Assignment in Computer Graphics II

– Assignment 8 –

Computer Graphics and
Multimedia Systems Group
Markus Kluge, Dmitri Presnov

Assignment 1 [2 Points] Work on data structures for polygon meshes

Develop pseudocode (using the reference labels `vert1, ...`) for Winged-Edge and Half-Edge data structures for the following tasks:

1. Given a polygon, all edges of this polygon have to be determined.
2. Given a vertex, all edges incident to this vertex have to be determined.

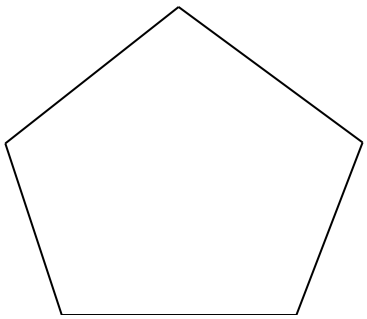
Annotation: For the data structure Half-Edge either the 'Outgoing'- or the 'Incoming'-edges have to be determined. It is not necessary to determine both edge types.

Annotation: Only the general case should be considered. No holes or mesh boarders.

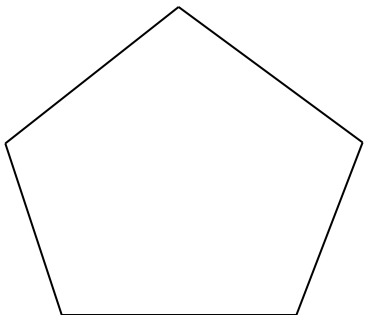
Annotation: Pseudocode should be based on the notation in the lecture. Also no solution should be longer than 12 rows. For longer solutions there will be no points.

Assignment 2 [1 Point] Subdivision Curves

Perform one step of a subdivision procedure for each of the pentagons below. Chaikin (left) and the 4-point method (right) by drawing the new polygon and its vertices (the exact position of the vertices are not relevant).



Chaikin



4-Point

Assignment 3 [2 Points] Subdivision

In this task you will implement two subdivision schemes. Download the framework `polygon-framework.zip` and take an initial look on the code.

All relevant files for this programming tasks, related to subdivision, can be found in the `Polygon` folder. In `Polygon/Polygon.hpp` you can find the abstract base class for all further subdivision classes. It provides an abstract member function `evalConstruct` which you will have to implement for all derived classes. Please study this class, read the comments and try to understand it.

To build the project in your preferred development environment use the included CMake project ("CMakeLists.txt"). CMake can be downloaded from the following website: `http://www.cmake.org/`. Use the instructions on the page
`http://www.cmake.org/cmake/help/runningcmake.html` and the tutorial page to create the project.

1. Implement the 4 point subdivision scheme in `Polygon/4Point.hpp` and `Polygon/4Point.cpp`.
   - `evalConstruct`: In this function you have to compute and store iterations of the 4 point subdivision scheme. The results have to be stored in the member variable `contructedPoints_`. It is a C++ `std::map` of point containers where the key value represents the algorithm iteration e.g. `contructedPoints_.at(0)` returns a `PointContainer` that contains the input points.
2. Implement the Chaikin subdivision scheme in `Polygon/Chaikin.hpp` and `Polygon/Chaikin.cpp`.
   - `evalConstruct`: In this function you have to compute and store the iterations of the Chaikin subdivision scheme. The results have to be stored in the member variable `contructedPoints_`. It is a C++ `std::map` of point containers where the key value represents the algorithm level e.g. `contructedPoints_.at(0)` returns a `PointContainer` that contains the input points.

Further hints can be found in the correpsonding source files.

Total points after sheet 8: 39 of 70.

Hand in: Until 11.06.2018 12:00 o'clock in mailbox of our chair (next to room 7115) and the programming assignment via e-mail (`jan.mussmann@student.uni-siegen.de`)..