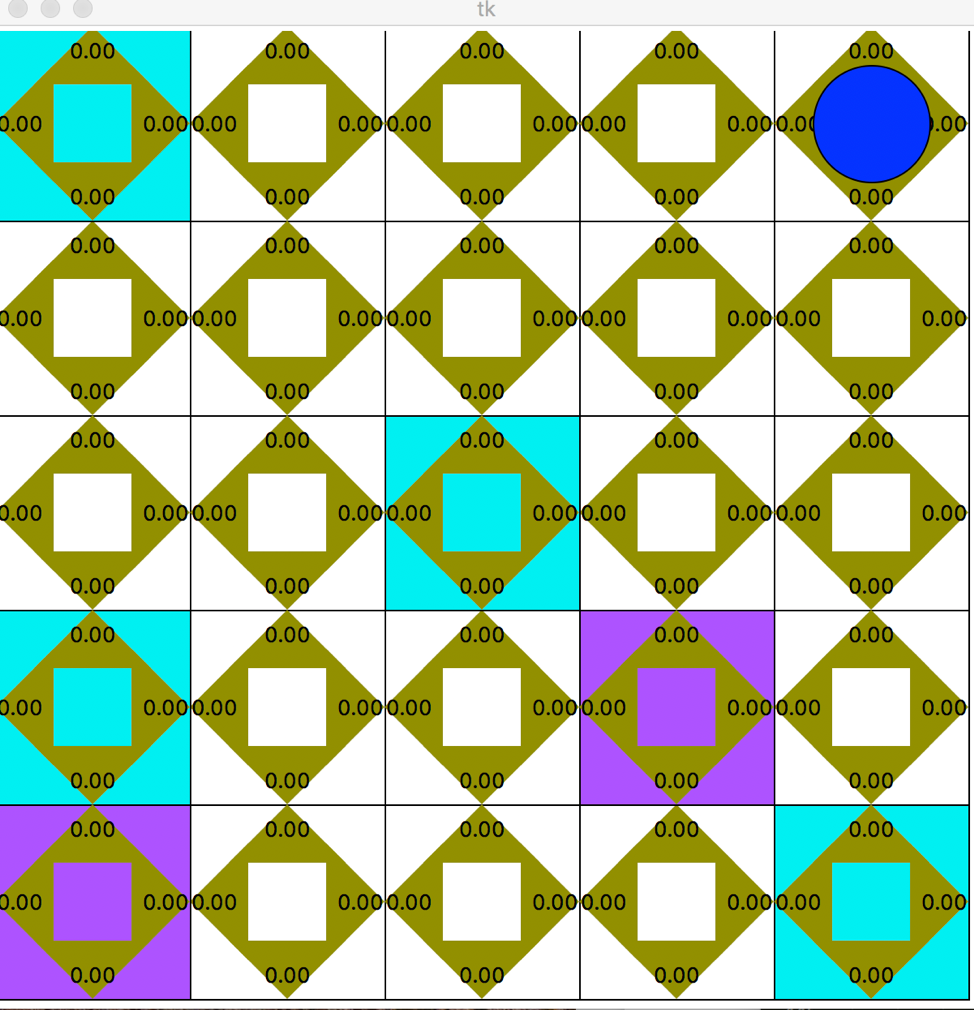
Reinforcement Learning

Using Q Learning and SARSA learning, we analyze an agent’s performance in a grid world. The given game does not have a goal state. It learns iteratively until the maximum number of steps is reached

VISUAL WORLD:



IDEALOGY:

\* An agent is referred to as a moving robot. The agent moves along the 5x5 grid.

\* An 5x5 grid is initialized with all the Q values as 0.

\* The agent can carry only one block at a time

\* There are 4 pickup states(blue) and the violet indicate the drop states

\* Each pickup state can contain maximum 4 blocks and drop state can contain maximum 8 blocks

\* Initially, the pickup is maximized to 4 while the drop-off is initialized to 0

\* When the agent reaches a pickup block and the agent does not have a block, it will pick.

\* When the agent reaches a drop state and the agent does have a block, it will drop the block.

\* When the agent picks, its maximum value is reduced while when agent drops, its value is incremented.

USAGE:

1. Please use python2.7 with tkinter installed.

2. Terminal commands

python Project.py("Change the experimentnumber in main() in project.py")

3. The program saves the Q\_values in a csv file

4. The number of operators performance measure is printed to terminal. SInce there can be more than 1 time a terminal can be reached in an experiment, total number of operators required to reach a terminal state is calculated everytime terminal is reached. ALL these are stored in num\_operators array ;printed to the terminal