Exercises for the Lecture: “Architecture and Programming Models for GPUs and Coprocessors”
Exercise Sheet № 5

Priv.-Doz. Dr. rer. nat. Stefan Zellmann

SS 2022

5 Computergrafik

Ex. 5.1

We assume that triangles are defined by their vertex positions v_1, v_2, and v_3 in window coordinates. We further only consider geometry that is closed and inspected from the outside. That allows us to exclude many of the triangles—namely those that are facing away from the viewing position. We now define a convention based on the *winding order* of the vertex positions. v_1 and v_2 form a directed edge that subdivides the 2D plane into two half-planes. If v_3 is located in the left half-plane with respect to that subdivision, we say that the winding order is counterclockwise and we (arbitrarily) assume that the associated triangle is a *frontface* and thus visible. If, conversely, v_3 falls into the right half-plane and is thus a *backface*, we can exclude it from the subsequent shading computations. This method is commonly known as *backface culling*.

The winding order of the triangles can easily be determined by computing the sign of the determinant of the following matrix over the vertex positions in homogeneous coordinates:

$$\det T = \begin{vmatrix} v_{1x} & v_{2x} & v_{3x} \\ v_{1y} & v_{2y} & v_{3y} \\ 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} v_{2x} - v_{1x} & v_{3x} - v_{1x} \\ v_{2y} - v_{1y} & v_{3y} - v_{1y} \end{vmatrix}$$

For non-degenerate triangles (i.e., the three vertex positions aren't colinear) the sign is—per our convention—positive if we encountered a frontface, and conversely, the sign is negative if we encountered a backface.

a.)
Determine for the following triangles if they are front or backfaces:

$$\begin{aligned} T_1 &= \{v_1 = (2,2), v_2 = (4,2), v_3 = (4,4)\}, \\ T_2 &= \{v_1 = (5,5), v_2 = (2,10), v_3 = (10,5)\} \end{aligned}$$

b.)
The scan conversion algorithm by Pineda that we discussed in the lecture makes use of this principle. Show how the determinant of the matrix from **Ex. 5.1 a.)** is related to the edge equations (EE) $E_i(x, y)$ known from Pineda's algorithm, given two vertex positions v_i, v_{i+1} and a raster point $p = (x, y)$ in the 2D plane. Per convention, we assume that the edges can be obtained via $e_i = v_i - v_{i+1}$.

c.)
Consider the triangle $T = \{v_1, v_2, v_3\}$ with

$$\begin{aligned} e_1 &= v_1 - v_2 \\ e_2 &= v_2 - v_3 \\ e_3 &= v_3 - v_1 \end{aligned}$$

and the three edge equations

$$\begin{aligned} E_1(x, y) &= (x - v_{1x})e_{1y} - (y - v_{1y})e_{1x} \\ E_2(x, y) &= (x - v_{2x})e_{2y} - (y - v_{2y})e_{2x} \\ E_3(x, y) &= (x - v_{3x})e_{3y} - (y - v_{3y})e_{3x}. \end{aligned}$$

For a point (x, y) in the 2D plane, show that

$$2A(T) = E_1(x, y) + E_2(x, y) + E_3(x, y),$$

where $A(T)$ is the area of T.

d.)
Barycentric coordinates w.r.t. triangles are triples $b(p) = b(x, y) = (\lambda_1, \lambda_2, \lambda_3)$, where $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$ are weights that are associated with the triangle vertices. For each point (x, y) in the plane and the triangle $T = \{v_1, v_2, v_3\}$ there exists such a triple with:

$$\begin{aligned} \lambda_1 v_1 + \lambda_2 v_2 + \lambda_3 v_3 &= (x, y), \\ \lambda_1 + \lambda_2 + \lambda_3 &= 1 \end{aligned}$$

We further have $b(v_1) = (1, 0, 0)$, $b(v_2) = (0, 1, 0)$ and $b(v_3) = (0, 0, 1)$.

Show that for the edge equations from **c.)** and for $b(x, y)$ that

$$\begin{aligned} \lambda_1 &= \frac{E_2(x, y)}{2A(T)} \\ \lambda_2 &= \frac{E_3(x, y)}{2A(T)} \\ \lambda_3 &= \frac{E_1(x, y)}{2A(T)}, \end{aligned}$$

where $A(T)$ is the area of T.

The exercise sheet will be discussed on April 27, 2022.