Alirza Guliyev 20026

Assignment 2

Variables all white cells in the nXm matrix represented by 'Y'. Black cells are represented by 'X'. Numbers are represented by numbers themselves. Domain for each cell is {0,1}. 0 Meaning that there was no bulb in a white cell and 1 meaning that there is a bulb in that cell. Constraints are defined after rows and columns are divided into sectors which edges are restricted by numbers or 'X's. After that MaxSumConstraint(1) is applied to every sector in the matrix because each can have at most one bulb in it. Another constraint is applied to the cells which contain a number in it which does the following: If a number in the cell is n then apply MaxSumConstraint (n) to horizontally and vertically adjacent variables. The last constraint is applied for every white cell checking which sector this cell belongs to row and column wise and applying MinSumConstraint(1) to those cells because every white must be illuminated at least once.

CSP is more appropriate for this problem because it needs less computation considering the solutions will be in the depth of the graph because it works similar to Depth First Search and favors depth over breadth whereas A* favors breadth over depth.

Test1:

graph = [['Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'X'],   # 7X7 hw2

['Y', 'Y', '4', 'Y', 'Y', 'Y', 'Y'],

['0', 'Y', 'Y', 'Y', '1', 'X', 'Y'],

['Y', 'Y', 'Y', '1', 'Y', 'Y', 'Y'],

['Y', 'X', '1', 'Y', 'Y', 'Y', 'X'],

['Y', 'Y', 'Y', 'Y', 'X', 'Y', 'Y'],

['1', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y']

]

Test2:

graph = [['Y', 'Y', 'Y', 'Y', 'X', 'Y', 'Y'], # 7X7 Hard

['Y', '2', 'Y', 'Y', 'Y', '0', 'Y'],

['1', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y'],

['Y', 'Y', 'Y', '2', 'Y', 'Y', 'Y'],

['Y', 'Y', 'Y', 'Y', 'Y', 'Y', '2'],

['Y', '2', 'Y', 'Y', 'Y', '3', 'Y'],

['Y', 'Y', 'X', 'Y', 'Y', 'Y', 'Y']

]

Test3:

```
graph = [ ['Y', 'Y', '1', 'Y', '2', 'Y', 'Y'],  # 7X7 Normal

    ['Y', 'Y', 'Y', '2', 'Y', 'Y', 'Y'],

    ['2', 'Y', 'Y', 'Y', 'Y', 'Y', 'X'],

    ['Y', 'X', 'Y', 'Y', 'Y', 'X', 'Y'],

    ['1', 'Y', 'Y', 'Y', 'Y', 'Y', '3'],

    ['Y', 'Y', 'Y', 'X', 'Y', 'Y', 'Y'],

    ['Y', 'Y', '2', 'Y', 'X', 'Y', 'Y']

    ]
```

Test4:

```
graph = [ ['Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y','Y','Y','Y'],  # 10X10 Hard

    ['Y', 'X', '0', 'Y', 'Y', '1', 'Y','Y','1','Y'],

    ['Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y','Y','1','Y'],

    ['Y', 'Y', 'Y', 'X', '1', 'Y', '0','Y','Y','Y'],

    ['Y', '4', 'Y', 'Y', 'Y', 'Y', '0','Y','Y','Y'],

    ['Y', 'Y', 'Y', 'X', 'Y', 'Y', 'Y','Y','X','Y'],

    ['Y', 'Y', 'Y', 'X', 'Y', 'X', '0','Y','Y','Y'],

    ['Y', 'X', 'Y', 'Y', 'Y', 'Y', 'Y','Y','Y','Y'],

    ['Y', '0', 'Y', 'Y', '1', 'Y', 'Y','2','0','Y'],

    ['Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y','Y','Y','Y']

    ]
```