MAC0331 - Lista 8

Matheus T. de Laurentys, 9793714

July 12, 2020

Q 4:

```
1  // Using Wikipedia's API:
2  class Edge {
3          Vertex *vert_origin, *vert_destination;
4          Face *face_left, *face_right;
5          Edge *edge_left_cw,
6                  *edge_left_ccw,
7                  *edge_right_cw,
8                  *edge_right_ccw;
9  };
10 class Vertex {
11         float x, y, z;
12         Edge *edge;
13 };
14 class Face {
15         Edge *edge;
16 };
```

a)

```
1  std::vector<Vertex*> RETURN_VERTICES(Face *face) {
2          Edge *edge = face->edge;
3          Vertex *origin = edge->vert_origin;
4          std::vector<Vertex*> vertices = {origin};
5          do {
6                  verices.push_back(edge->vert_destination);
7                  if (*face == *(edge->face_left))
8                          edge = edge->edge_left_cw;
9                  else
10                         edge = edge->edge_right_cw;
11         }
12         while (*(edge->vert_destination) != *origin);
13         return vertices;
14 }
```

b)

```
1  std::vector<Vertex*> RETURN_ADJACENTS(Vertex *v) {
2          std::vector<Vertex*> adj;
3          Edge *edge = v->edge;
4          Edge *initial = v->edge;
5          do {
6                  if (*(edge->vert_origin) == *v) {
7                          adj.push_back(edge->vert_destination);
8                          edge = edge_left_ccw;
9                  }
10                 else {
11                         adj.push_back(edge->vert_origin);
12                         edge = edge_left_cw;
13                 }
14         } while (*initial != *edge);
15         return adj;
16 }
```