In-Class Exercise 2

Given vertex \( v \) in a cell complex of a 2-manifold, the *link* of \( v \) is defined to be the edges that bound the faces that are incident to \( v \), excluding the edges that are incident to \( v \) itself. Present a procedure (in pseudocode) that, given a vertex \( v \) of a DCEL, returns a list \( L \) consisting of the half edges of \( v \)'s link ordered counterclockwise about \( v \). For example, in the figure below, a possible output would be \( \langle e_1, \ldots, e_{11} \rangle \). (Any cyclic permutation would be correct.)

\[ \]

Solution:

The solution provided below is very short, but a bit tricky. We start with any edge \( e \) that is directed out of \( v \). We start following edges around the face lying to \( e \)'s left side, adding each to the link. (In the above figure, this will add \( e_1 \) through \( e_4 \) to the list, and the next edge visited will be directed into \( v \).) When we return to \( v \) (that is, when the destination of the edge is \( v \)) we make a U-turn by setting \( e \) to its twin, and resume from there. (In the figure above, the next edge to be visited will be \( e_5 \).)

```
\text{link(} \text{Vertex } v \text{) \{} \\
\quad L = \text{new empty-list} \\
\quad e = e0 = v.\text{incident; } // \text{any edge coming out of } v \\
\quad \text{do } \{ \\
\quad \quad \quad e = e.\text{next; } // \text{next edge about } e\text{'s left face} \\
\quad \quad \quad \text{if } (e.\text{dest} == v) \text{ } // \text{returning to } v? \\
\quad \quad \quad \quad e = e.\text{twin;} \\
\quad \quad \quad \text{else} \\
\quad \quad \quad \quad \text{add } e \text{ to } L; \text{ } // \text{ } e \text{ is an edge of the link} \\
\quad \quad \text{\} while } (e != e0); \\
\}\n```