**Readme:**

**Libs required:**

1. pygame: Library for game development and multimedia applications in Python.
2. random: Built-in Python module for generating random numbers.
3. numpy (np): Library for efficient numerical computations and array manipulation.
4. math: Built-in Python module providing mathematical functions.
5. torch: Main library in PyTorch for building and training machine learning models.
6. torch.nn: PyTorch submodule for constructing neural networks.
7. matplotlib.pyplot: Module for data visualization and plotting.
8. cv2: Python module for OpenCV, used for image and video processing tasks.

**Q-Learning:**

* For running the map with Q-learning run **python3 Q-learning.py**
* This creates a pygame display where the agent learns and tests with 25 defined goal points.Initial Learning takes some time.
* Finally a plot is created to display the steps taken to reach the goal.

**Deep-Q Learning:**

* For training the model run **python3 dqn\_train.py**
* This creates a trained model file **dqn\_model\_train.pth**
* The trained model path is added in dqn\_test.py. Run **python3 dqn\_test.py**.
* This runs the trained model for pre-defined 25 goal points.
* Graph is plotted for steps taken and total rewards .
* Finally, the pygame is saved as **output.mp4**

**Video Link:** <https://drive.google.com/file/d/1A_CswYDB5gAEbK1yeVwowI6KVEVEddWb/view?usp=sharing>

**Trained Model Link:**

<https://drive.google.com/file/d/1PPmrTZ3vYeRYKdiyrTn7_RYFjnZeFxsN/view?usp=sharing>