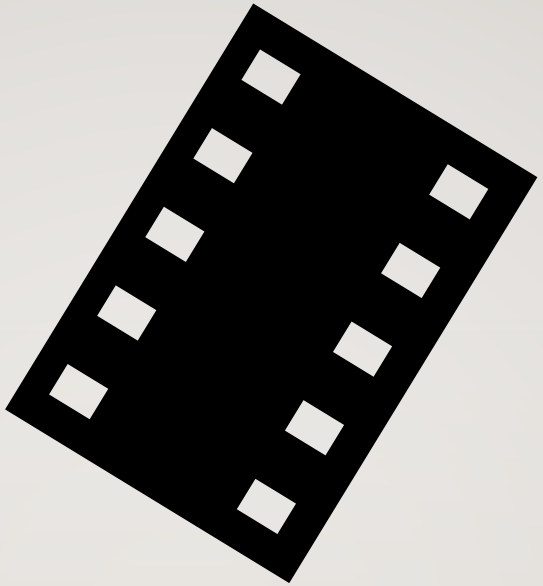
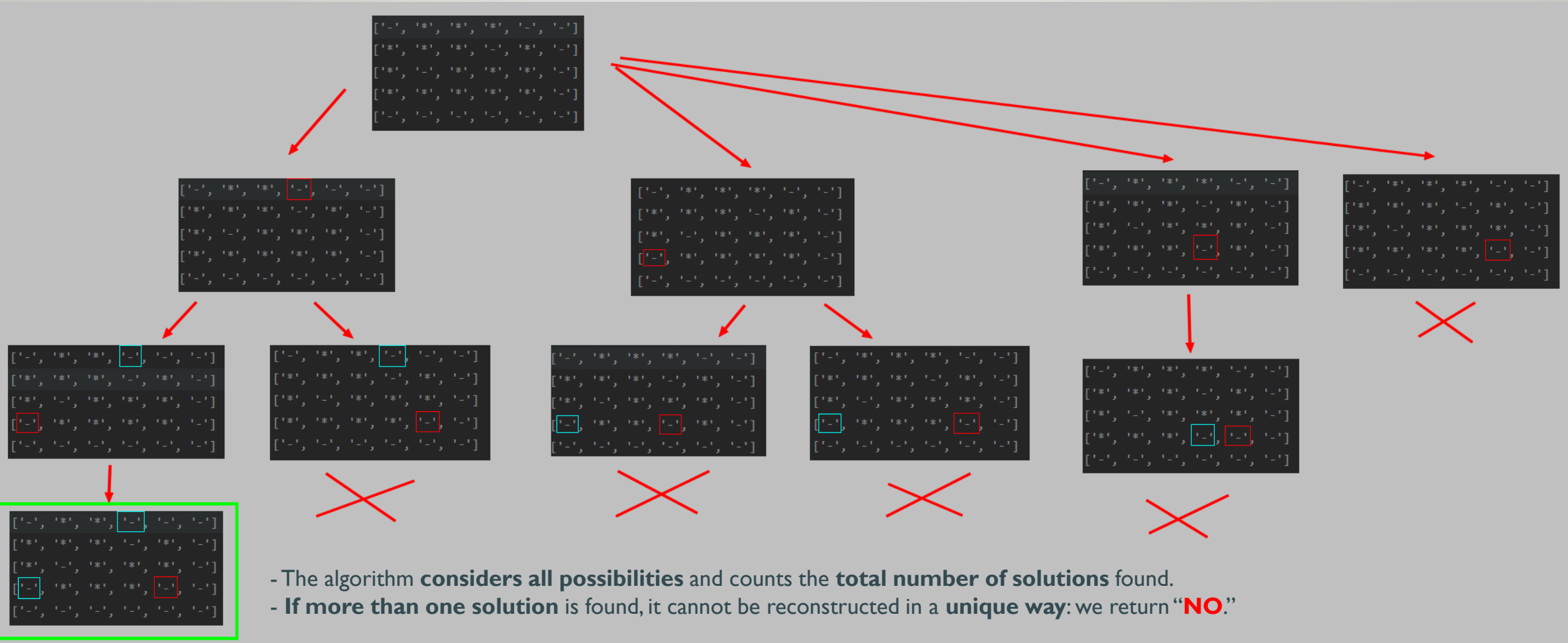
08/01/2026



DAMAGED TICKET

Yann Martin D'Escrienne

OVERALL FUNCTIONING OF THE BASE ALGORITHM

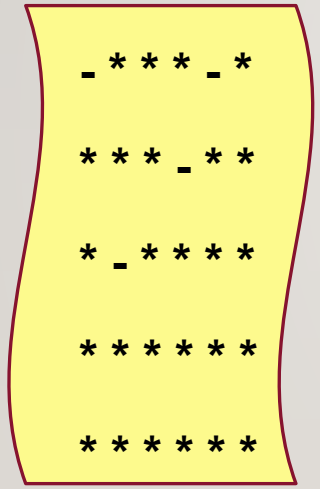


- The algorithm **considers all possibilities** and counts the **total number of solutions** found.
- **If more than one solution** is found, it cannot be reconstructed in a **unique way**: we return "**NO**."

EXAMPLE #1

- Execution of the algorithm with the input:

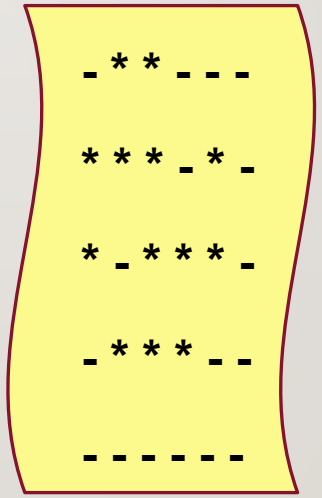
5 6

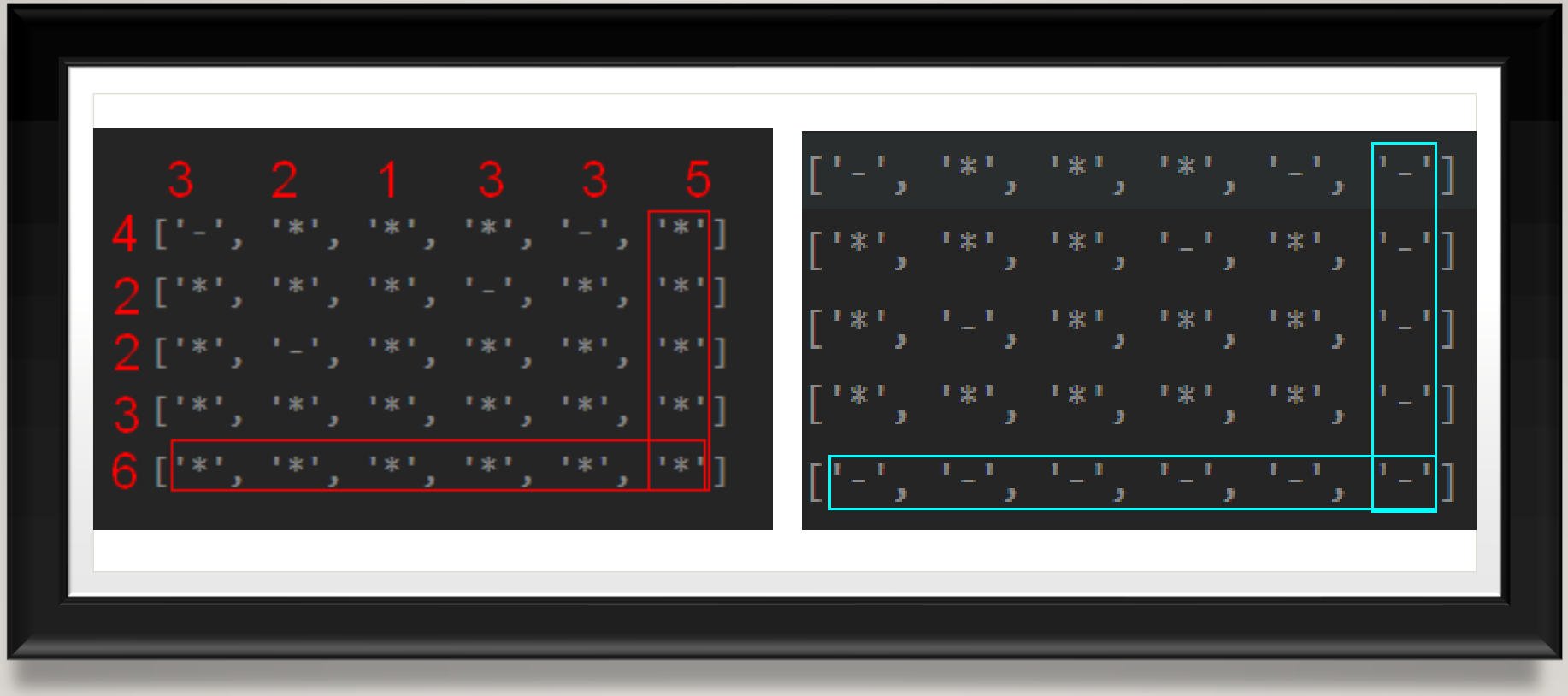


4 2 2 3 6

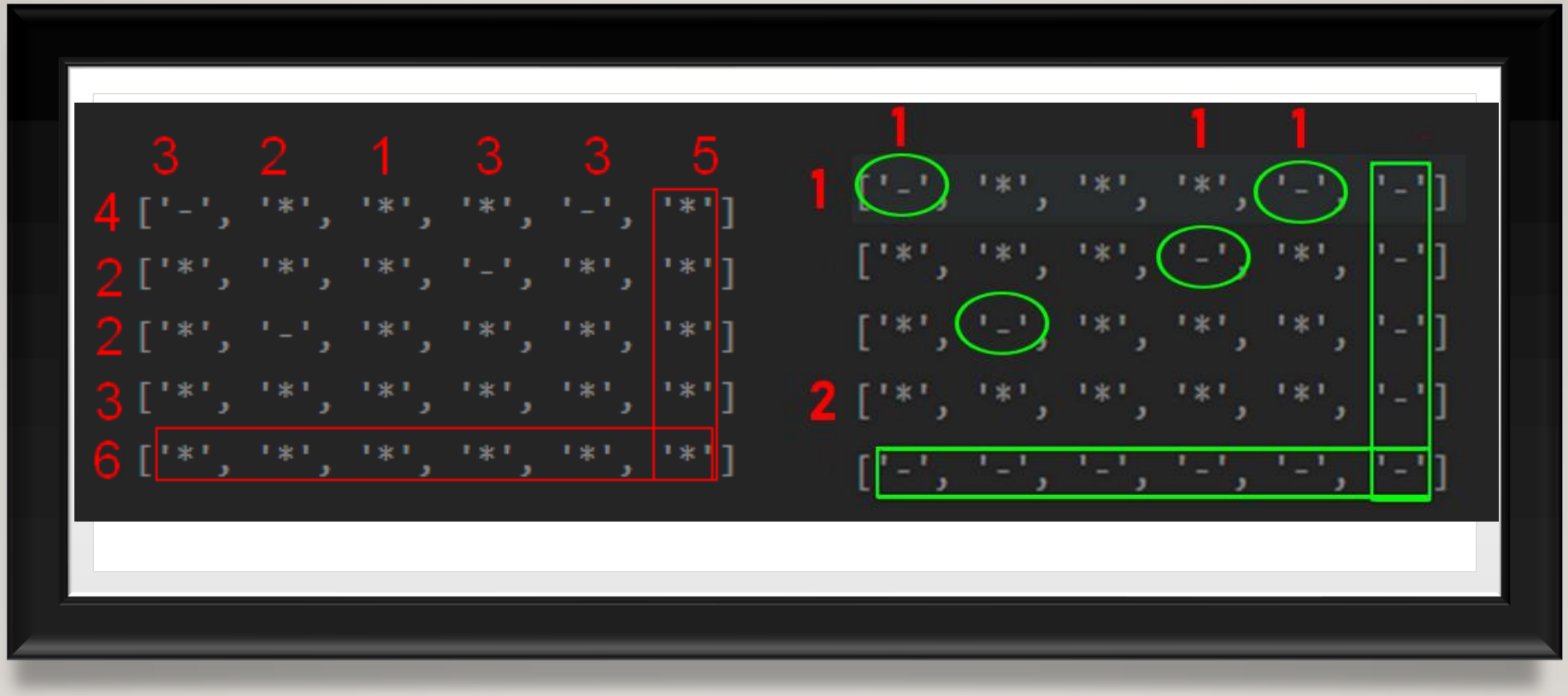
3 2 1 3 3 5

- Which should result in:

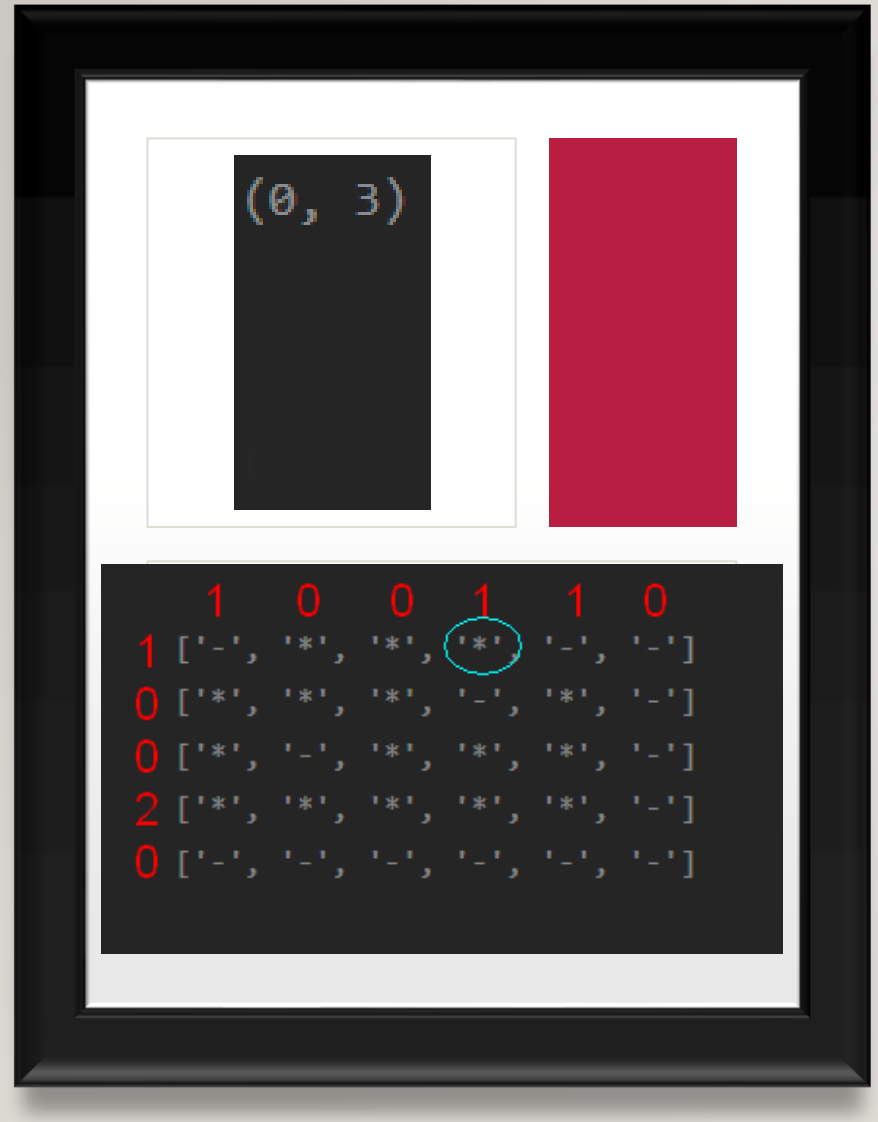




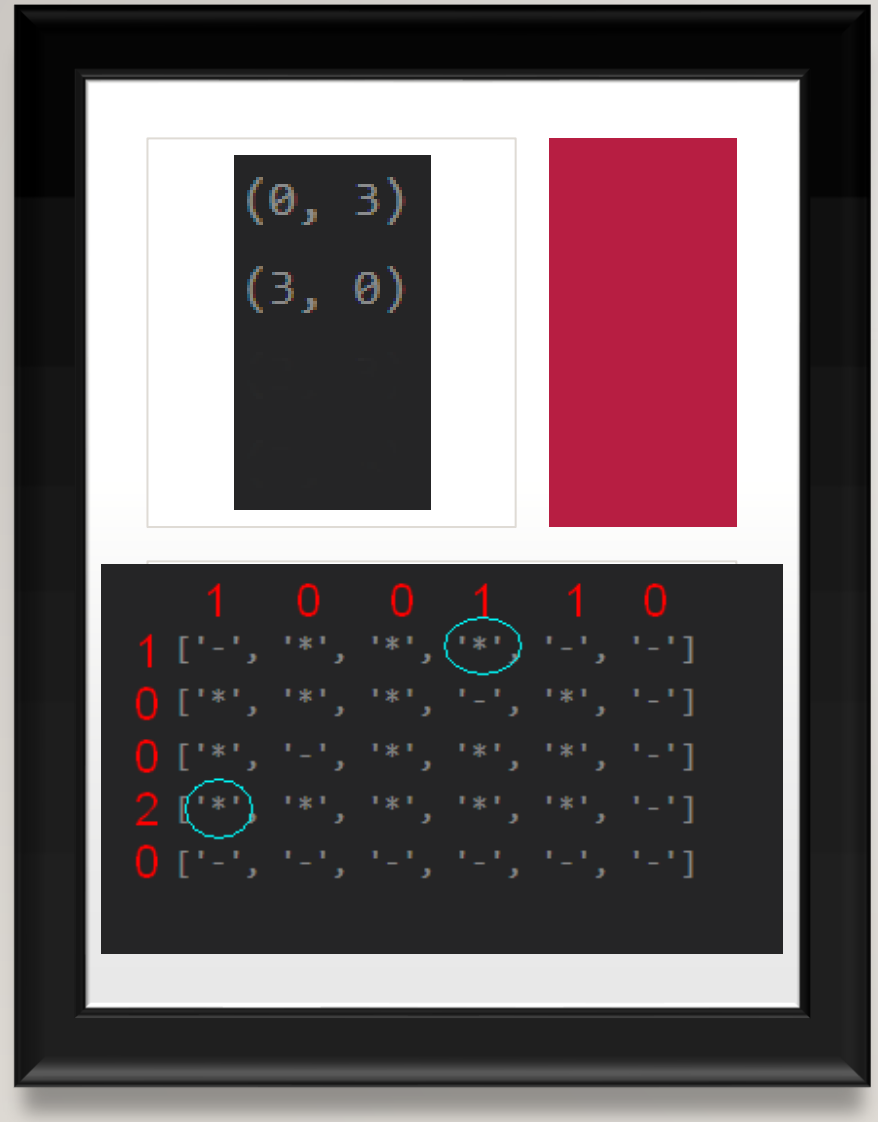
TREATMENT OF TRIVIAL CASES



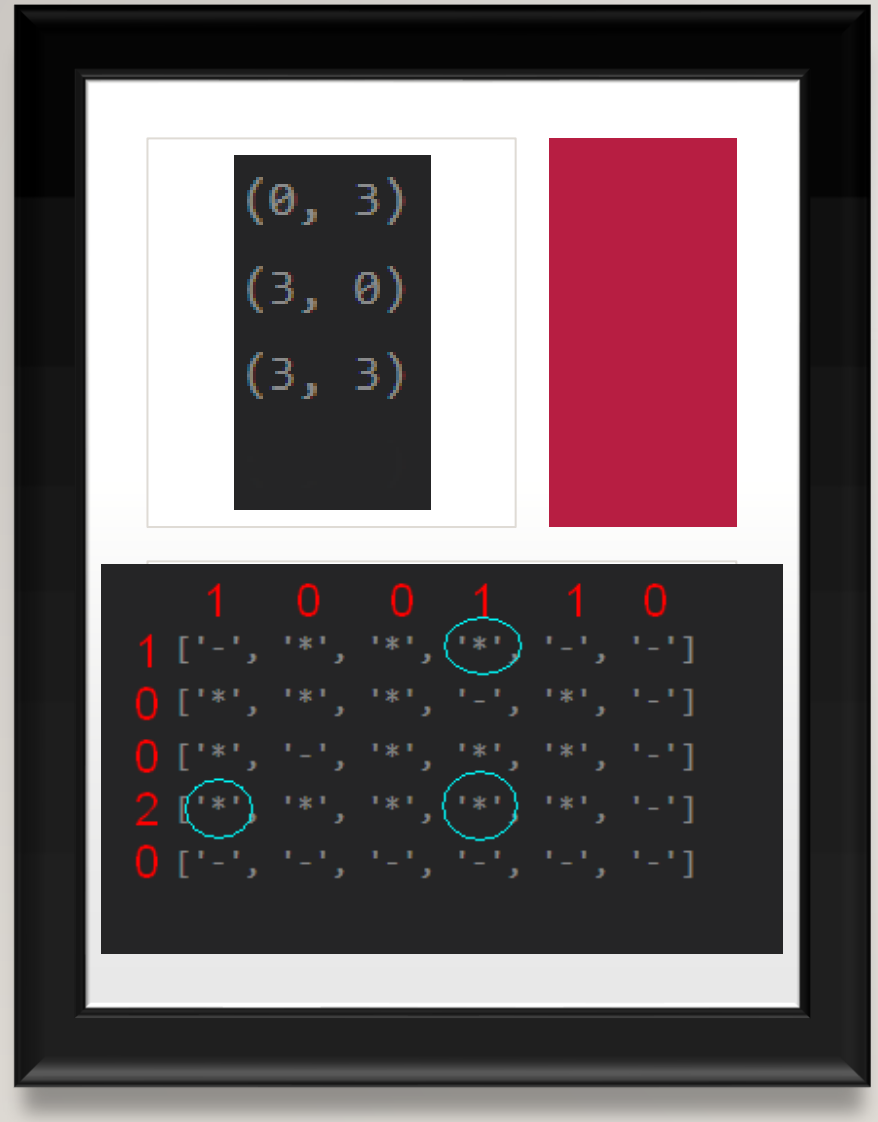
REDUCE COUNTER OF "-" WITH EXISTING



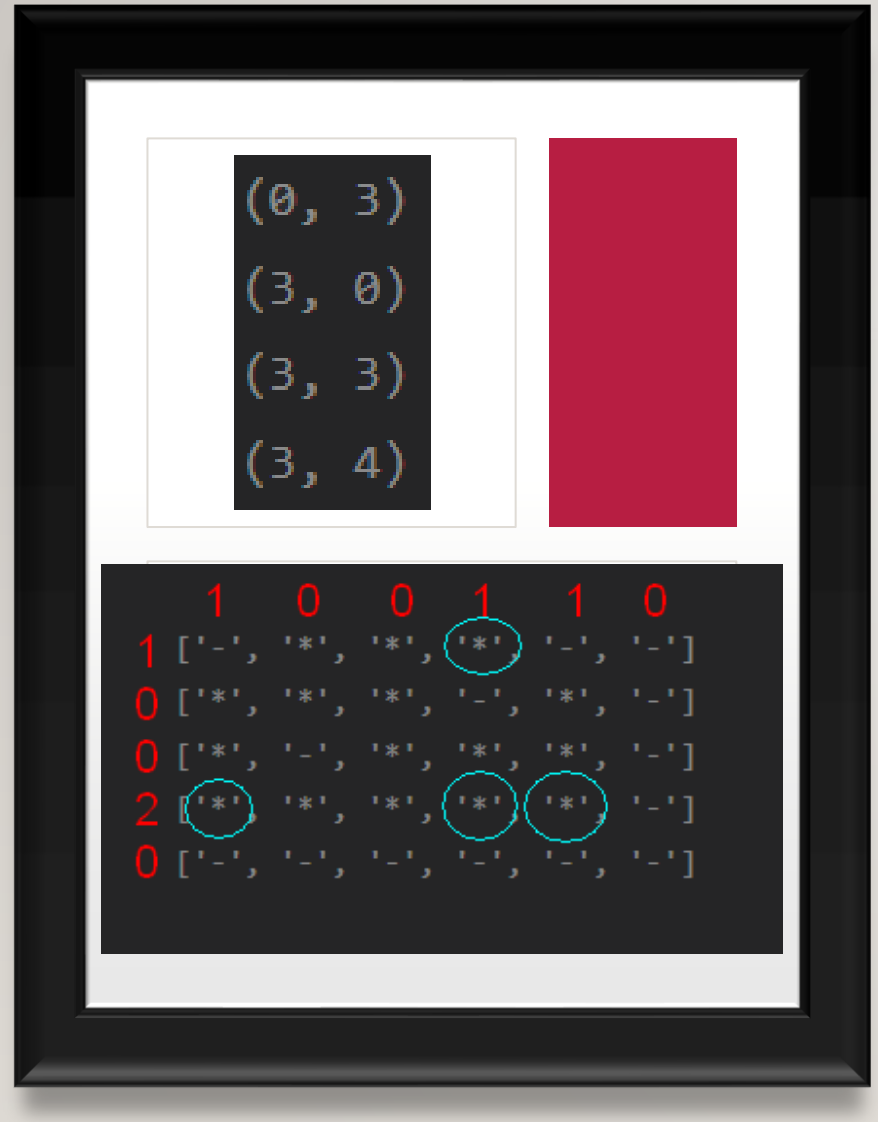
LIST OF POSSIBILITIES



LIST OF POSSIBILITIES



LISTE DES POSSIBILITES

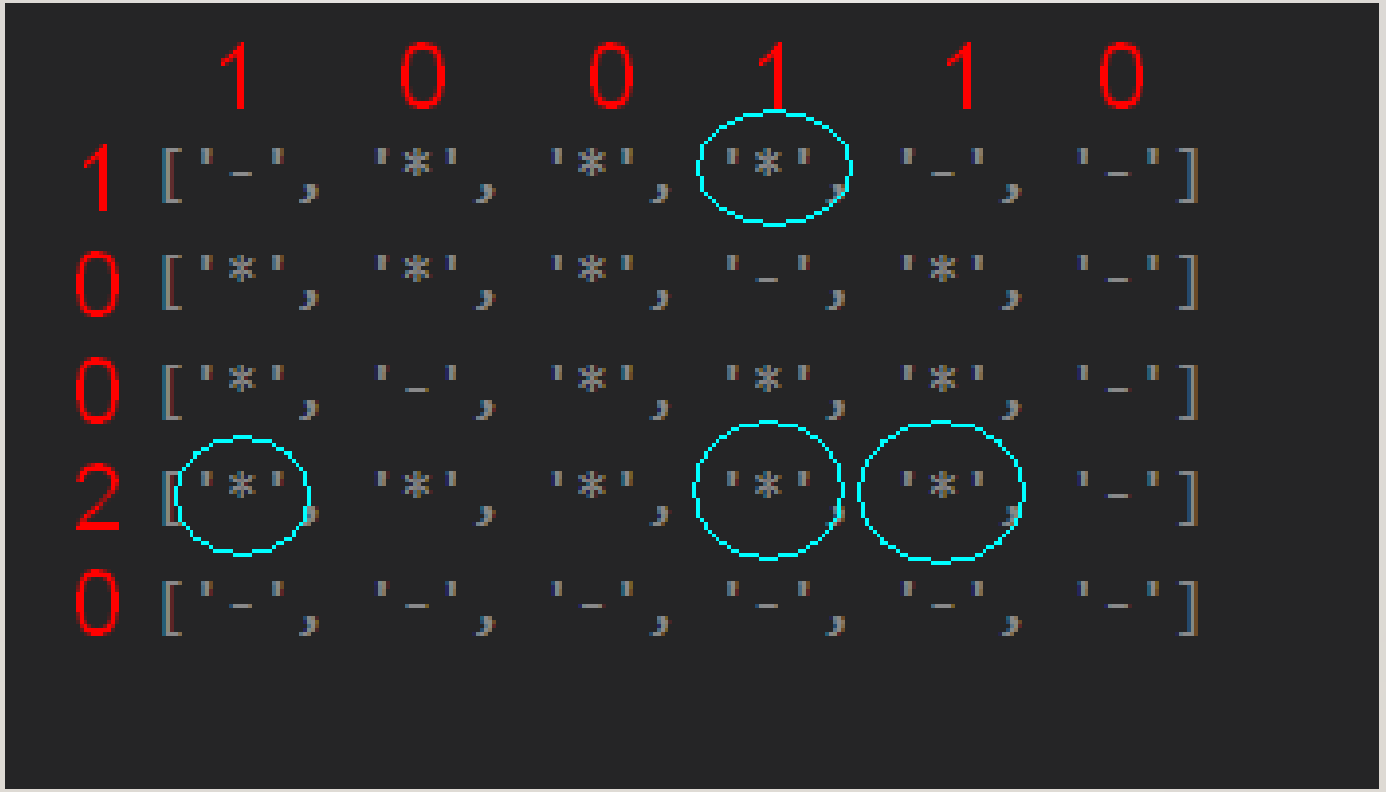


LIST OF POSSIBILITIES

PATH OF POSSIBILITIES:

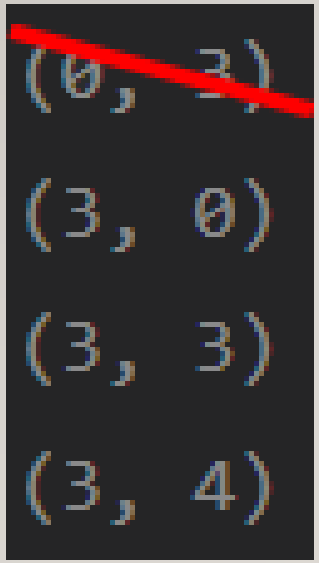
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

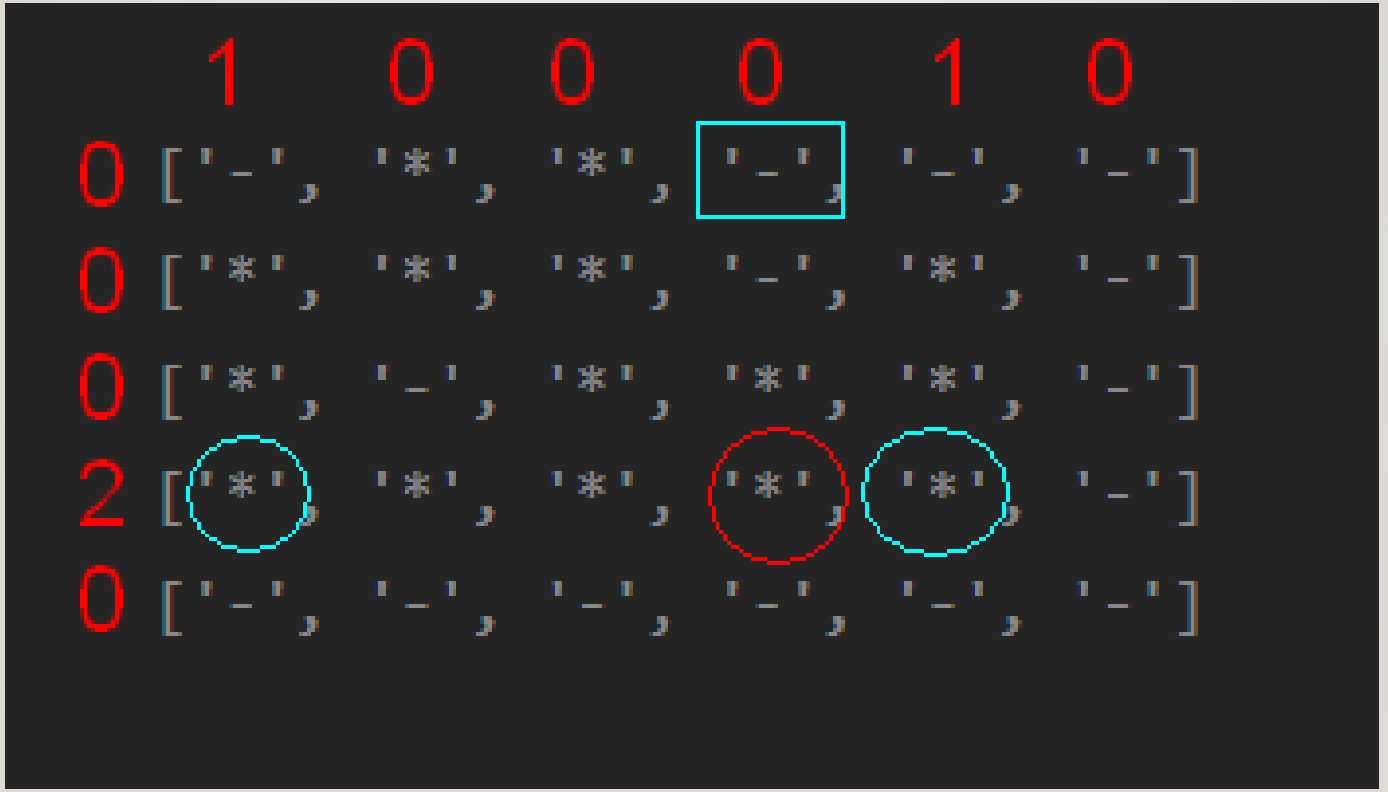
```
(0, 3)
(3, 0)
(3, 3)
(3, 4)
```



PATH OF POSSIBILITIES:

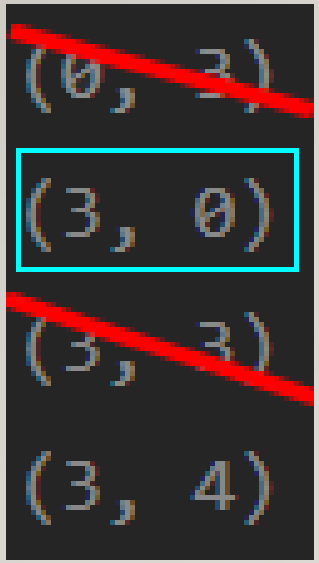
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

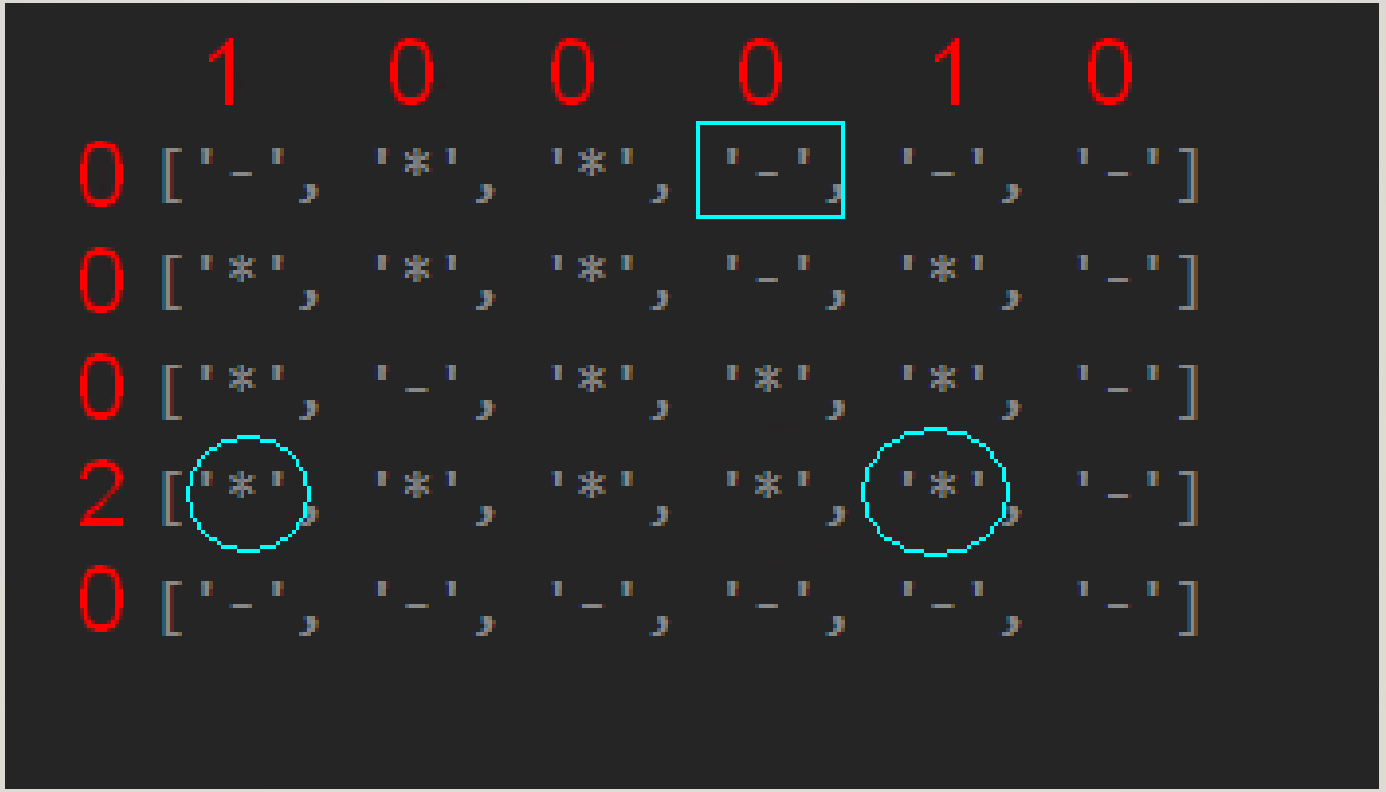




PATH OF POSSIBILITIES:

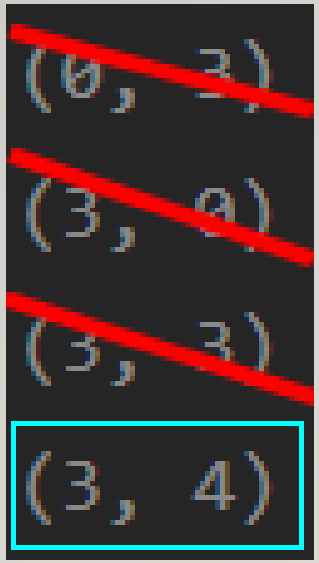
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

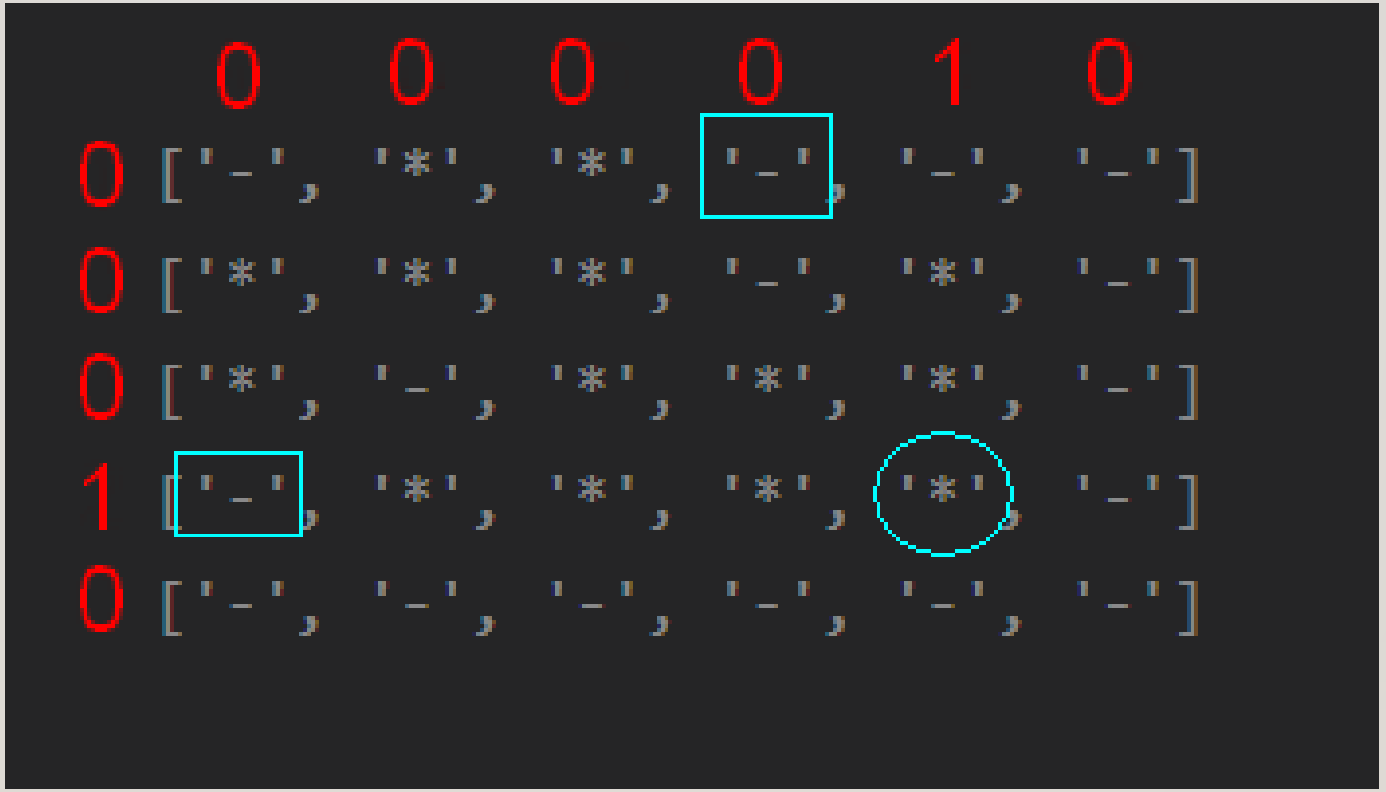




PATH OF POSSIBILITIES:

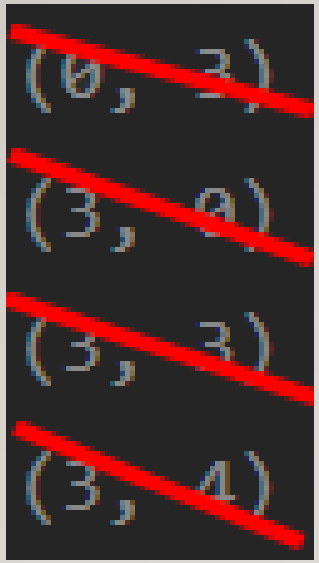
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

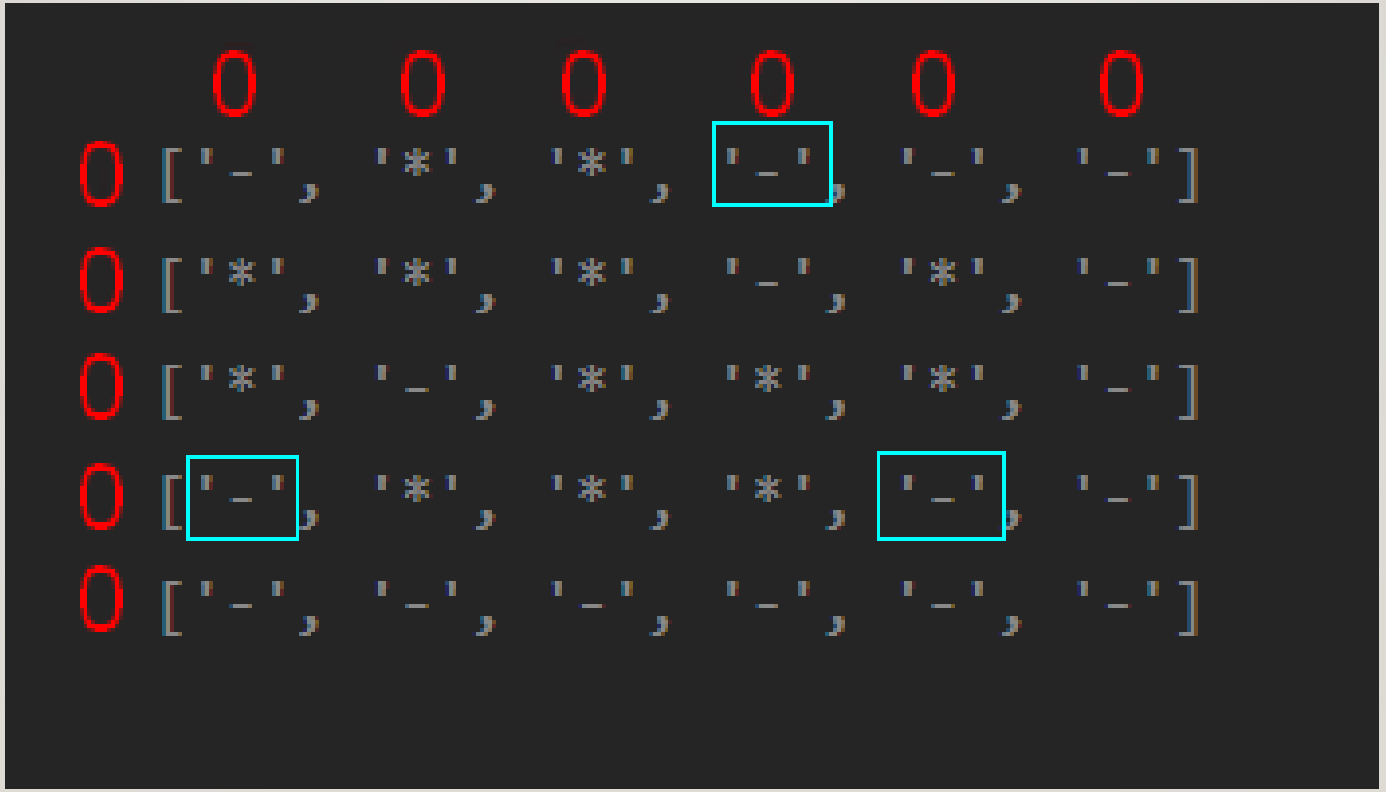




PATH OF POSSIBILITIES:

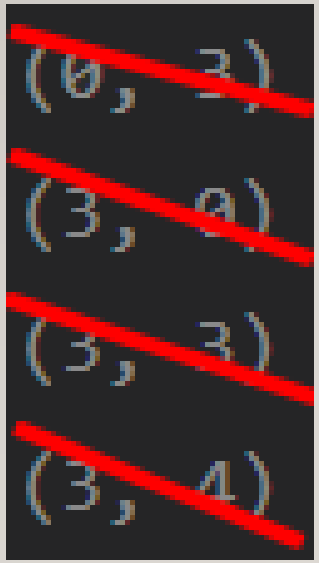
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

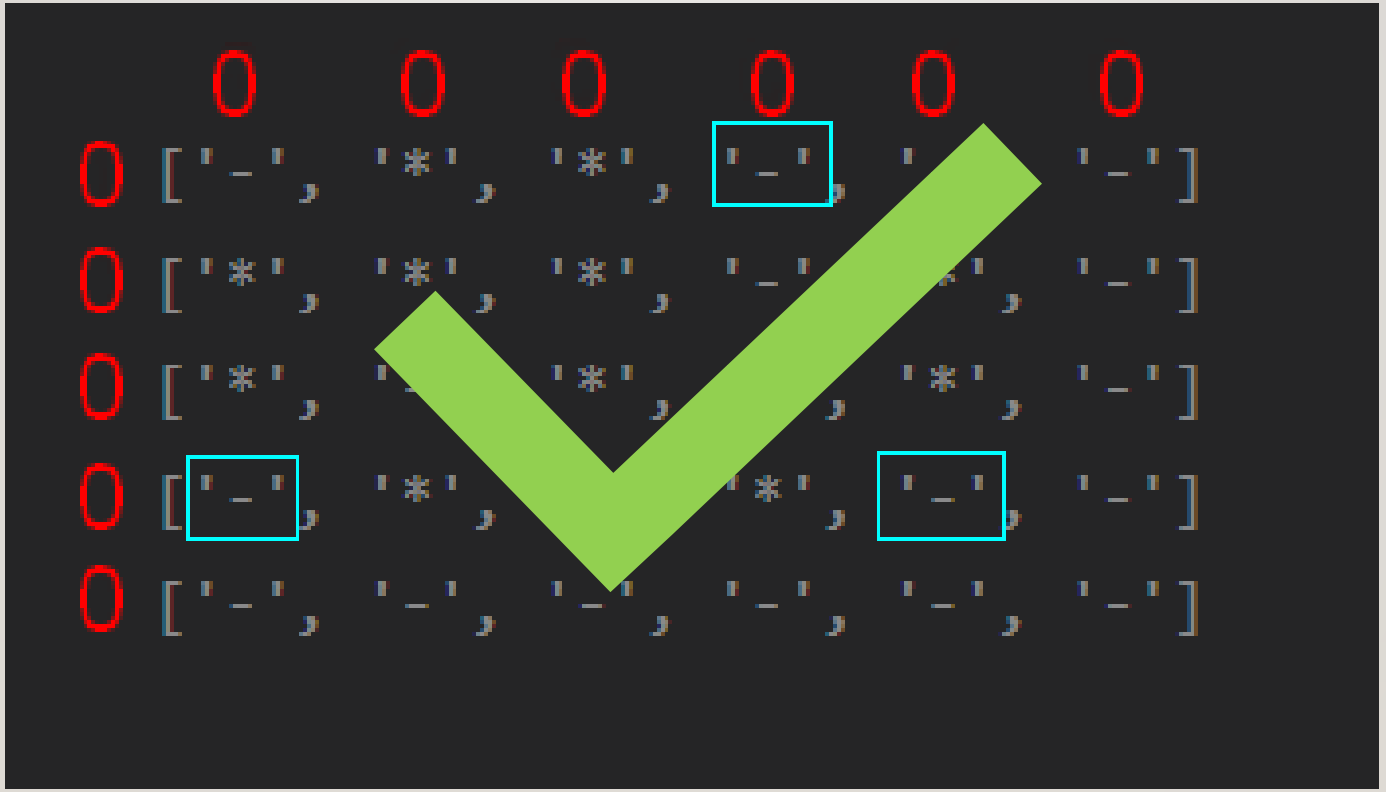


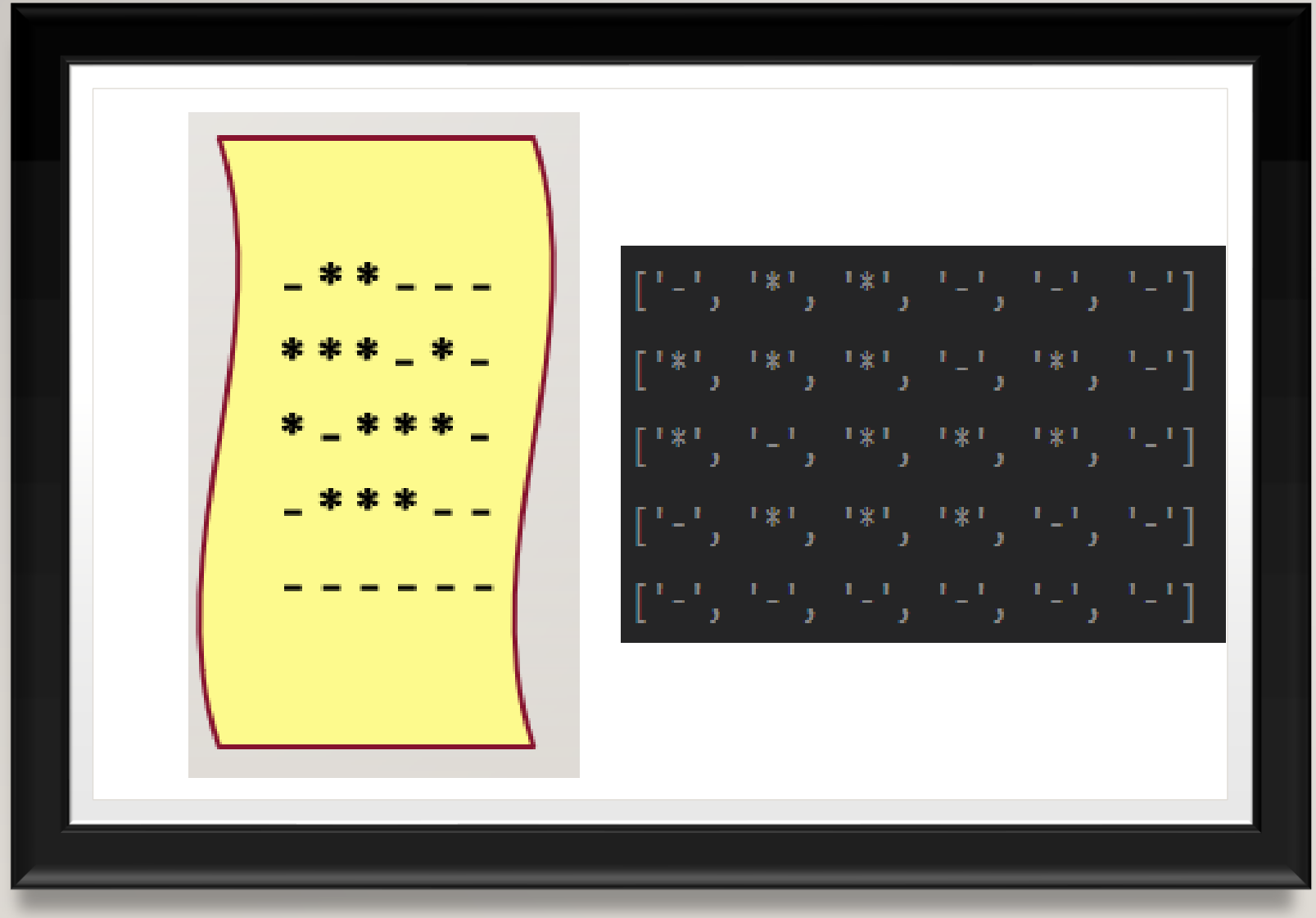


PATH OF POSSIBILITIES:

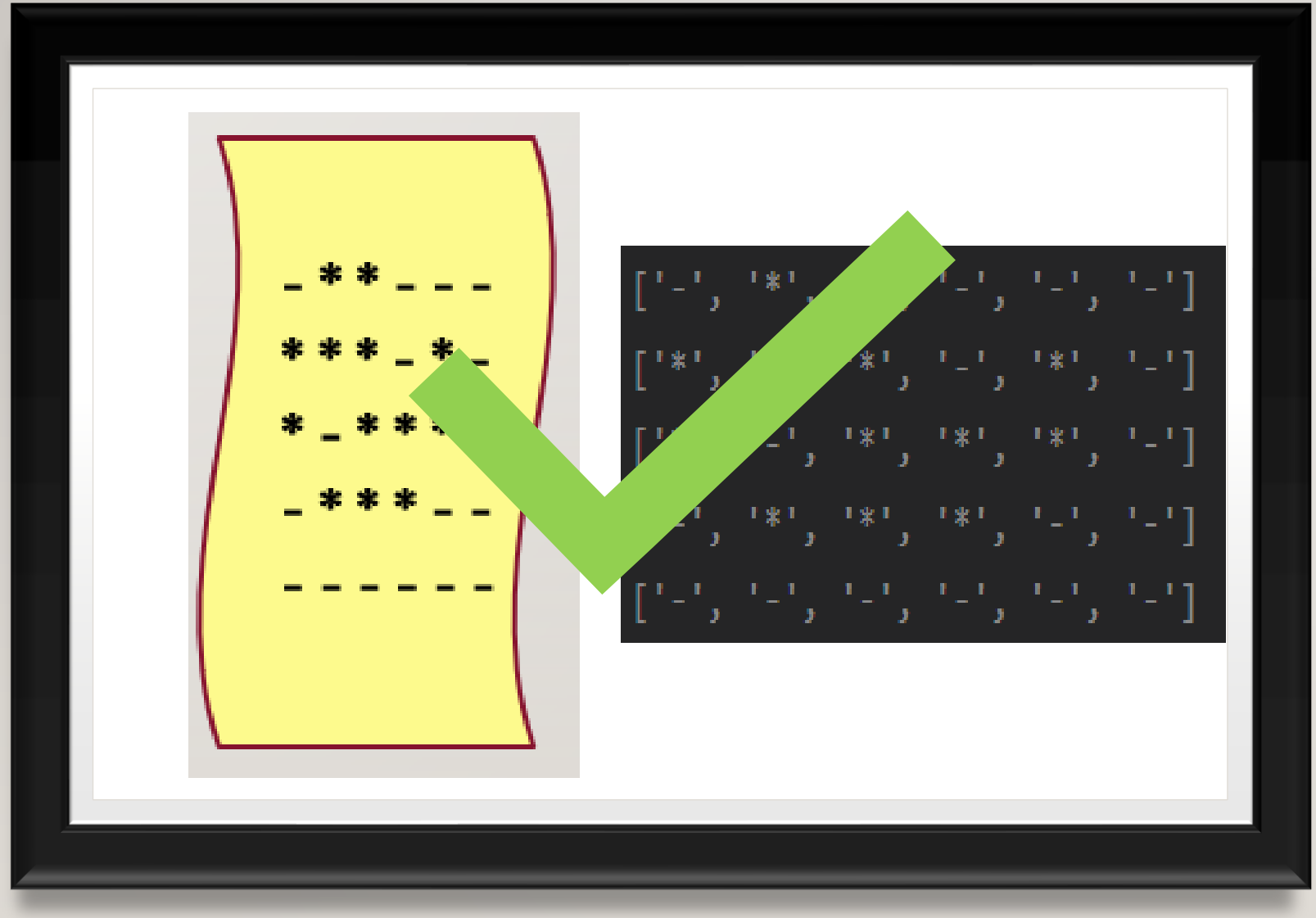
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"







COMPARISON WITH EXPECTATIONS

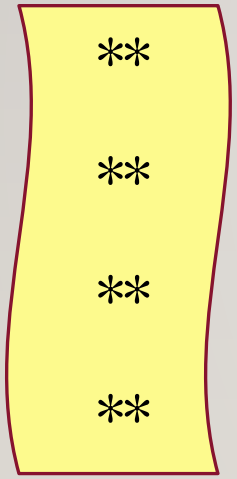


COMPARISON WITH EXPECTATIONS

EXAMPLE #2

- Execution of the algorithm with the input:

4 2



1 0 0 1

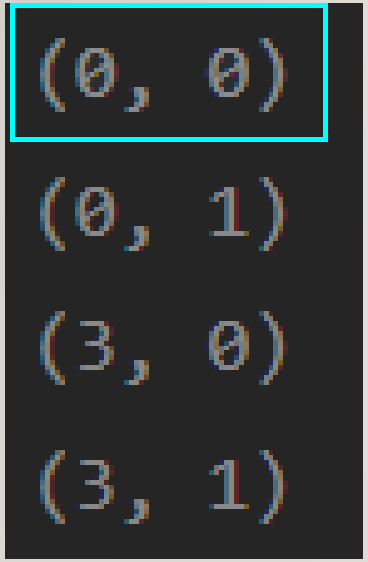
1 1

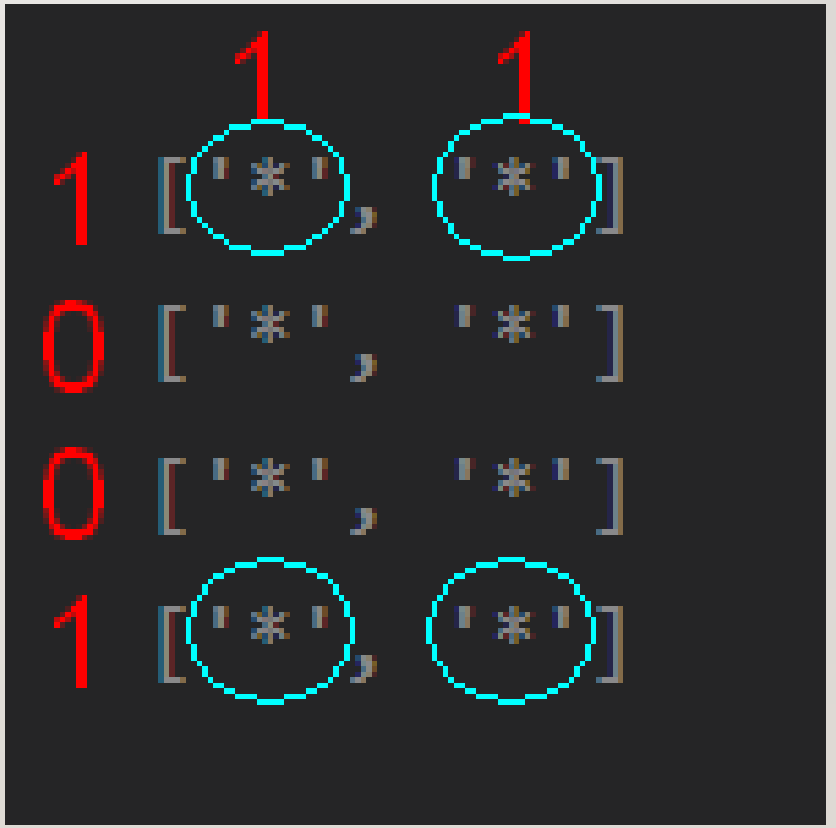
- Which should result in:

NO

PATH OF POSSIBILITIES:

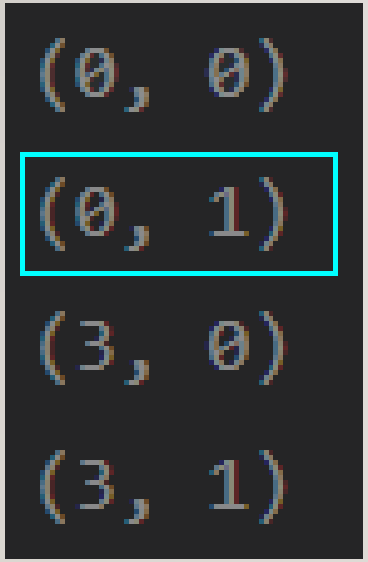
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

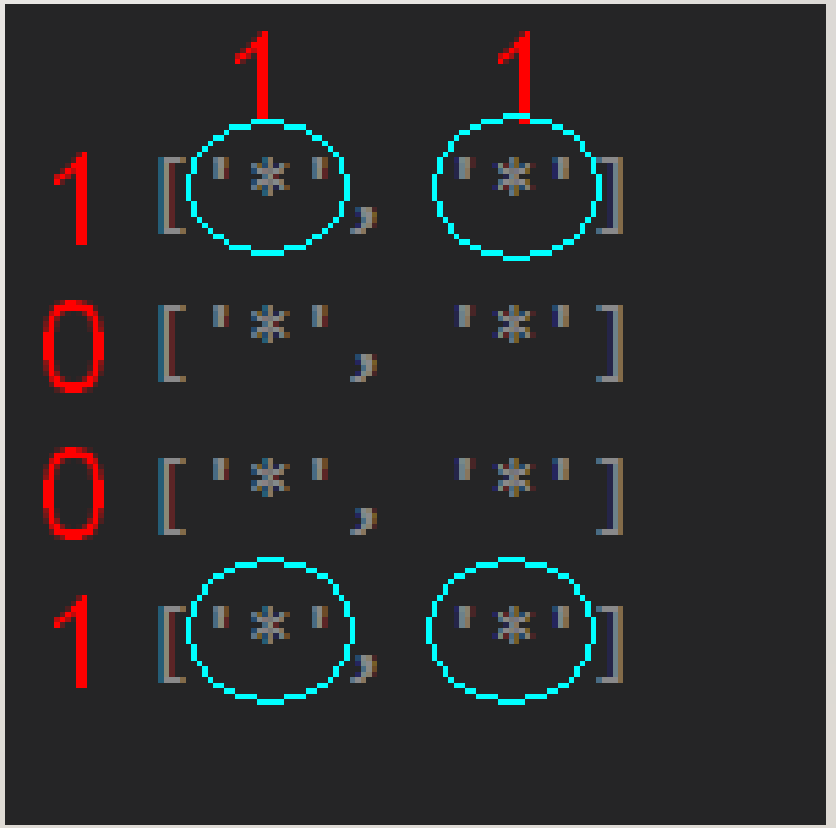




PATH OF POSSIBILITIES:

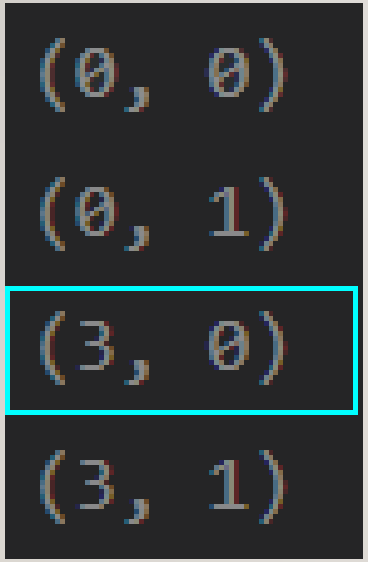
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

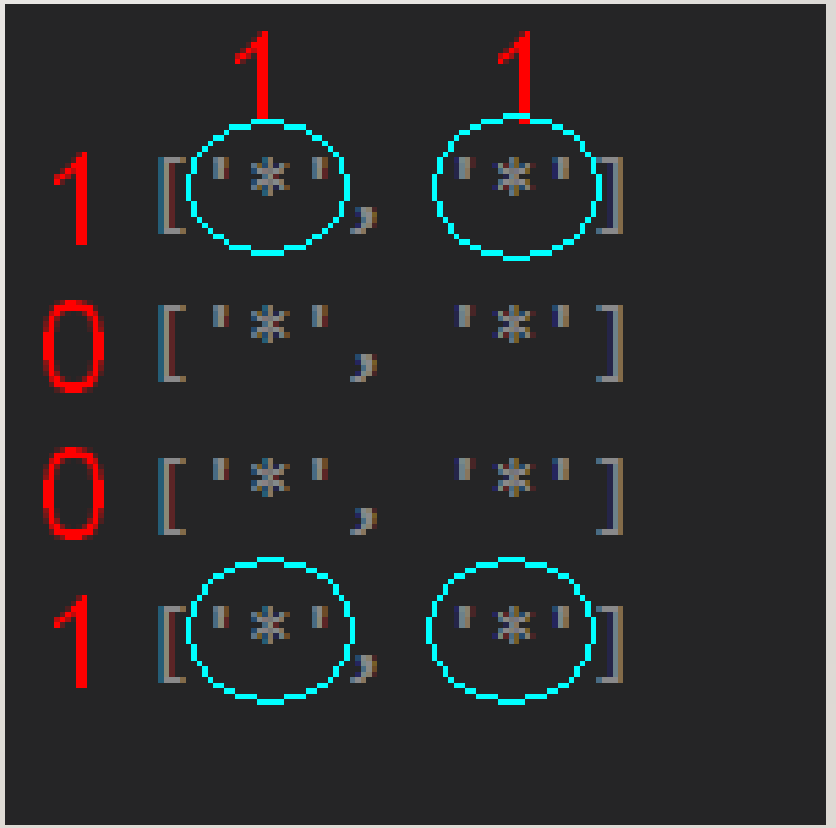




PATH OF POSSIBILITIES:

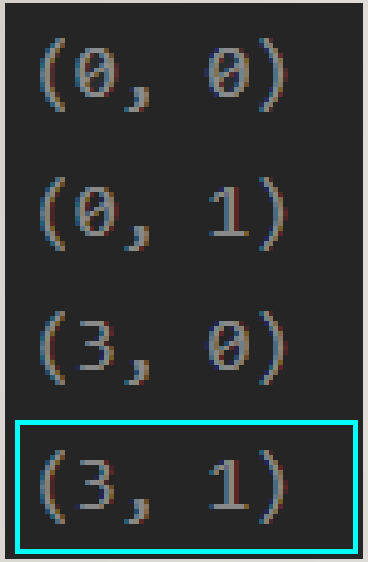
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

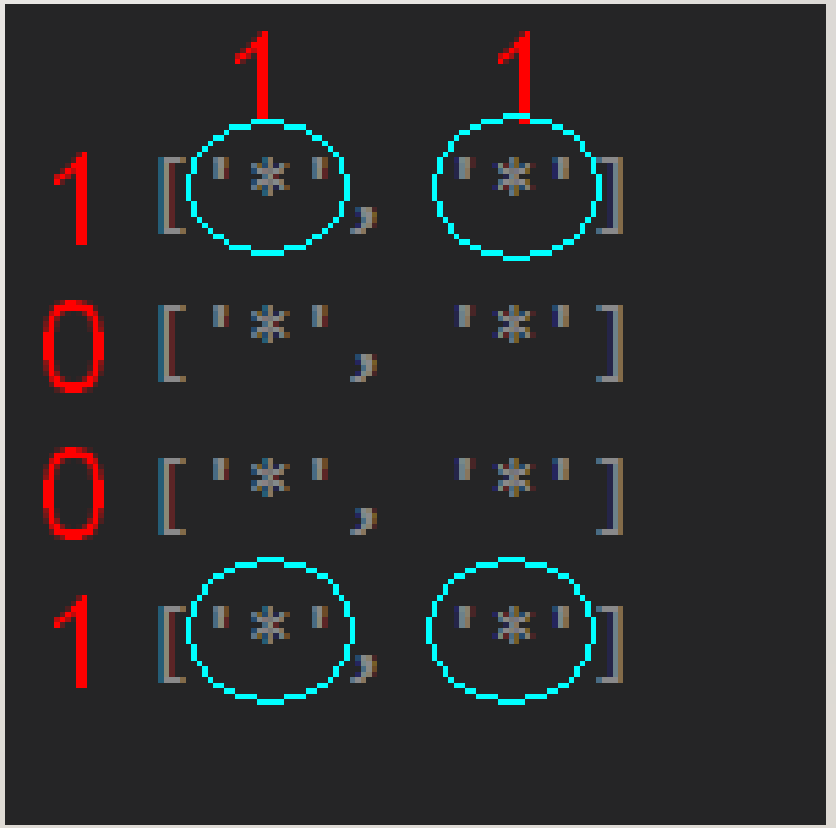




PATH OF POSSIBILITIES:

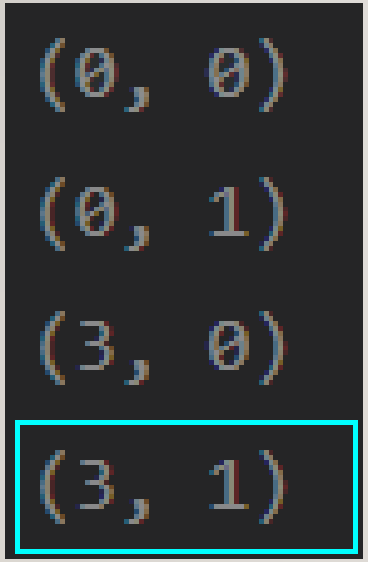
- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"

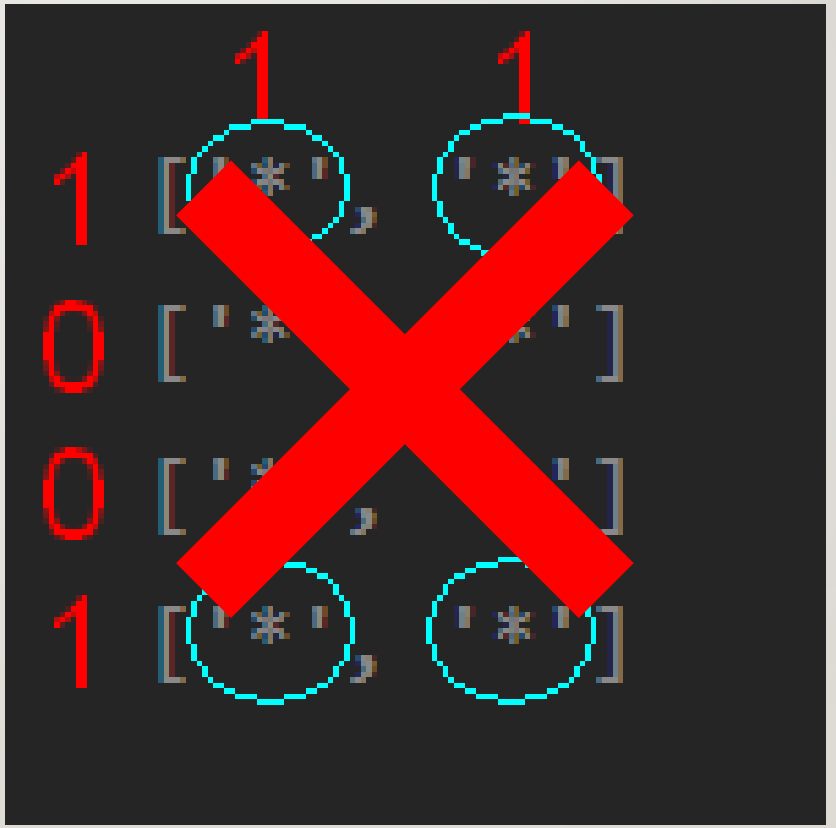




PATH OF POSSIBILITIES:

- If there are as many possibilities as missing "-" in a row or column: **Replace**
- If a possibility is no longer possible: **Remove**
- If no simplification is possible: **Return** "**NO**"





COMPARISON WITH THE BASIC ALGORITHM

My algorithm:

#4275938	October 27, 2020 21:53	27712 kB	0.27 s	Python 3.x / 100%	accepted
#4275942	October 27, 2020 21:56	27712 kB	0.29 s	Python 3.x / 100%	accepted

Basic algorithm:

#4275940	October 27, 2020 21:55	27728 kB	0.35 s	Python 3.x / 100%	accepted
#4275941	October 27, 2020 21:55	27728 kB	0.31 s	Python 3.x / 100%	accepted

=> No **real** difference in execution time between the two algorithms

COMPARISON WITH THE BASIC ALGORITHM

My algorithm is easier to understand (recursion is always complex).

And is more effective in cases with numerous possibilities.

THANK YOU FOR YOUR ATTENTION