CODE JOURNAL – PATH PLANNER

**CLASS 1 – Graph Node:**

Member Variables are:

1. Cell Width
2. Cell Height
3. X and z coordinates
4. Boolean Obstacle Variable
5. Boolean Within Lane Variable

Member Functions are:

1. Default Constructor:

* Applies x and z as 0. Cell Width and Cell Height as 10cm.
* Boolean Obstacle is default marked as Non Obstacle
* Initially marked as within lane.

1. Overloaded Constructor:

Inputs: x and z coordinates as double values. Width and height with default 10 cm value

* Performs all the above with the given x,z,w and h params.

1. Get Cell Param:

* Gives details of all the variables of the node to the output screen.

1. Set Obstacle : sets the obstacle Boolean to true.
2. CheckPresence: Inputs(double values x and z)

* Checks and returns true if coordinate is present in the current cells

**CLASS 2 – Grid2D:**

Member Variables are:

1. 2D vector of Graph Nodes

Member Functions are:

1. Default Constructor:

* Constructs a 30 by 20 grid of Nodes with the 300 corresponding to the z axis.

1. Overloaded Constructor:

Inputs: x and z coordinates as double values. Width and height with default 10 cm value

* Performs all the above with the given x,z,w and h params.

Cood system

Z = 3m

X = -1m

0

X = 1m

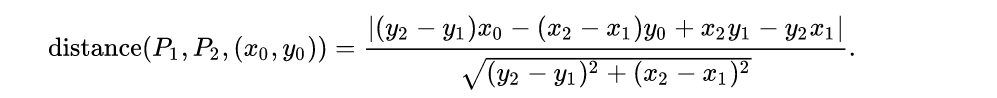
PRM Planner:

The PRM planner class will create a graph with the 2D grid as input. The 2D grid has a list of obstacles with their inflated endpoints. This is done to quicken the collision check algorithm

Collision check:

* The collision check algorithm will take 2 nodes as input.
* It will use the centroid of the 2 points to create a line and find the perpendicular distance to the line from all 4 end points of the obstacle.

Formula for normal distance of a point from a line is



The denominator as well as the modulus on the numerator are irrelevant. All we need to see if there are any sign changes between the distance function values of the 4 end points of the obstacle.