

Programming Lab 12

Implement a class for clipping line segments and polygons to a convex polygon. I will supply you with the header file Clip.h, which has public and private declarations for this class:

```
class Clip {
public:
    Clip(void);
    Clip(int n, const Hcoords* clip_to_hspaces);
    bool operator()(Point& A, Point& B);
    bool operator()(std::vector<Point>& verts_to_clip);
private:
    std::vector<Hcoords> hspaces;
    std::vector<Point> temp_points;
};
```

(the file Affine.h and the standard header file vector have been included). The member functions are described below. You are free to use the private data members as you see fit.

Clip() – (default constructor) creates an instance of the class for clipping to the standard cube; i.e., the cube with vertices (1 1 1).

Clip(n, clip_to_hspaces) – (non-default constructor) creates an instance of the class for clipping to the convex polyhedron composed of n half-spaces, which are stored in the array clip_to_hspaces.

operator(A,B) – computes the result of clipping the line segment AB to the convex polyhedron. If there is no intersection between the line segment and the polyhedron, the return value is false. Otherwise, the return value is true, and the values of A and B are modified to hold the endpoints of the clipped line segment.

operator(verts_to_clip) – computes the result of clipping the polygon P with vertices stored in the vector verts_to_clip to the convex polyhedron. If there is no intersection between P and the convex polyhedron, the return value is false. Otherwise, the return value is true, and the vector verts_to_clip is modified to hold the vertices of the clipped polygon.

For the line clipping function, you are expected to use the parametric line clipping algorithm discussed in class. And for the polygon clipping function, you are to use the Sutherland Hodgman polygon clipping algorithm. Part of your grade will be for the efficiency of your implementations.

Your submission should consist of a single file: the implementation file Clip.cpp. You may only include the header files Affine.h, HalfSpace.h, Clip.h, and the standard C++ header file vector.