The file **prm.py** is a Python script implementing a PR algorithm using as input the ***obstacles.csv*** file, and producing as outputs the ***nodes.csv*** and ***edges.csv*** files.

The **astar3.py** is a Python script implementing the A* search algorithm, using as inputs the above ***nodes.csv*** and ***edges.csv*** files, and producing as output the ***path.csv*** file. This Python script has been developed in the previous assignment. The four ***.csv*** files are needed for the CoppeliaSim visualization.

Phase 1, sampling:

The sampling uses a Bridson algorithm merged and documented at the beginning of the ***prm.py*** script in the function *Bridon_sampling*. The script calls the *Bridon_sampling* function on the whole C-space, then deletes the sampling points in collision with the obstacles. The remaining points are saved in the ***nodes.csv*** file, with the start and goal nodes.

Phase 2, creating edges:

The number of neighbours connected to each node is k=3. After the edges are created, they are tested for collisions with the obstacles by the function *interseg*. The edges passing trough obstacles are deleted and the file ***edges.csv*** is saved.

Phase 3, searching the graph:

This is done by executing the script ***astar3.py***.

Usage:

- Run the script ***prm.py***
- Run the script ***astar3.py***. If the script does not report a SUCCESS, repeat the previous step that will create a new set of nodes and edges and try again
- See the results in CoppeliaSim, using the latest *.csv* files