TECHNISCHE UNIVERSITÄT DARMSTADT

Geometrische Methoden des CAD / CAE

WS2016/2017

Prof. Dr. André Stork

Thu Huong Luu

Felix Schuwirth

Telefon: 06151/155-467

E-Mail: Andre.Stork@igd.fraunhofer.de

E-Mail: Thu.Huong.Luu@igd.fraunhofer.de

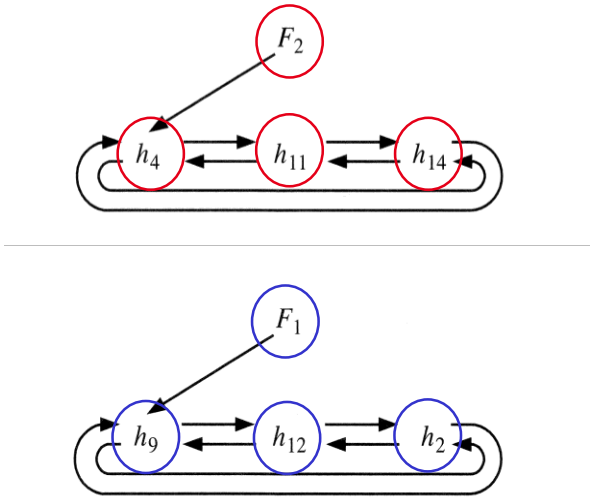
E-Mail: Felix.Schuwirth@igd.fraunhofer.de

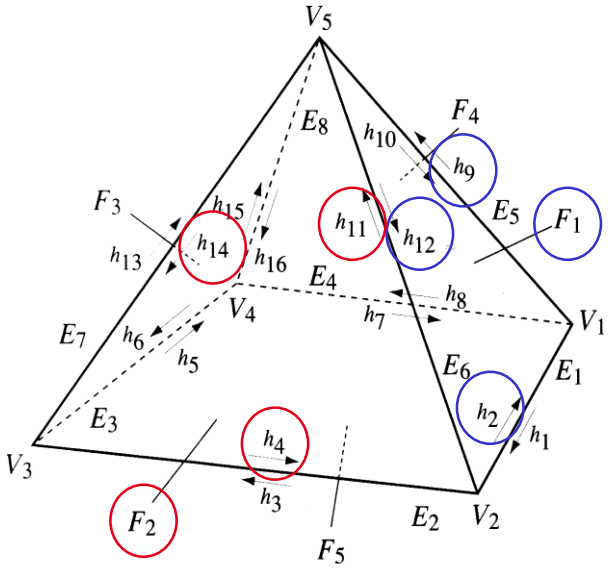
Programmierpraktikum 1

TECHNISCHE UNIVERSITÄT DARMSTADT

Aufgabe 0: Vorbereitung

Aufgabe 1: Erzeugung und Visualisierung einer Halbkanten-Datenstruktur





Quelle: Kunwoo Lee, Principles of CAD / CAM / CAE Systems

Klassenstruktur

TECHNISCHE UNIVERSITÄT DARMSTADT

HalfEdgeDSElements

- Data structure elements (minimal extract of the data structure presented in the slides)

```
struct Vertex
{
    HalfEdge* outgoingHE;      // can be any outgoing half edge
    Vec3f coordinates;         // stores the 3d coordinates of the vertex
    …
};
```

HalfEdgeDS

- Owns all created data structure elements (unsorted)

```
std::list<Face*> faces;
std::list<Edge*> edges;
std::list<Vertex*> vertices;
…
```

- Handles creating, verifying and modifying a half edge data structure
- Implement Euler Operators
- Verify with Euler-Poincaré Formula

Klassenstruktur

TECHNISCHE UNIVERSITÄT DARMSTADT

Main

- Instance of our data structure
  - `HalfEdgeDS heDS;`
- The currently active half edge
  - `HalfEdge* activeHE;`

- Handel Keyboard Events
  - `void keyPressed(unsigned char key, int x, int y)`
- Render Scene
  - `void renderScene()`
  - Executed if `glutPostRedisplay()` is called

HalfEdgeDSRendering

- Render Data Structure Elements with OpenGL

OpenGL - Beispiele

TECHNISCHE UNIVERSITÄT DARMSTADT

```
Vec3f colorGreen(0.0f, 1.0f, 0.0f);
Vec3f colorRed(1.0f, 0.0f, 0.0f);
Vec3f p1(1.0f, 2.0f, 4.0f);
Vec3f p2(2.0f, 0.0f, 1.0f);

//drawing a green line from p1 to p2:
glBegin(GL_LINES);
glColor3fv(&colorGreen.x);
glVertex3fv(&p1.x);
glVertex3fv(&p2.x);
glEnd();

//drawing two red points at p1 and p2:
glBegin(GL_POINTS);
glColor3fv(&colorRed.x);
glVertex3fv(&p1.x);
glVertex3fv(&p2.x);
glEnd();
```

HalfEdgeDS:createDefaultObject()

```
void HalfEdgeDS::createDefaultObject()
{
      // create example elements.
    Vertex* v1 = new Vertex;
    Vertex* v2 = new Vertex;
    HalfEdge* he1 = new HalfEdge;
    HalfEdge* he2 = new HalfEdge;
    Edge* e = new Edge;
    // set up connections
    v1->coordinates = Vec3f(1.0f, 2.0f, 3.0f);
    v2->coordinates = Vec3f(3.0f, 2.0f, 1.0f);
    he1->startV = v1;
    he1->nextHE = he2;
    he2->startV = v2;
    he2->nextHE = he1;
    e->he1 = he1;
    e->he2 = he2;
    // add elements to the lists
    vertices.push_back(v1);
    vertices.push_back(v2);
    halfEdges.push_back(he1);
    halfEdges.push_back(he2);
    edges.push_back(e);
    …
}
```

Attention: Don‘t forget to initialize pointers!

Attention: For every "new" we need a "delete“! If an element is added to the according list, it is deleted automatically within `clearDS()`.

Ansonsten

TECHNISCHE UNIVERSITÄT DARMSTADT

- Fragen können im Moodle Diskussionsforum gestellt werden