Conceptual Questions:

(5 pts) When quadrangulating a face during Catmull-Clark subdivision, what information must you temporarily store in order to properly link your half-edge pointers without creating inaccessible edges?

(You can see the line 651 of mesh.cpp), I created a vector to store the information that the first vertex of each new quadrangle. Because when we are creating the new face from edge HE, the start point of HE is not pointed by the previous halfedge, because that halfedge is pointing to the new halfedge created in the previous new face. Thus we have to store this temporary information to properly link the last halfedge with the startpoint.

(5 pts) When extruding a face, what operation must be performed after all edges have been turned into quadrangles? What is an edge case in the scope of this operation?

To realize extrude function, first I extrude the startedge of the face, it will create 2 vertices, 4 half-edges and 1 face. Second, if the face is not the last one, it will create 1 vertex, 4 half-edges and 1 face, which it will use the vertex of the face before it. Then, the last face will create 4 half-edges and 1 face, no vertex.

Finally, we just need to set the sym pointers of all the new edges created by the extrude.

I didn't find any edge case in the scope of this operation, if all of the face is connected to the other one, which means there is no half-edge point to the nullptr.

Code Requirement:

All done.

Extra Credit:

5.1 there is a button called "Planarity Testing" to do this.

5.2. as follow:

Add int Sharpness (0-3) to the Vertex;

Add bool Sharp to the halfedge and face;

If a Face is set as Sharp, then set its every vertices to sharpness + 3;

If a halfedge is set as Sharp, then set its sym to sharp and let its endpoints' sharpness++

If a vertex is set as Sharp, then set its sharpness = 3;

Subdivision:

When adding face point,

if face is sharp, the vertex’s sharpness = 3;

if all edges are sharp, the vertex should be sharpness is 3;

When adding new edges and new faces,

If the face is sharp, then the new faces are sharp;

If all edges are sharp, then the new edges should be sharp, too;

When adding a vertex to an edge,

if this edge->face is Sharp, the added vertex’s sharpness = 3;

else if this edge is Sharp, the added vertex’s sharpness = 2 and 2 new halfedges are also set as Sharp;

When reposition,

If vertex’s sharpness == 1, then as normal.

Else if vertex’s sharpness == 2, its smoothed position is (0.75 * original) + (0.125 * edge_1_endpoint) + (0.125 * edge_2_endpoint).

Else if vertex’s sharpness == 3, it will not change its position at all.

Just for testing:

Only set three button, “SetSharpVert”, “SetSharpEdge”, “SetSharpFace” to test, and they cannot be canceled(I haven’t implemented this function yet).

To be more easy to test, I set the “SetSharpFace” to sharp Faces[1] and Faces[4].

The “SetSharpFace” function may have some problems, I don’t know why.

5.4 :

There are 2 buttons to realize following functions:

Beveling the edge:

Reposition each vertex by minus 0.3*face_normal, then extrude and the endpoints of this edge’s reposition should be adjusted, besides add 0.3*face_noraml, the startpoint’s position should also add 0.3* (startpoint.prev - stratpoint), the endpoint should also add 0.3 * (endpoint.next - endpoint)

Beveling the face:

Reposition each vertex by minus 0.3 * face_normal, then calculate new position of each new point. And extrude face to new position.