**YouTube Link :** [**https://www.youtube.com/watch?v=cbkws2IFAvo**](https://www.youtube.com/watch?v=cbkws2IFAvo)

**GitHub Repository :** [**https://github.com/N-ADA/Ghost-Buster.git**](https://github.com/N-ADA/Ghost-Buster.git)

Text

Description automatically generated

**Produced by Nada Bounajma**

**&**

**Nour El Houda Touti**

**credit goes to Tarodev - used his algorithm to build the grid**

[**https://www.youtube.com/watch?v=kkAjpQAM-jE**](https://www.youtube.com/watch?v=kkAjpQAM-jE)

In the Game Grid there is one Ghost hiding and we don't know where the ghost is. Therefore, to find it, we used measurements which are probabilities. Each box/tile is assigned a probability, and once clicked, we get a color back (red, yellow, green, or orange)

The further away we get from the ghost, the probability of Red goes down, orange goes up, yellow goes higher than orange, and Green gets the highest probability :

P(Red) < P(Orange) < P(Yellow) < P(Green)

In the game, we keep clicking on boxes and once we think we hit the ghost(meaning once the box turn RED) we press the Button that is seen on top of the Grid (check the Youtube video). We either receive a 'Success' Message, meaning we busted the ghost, or 'Fail' Message, meaning we missed the ghost.

Green color ==> means we are at least 5 blocks away from the ghost

Red color ==> means we are more likely on the ghost, but we still don't know that for sure, it could be just a noisy measurement. That is why it's better to keep clicking on the boxes around it to get assured then proceed to click on the Button on top of the Grid to bust it(check the YouTube video)

**We first create our Game Grid and give it a size of 8x20. We randomly place the ghost as the following code shows**Text

Description automatically generated with low confidence

**And distribute the equal probabilities on all nodes**

Text

Description automatically generated with low confidence

**The following code runs on each click == > it changes the probability of the tile that we clicked so the tile reflects and matches with the corresponding color that it shows. After that, we normalize the rest of the tiles to represent a logical probability distribution.**

**Text

Description automatically generated**

**The following code distribute probabilities on the tiles based on their distance from the ghost**

**Text

Description automatically generated**