Project 2 Notes from Info Session

Np.random.rand()

* Will be used within rrt

There are functions that grab the nearest nodes

* Etend?
* N\_node = n+step

Polygonenvironmen has

Flags = \_Trapped, \_Advanced, \_REACHED

RRT connect

* Choose a small enough epsilon

RRT bi-directional

For bi-directional, store one as an “A” tree, and one as a “B” tree

PPRM

Straight line planner is on us to implement.

* We need to plan a path between A and B, and return a flag if the connection isn’t possible.
  + There’s an extra credit problem to build an RRT between A and B
* Take ~100 samples, if all samples are NOT in collision, then we’re good
  + You can return the roadmap, AND the plan
* If you can’t find a plan, add more samples until you can find a plan there.