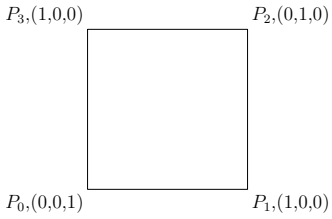
Prof. Dr. F. Sadlo, D. Aumiller *Heidelberg, November 14, 2016*

Exercise Sheet 4

Assignment 4.1 Barycentric Coordinates [4 Points]

1. How many barycentric coordinates does one need for a n-dimensional simplex? [*0.5 Points*]
2. Given the following square with (R,G,B) color values defined at the vertices:



   Imagine splitting the square into two triangles along the diagonal between P_3 and P_1, and interpolate the color values using barycentric interpolation. Briefly (!) describe the color variation between the corner points (see also the next problem below). [*1 Point*]
3. How does the color variation change if you split along the other diagonal instead? [*0.5 Points*]
4. The geometric interpretation of barycentric coordinates within triangles can be extended to rectangles as well. Come up with a way to directly calculate barycentric coordinates for a rectangle without prior subdivision into triangles. How many barycentric coordinates do you need and how are they defined for the vertices? Is it possible to apply this method to any arbitrary quadrangle? [*1 Point*]
5. How would you determine barycentric coordinates in a tetrahedron? [*1 Point*]

Assignment 4.2 Analytic Geometry [3.5 Points]

To generate an image with raytracing, we need so-called view rays. Given a view ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$, with a position vector $\mathbf{o} = (0, 0, 1)^T$ and the direction vector $\mathbf{d} = (1, 1, 0)^T$.

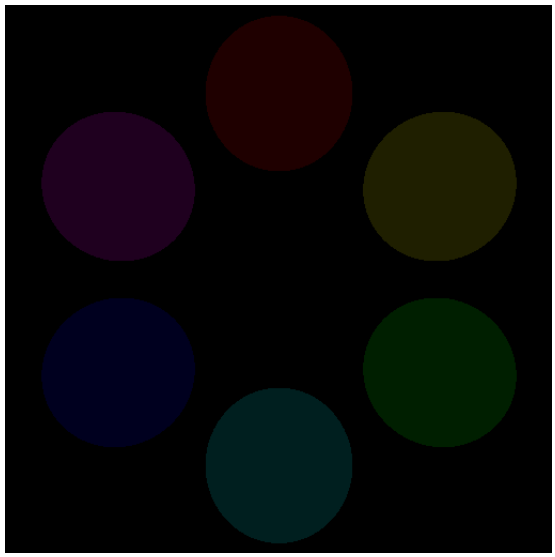
1. Calculate the intersection points between the view ray and a sphere with radius $r = 3$ and center $M = (5, 5, 2)$. Which one of the two intersections is toward the view ray and is therefore visible to the observer? [*1.5 Points*]
2. Calculate the intersection between the view ray and a plane with distance $d = 3$ from the origin and normal $n = (1, 1, 0)^T$. [*1 Point*]
3. Calculate the intersection between the view ray and the triangle $\triangle_{ABC}$ with $A = (6, 0, 0)$, $B = (0, 6, 0)$, and $C = (0, 0, 6)$. [*1 Point*]

Assignment 4.3 Ray Tracing [4.5 Points]

For this exercise, you should extend the code skeleton of a simple ray tracer, provided via Moodle. The ray tracer is already compiling, but only displaying a black image ("image.bmp"). Implement the intersection for rays with spheres in the methods

`Raytracer::Objects::Sphere::HitTest()` and
`Raytracer::Objects::Sphere::GetIntersection()`.

The files are located in `src/Raytracer/Objects/Sphere.cpp`. Hint: A correct solution should look like the image shown below, since this only uses ambient lighting. Other methods will be implemented in a later exercise.



Submission: November 21, 2016, 6:00 pm via Moodle