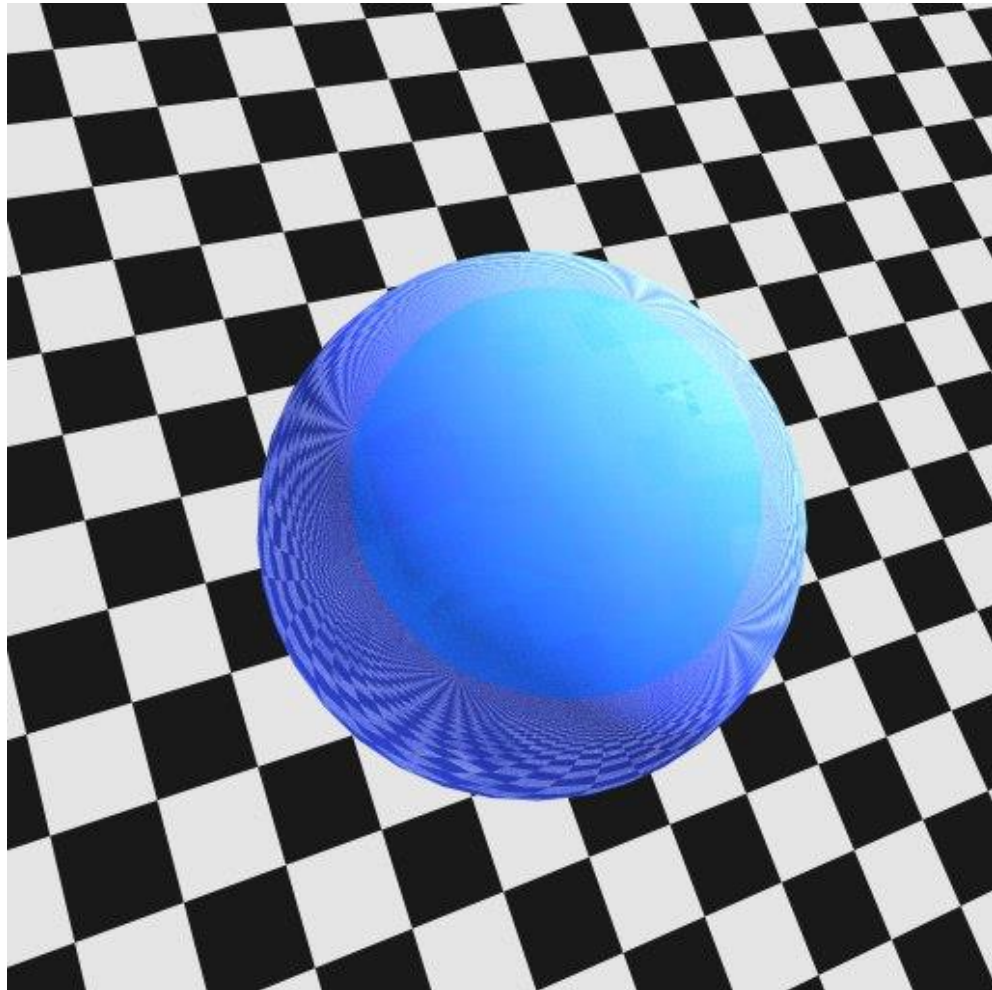
Duration

Extras

Demo

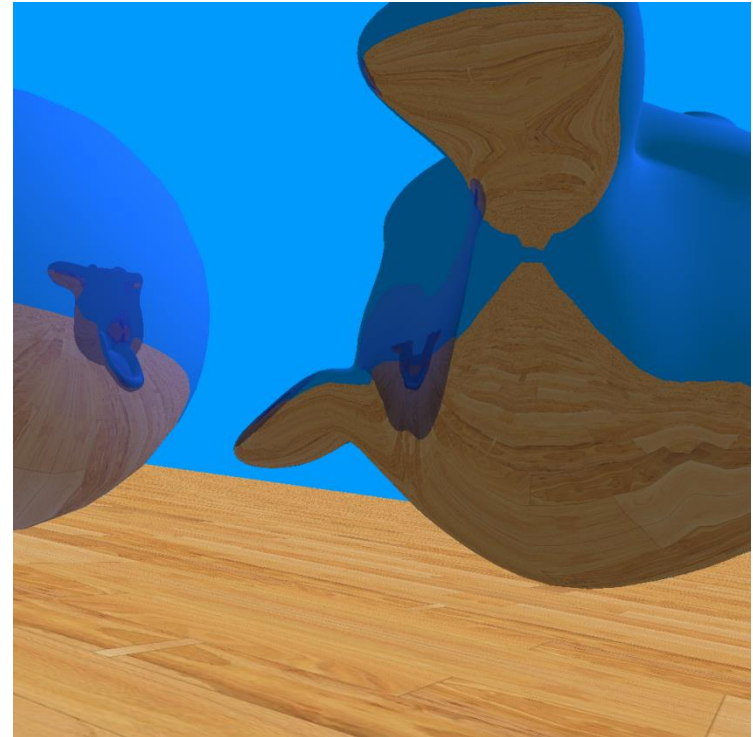
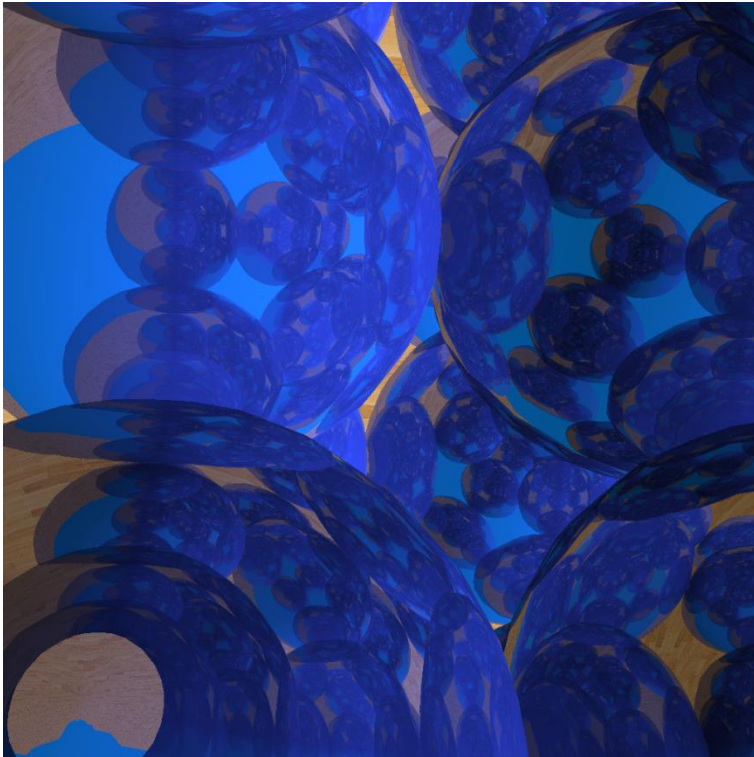
Reflection

Extras



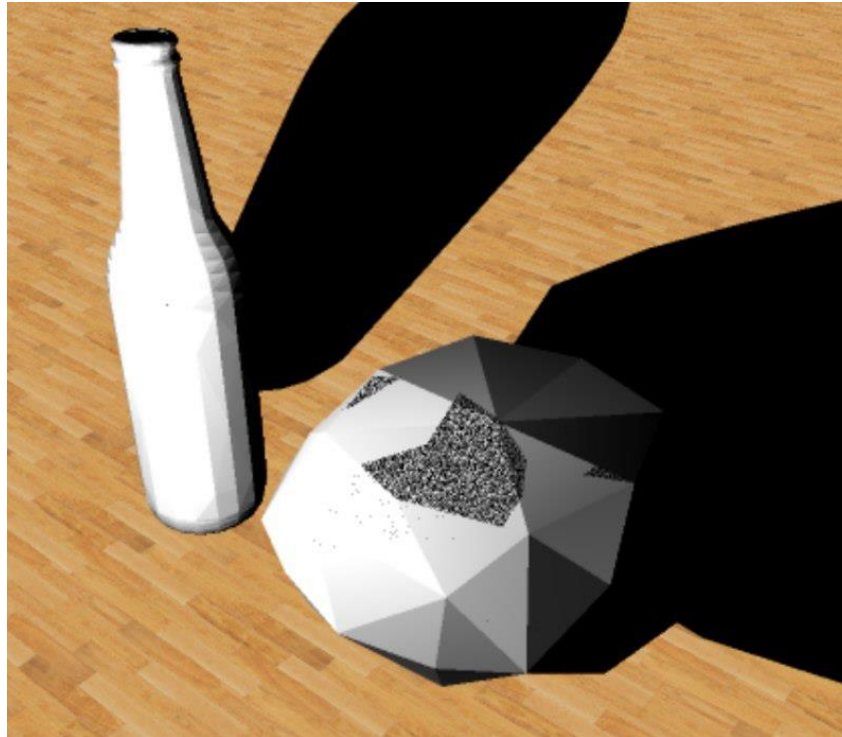
Reflection, onto other objects

Extras



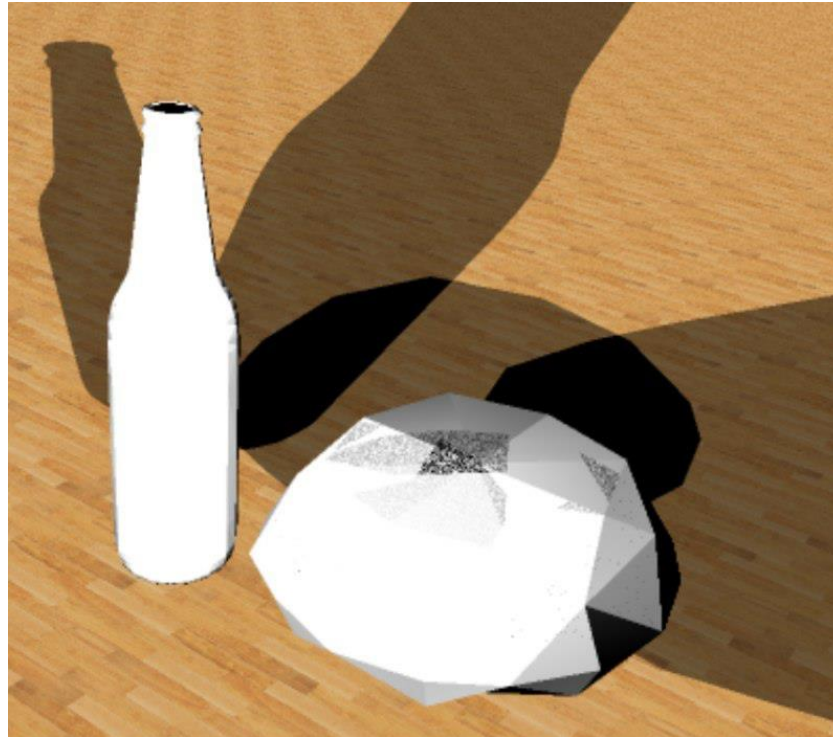
Shadows

Extras



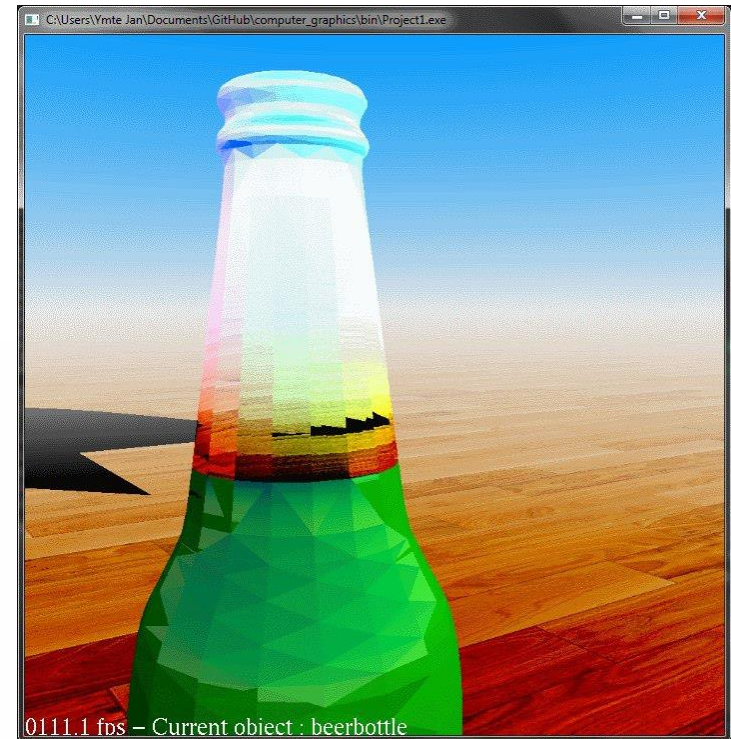
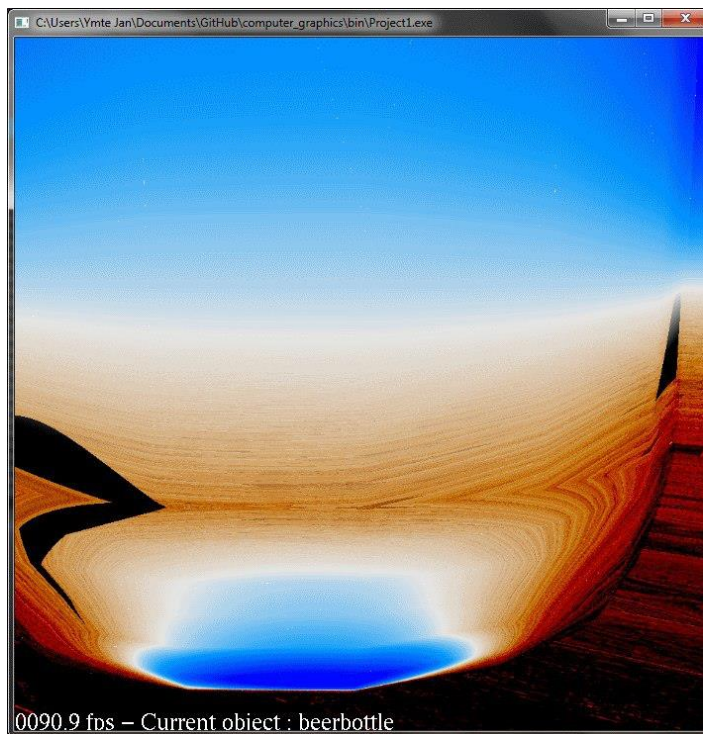
Shadows, multiple light sources

Extras



Refraction

Extras



Toggle functions

Extras

```
case '4':
    g_ambient = !g_ambient;
    printf("Set ambient to %d\n", g_ambient);
    break;
case '5':
    g_diffuse = !g_diffuse;
    printf("Set diffuse to %d\n", g_diffuse);
    break;
case '6':
    g_specular = !g_specular;
    printf("Set specular to %d\n", g_specular);
    break;
case '7':
    g_reflect = !g_reflect;
    printf("Set reflect to %d\n", g_reflect);
    break;
case '8':
    g_refract = !g_refract;
    printf("Set refract to %d\n", g_refract);
    break;
case '9':
    g_occlusion = !g_occlusion;
```

Presentation

The actual topics

Start of Project

Raytracing

Normals

Duration

Extras

Demo