Presentation – Progress II

Introduction to Artificial Intelligence

Student ID	Full Name	Email	Assigned tasks	Complete percentage
521H0272	Nguyen Gia My	521H0272@student.tdut.edu.vn	Research, Implement	100%
521H0201	Truong Gia Bao	521H0201@student.tdut.edu.vn	Research	100%
521H0324	Nguyen Van Truong	521H0324@student.tdut.edu.vn	Research	100%
521H0492	Nguyen Duc Duy Bao	521H0492@student.tdut.edu.vn	Research	100%

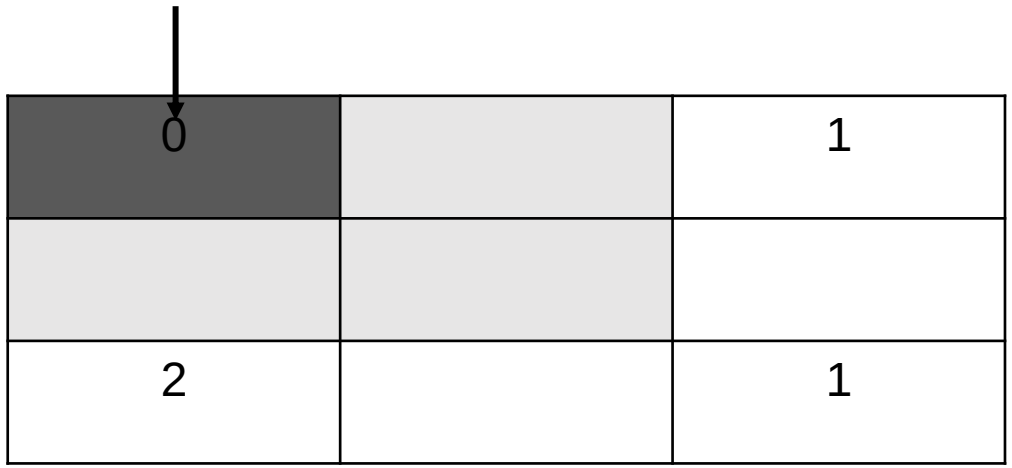
File structure

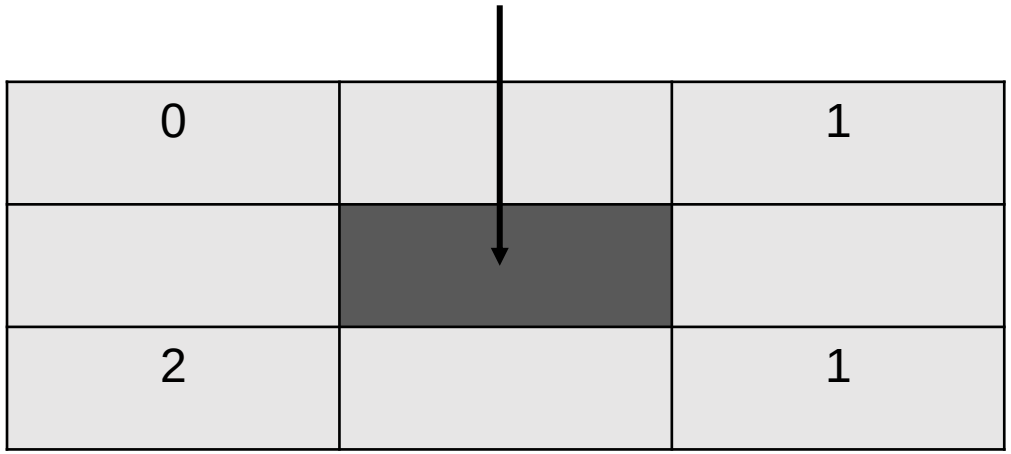
3	3	
0		1
2		1

3	3	

0		1
2		1

Define Adjacents Cells



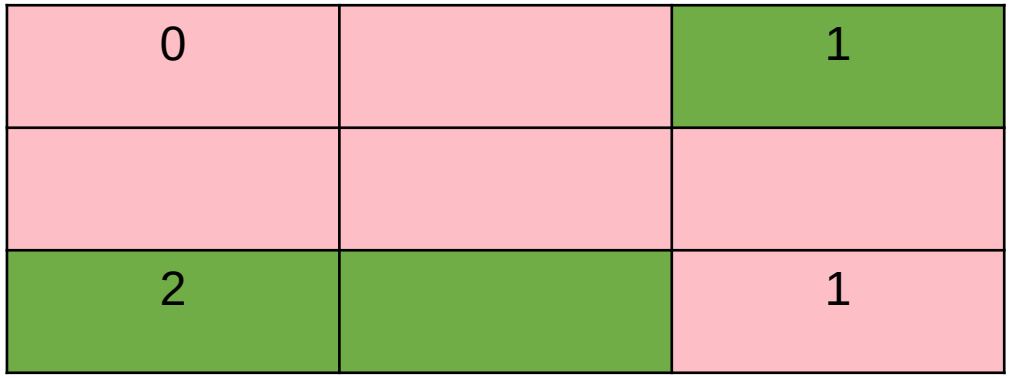


	0	1	2
0	0		1
1			
2	2		1

(0,0)	(0,1)	(0,2)
(1,0)	(1,1)	(1,2)
(2,0)	(2,1)	(2,2)

1	2	3
4	5	6
7	8	9

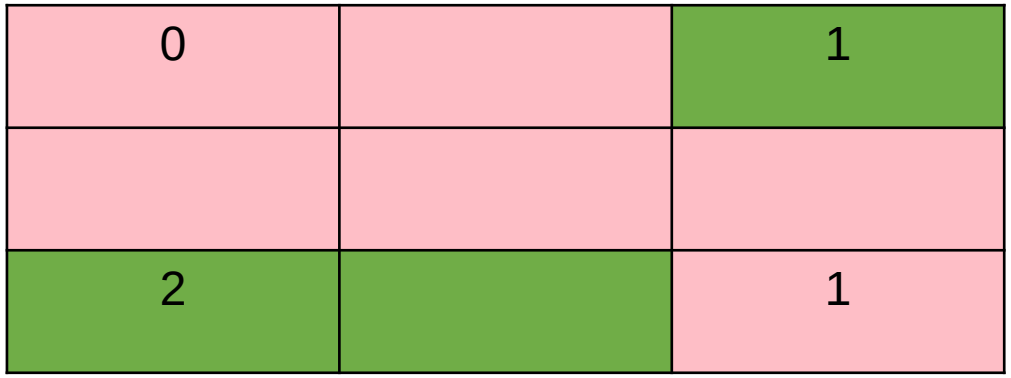
Approach



“A cell must be green or red”

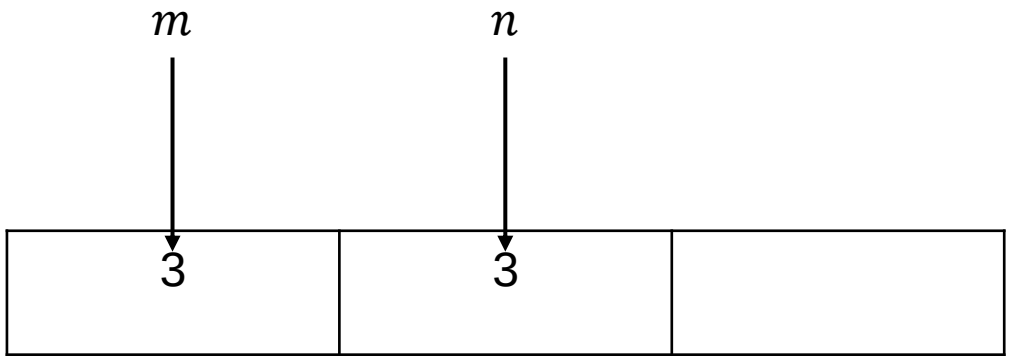
Green ∨ ¬Green

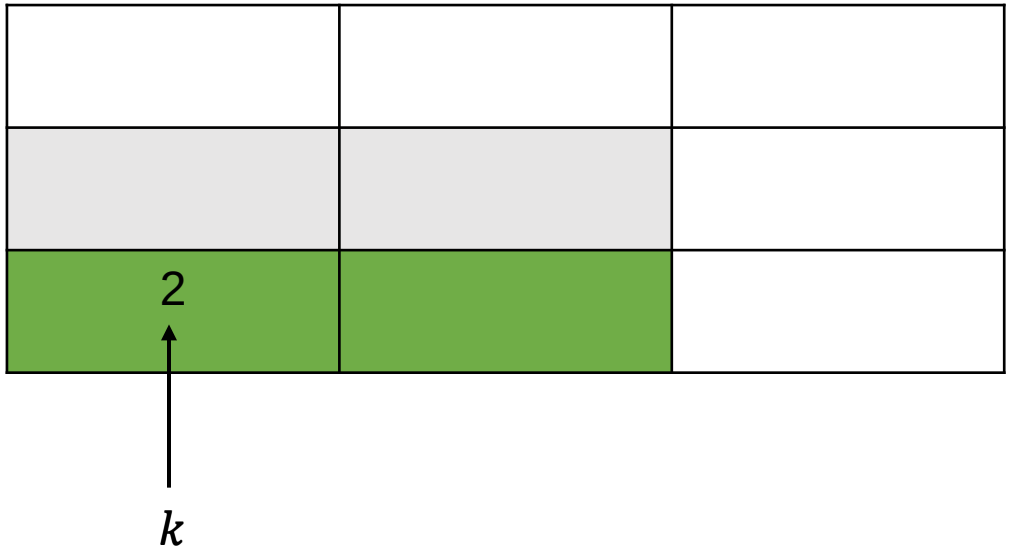
“A cell must be green or red”



To assign a clause symbol to each cell:

- Traverse each cell of the matrix
- In each cell:
  - Creates a list of 2 values: *i* (green) or *-i* (red), where | i | is the coordinate of the matrix converted to integer form

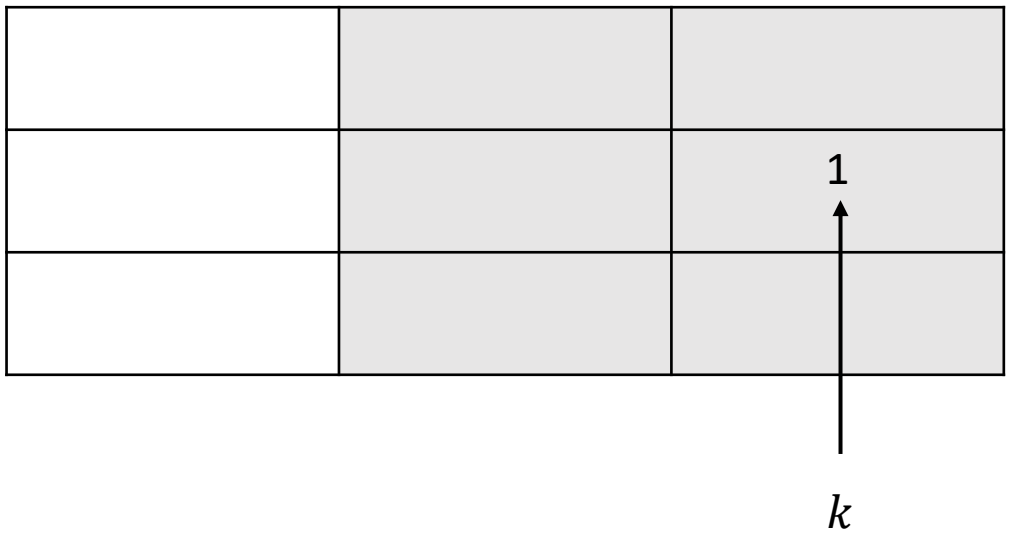




$$\sum_{i=1}^{m\times n} G_i = k$$

G_i: *a green cell at position* i

adjacents: 6

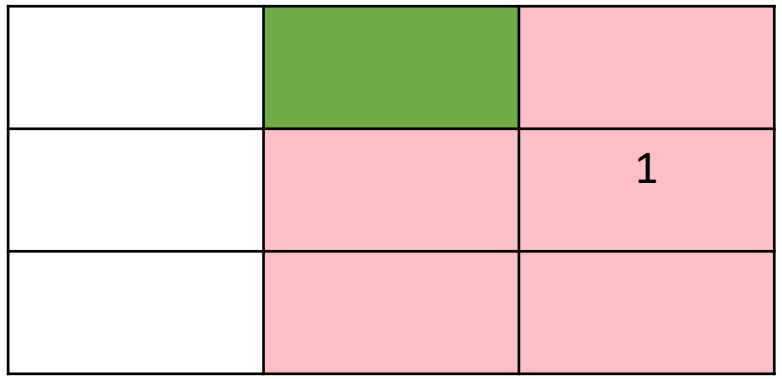


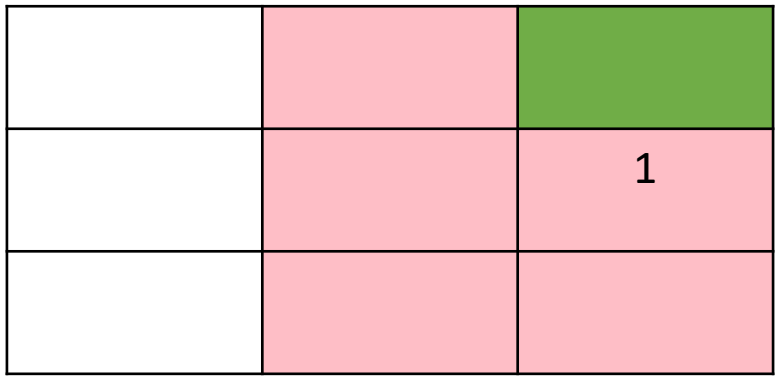
Ways to color

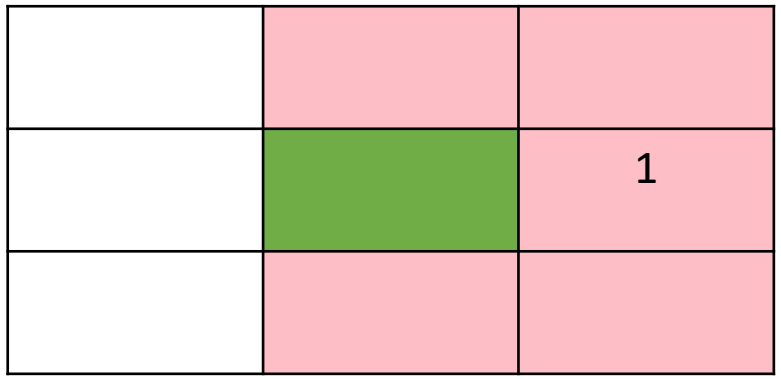
$$C^{k}_{adjacents}$$

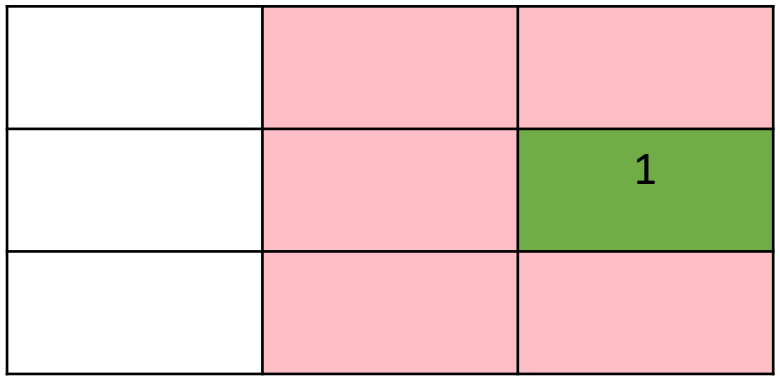
adjacents: 6

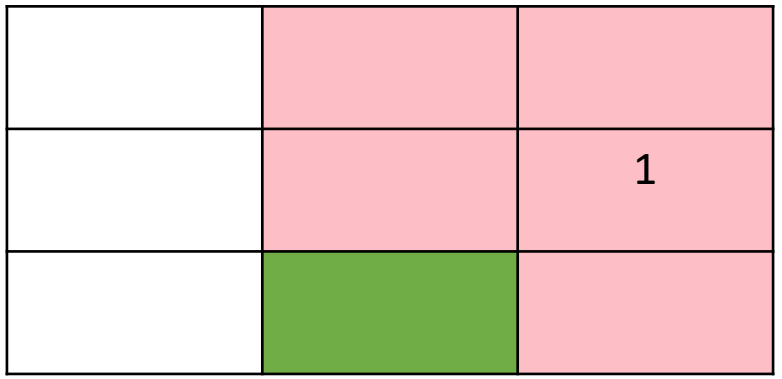
Ways to color: $C^{k}_{adjacents} = C^{1}_{6} = 6$

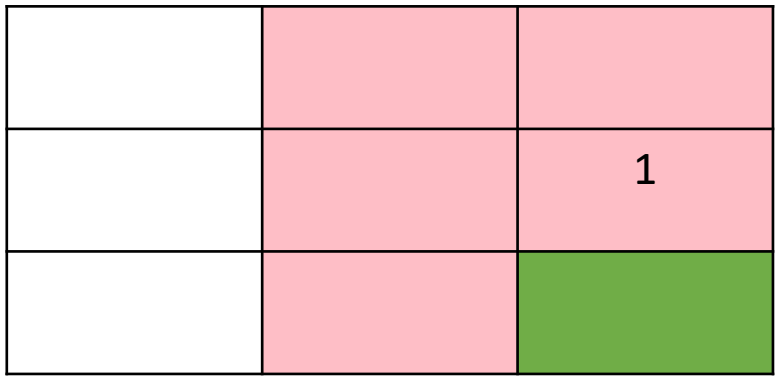




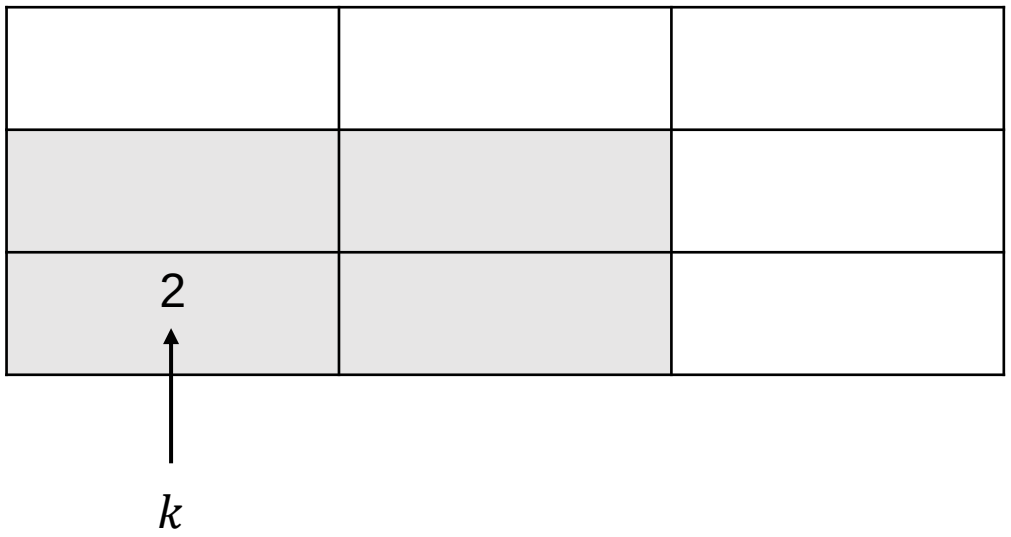






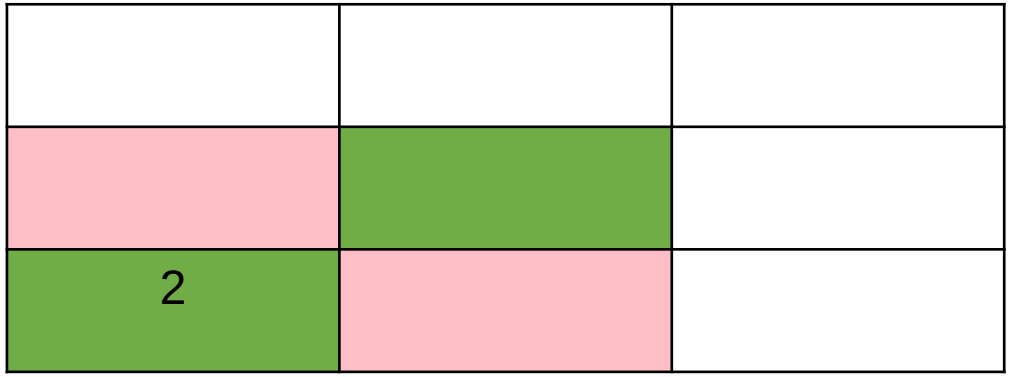


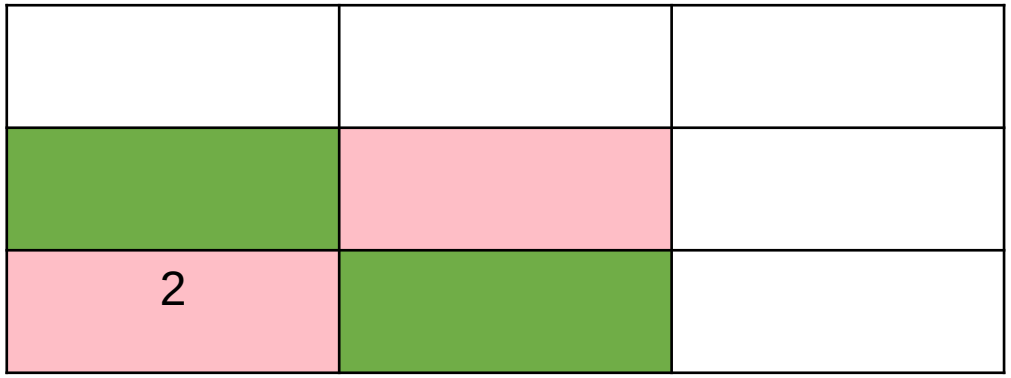
adjacents: 4

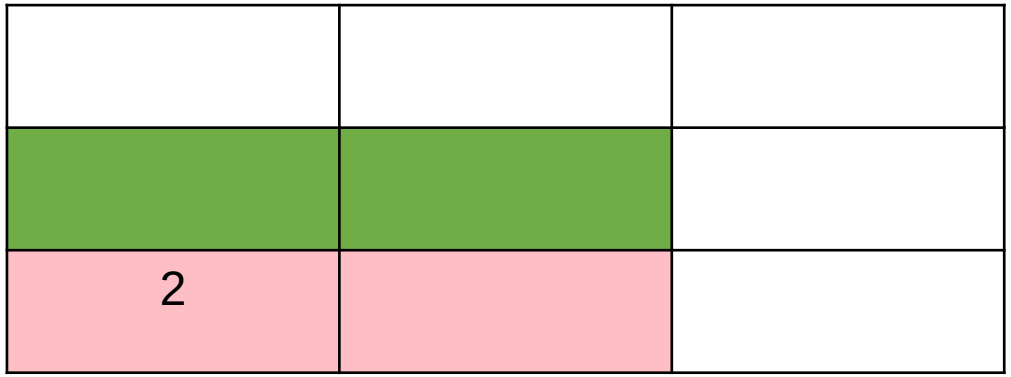


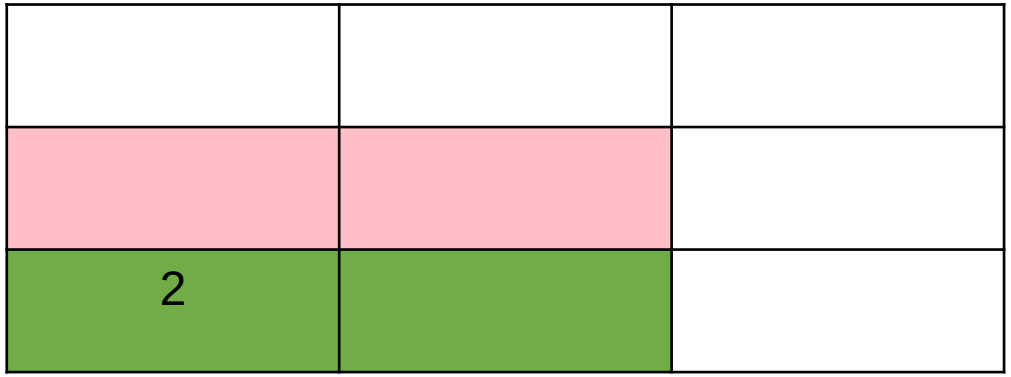
Ways to color

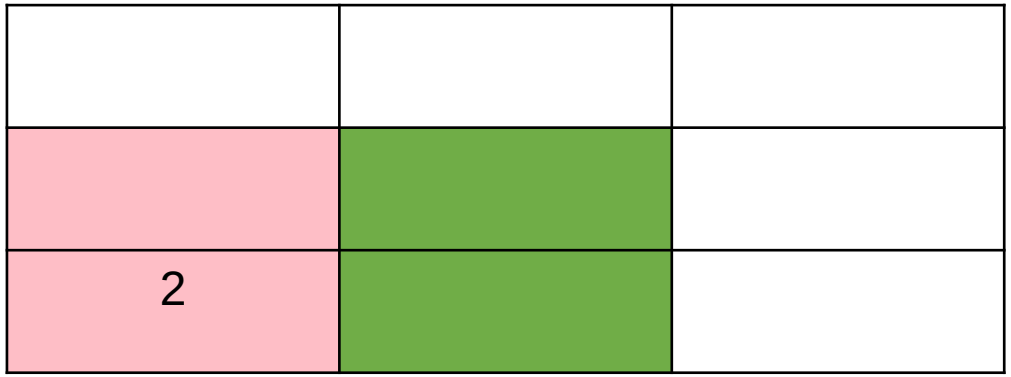
$$C^{k}_{adjacents}$$

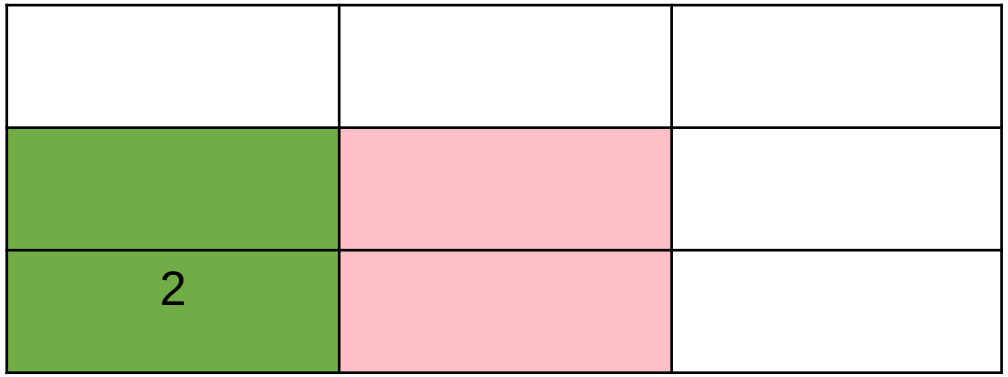

2

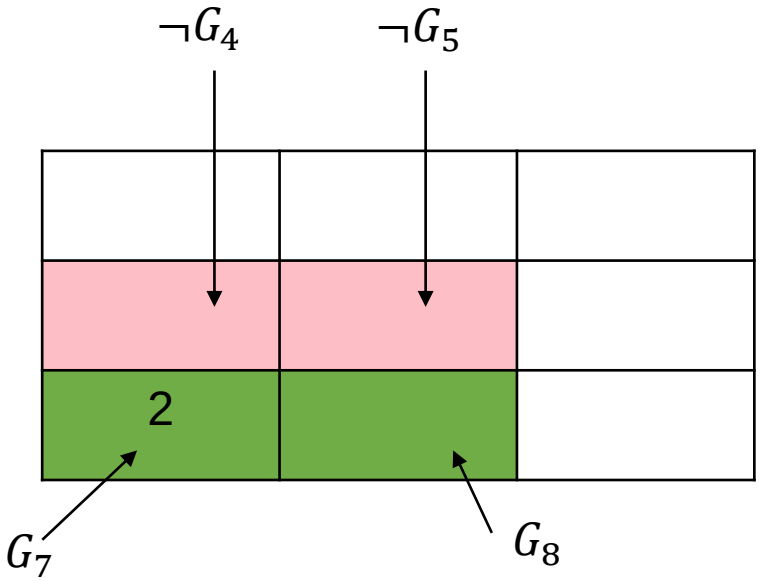

2


2


2


2


2



1	2	3
4	5	6
7	8	9

$$\sum_{i=1}^{m \times n} G_i = k$$

$$(\boldsymbol{G_7} \wedge \boldsymbol{G_8}) \Leftrightarrow (\neg \boldsymbol{G_4} \wedge \neg \boldsymbol{G_5})$$

$$\equiv [(G_7 \wedge G_8) \rightarrow (\neg G_4 \wedge \neg G_5)] \wedge [(\neg G_4 \wedge \neg G_5) \rightarrow (G_7 \wedge G_8)]$$ */Eliminate biconditional*

$$\equiv [\neg(G_7 \wedge G_8) \vee (\neg G_4 \wedge \neg G_5)] \wedge [\neg(\neg G_4 \wedge \neg G_5) \vee (G_7 \wedge G_8)]$$ */Eliminate implication*

$$\equiv [(\neg G_7 \vee \neg G_8) \vee (\neg G_4 \wedge \neg G_5)] \wedge [(G_4 \vee G_5) \vee (G_7 \wedge G_8)]$$ */De Morgan's Law*

$$\equiv [(\neg G_7 \vee \neg G_8 \vee \neg G_4) \wedge (\neg G_7 \vee \neg G_8 \vee \neg G_5)] \wedge [(G_4 \vee G_5 \vee G_7) \wedge (G_4 \vee G_5 \vee G_8)]$$

/Distributive Law

$$[\ (\neg G_7 \vee \neg G_8 \vee \neg G_4) \wedge (\neg G_7 \vee \neg G_8 \vee \neg G_5)] \wedge [(G_4 \vee G_5 \vee G_7) \wedge (G_4 \vee G_5 \vee G_8)]$$

$$\downarrow$$

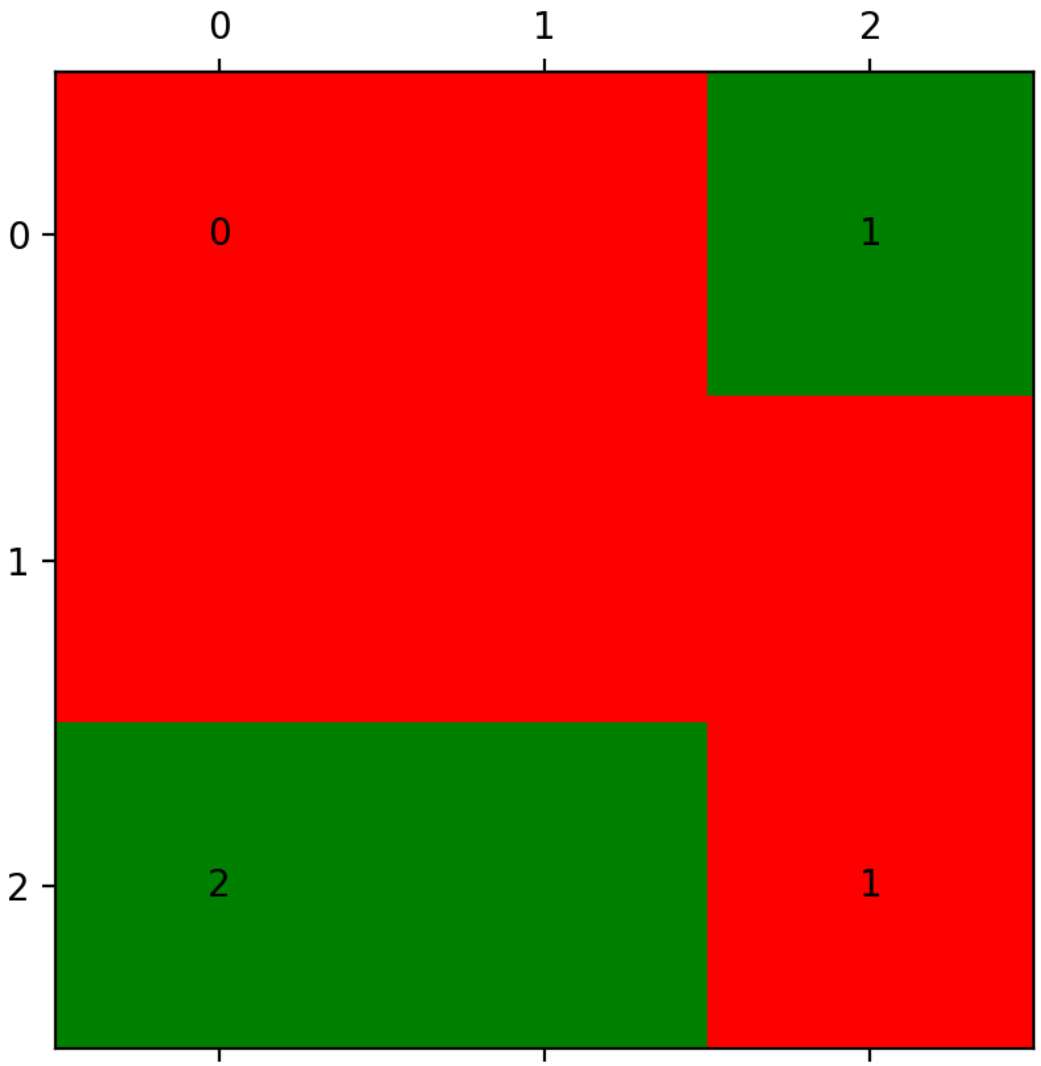
$$(\neg G_7 \vee \neg G_8 \vee \neg G_4) \wedge (\neg G_7 \vee \neg G_8 \vee \neg G_5) \quad \wedge \quad (G_4 \vee G_5 \vee G_7) \ \wedge \ (G_4 \vee G_5 \vee G_8)$$

```
Solution:  [-1, -2, 3, -4, -5, -6, 7, 8, -9]
```

-1	-2	3
-4	-5	-6
7	8	-9

0		1
2		1

```
Solution:  [-1, -2, 3, -4, -5, -6, 7, 8, -9]
```



Advantages	Disadvantages
There are many documents on propositional logic	Still having logic errors when implementing into programming
Libraries are available to solve the problem	It takes a long time to read and understand the functions of Pysat
There are descriptions and suggestions from the topic	It takes a long time to understand the logic

Table of complete percentages for each task

Task	Complete Percentages
Read file	100%
Get matrix data (rows, columns, cell coordinates, cell values)	100%
Find adjacent cells of a cell	100%
Find model (solution)	100%
Draw matrix	100%

References

Books

[1] Stuart Russell, Peter Norvig, [2005], Artificial Intelligence: A Modern Approach, 3rd edition, pp. 224-249.

Notes

[2] Harvard CS50's Introduction to Artificial Intelligence with Python, Lecture Note 1.