---

Purpose:

- Apply basic concept of Object Oriented Programming in C++
- Ability to code Data Structure.
- Work with multiple file.
- Test problem solving skill.

Problem:

The doubly connected edge list (DCEL), also known as half-edge data structure, is a data structure to represent an embedding of a planar graph in the plane, and polytopes in 3D. This data structure provides efficient[quantify] manipulation of the topological information associated with the objects in question (vertices, edges, faces). It is used in many algorithms of computational geometry to handle polygonal subdivisions of the plane, commonly called planar straight-line graphs (PSLG). For example, a Voronoi diagram is commonly represented by a DCEL inside a bounding box.
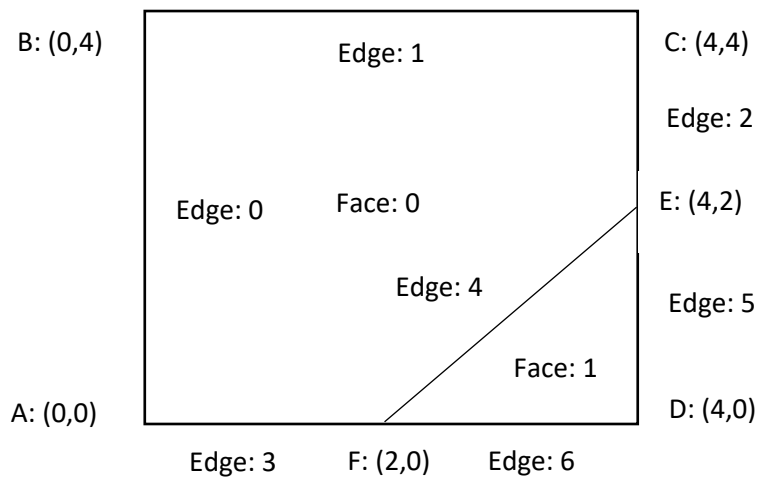
Doubly connected edge list - Wikipedia

Your task is to implement a Doubly Connected Edge List Data Structure and perform certain operation on it. Shape of the polygon will be square for the first assignment. Your program will receive input file name as first argument (Domjudge std input) containing the vertices of Doubly Connected Edge List. Second argument (Domjudge std output) will be the name of output file.
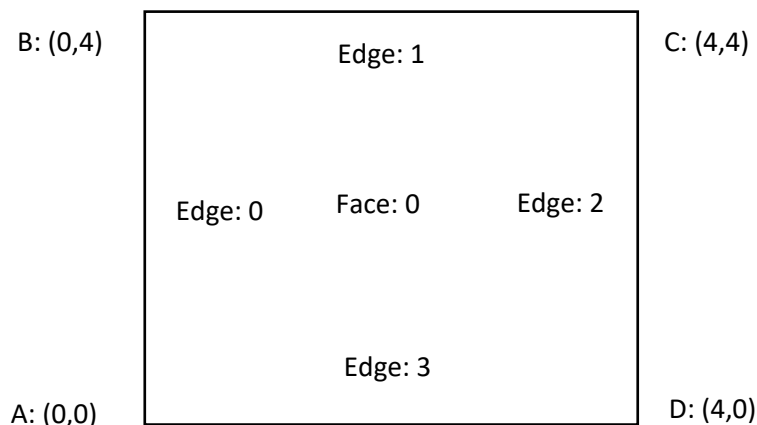
Split Task:

You will receive a list of pairs of integers from file named Split. Each pair represents a split which should

occur in the polygon this split should bisect the two edges, beginning in the middle of the first edge

and finishing at the middle of the second edge.

B: (0,4) Edge: 1 C: (4,4)

Edge: 2

Edge: 0 Face: 0 E: (4,2)

Edge: 4 Edge: 5

Face: 1

A: (0,0) D: (4,0)

Edge: 3 F: (2,0) Edge: 6

Task: Points inside the Face

You will receive an additional file(point_filename) containing the list of the points with its id. Your program should output in increasing order the faces. First print the face number then details of points inside the face.

Input Format:

Input file will contain a list of points, one per line, of the square, each of the coordinates separated by space. First point will be starting point then next point in clockwise direction.

Output Format:

Your program should write to output file. Firstly, print the edge index in clockwise direction. Then half-edges with start vertex index, end vertex index, next edge index, previous edge index and edge index. You also need to output face as mentioned in the output. In the end of the file display point inside the face as mentioned in the task.

Note: Exact output format will be shared by TA's.

Sample Input:

1 2

1 8

7 8

7 2

Split 2 3

Id:1 2 2

Id:2 3 3

Sample Output:

Submission:

You must submit solution on both Domjudge and LMS. Submission on LMS should consist of makefile which should generate final executable name assignment2. Commands to test code on local system:

$make assignment2

$./assignment2 input_filename split_filename point_filename output_filename