

THE EIGHT TILES PUZZLE

What is a 8-PUZZLE?

Is a puzzle with 8 TILES and 1 EMPTY SPACE

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

EMPTY SPACE

A tile can be moved in the empty spot if they are ADJACENT

FOR EXAMPLE

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

MOVE 4

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

MOVE 8

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

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

MOVE 6

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

MOVE 7

Which is the objective?

The OBJECTIVE is to obtain the ordered configuration

THE OBJECTIVE

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



How to represent it in python?

We can represent the puzzle as a MATRIX (list of lists)

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

represented as $[[1, 4, 3], [7, 0, 6], [5, 8, 2]]$

✗ lists are NOT HASHABLE OBJECTS so we cannot use sets to store them

a TUPLE (a series of numbers)

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

represented as $(1, 4, 3, 7, 0, 6, 5, 8, 2)$

✓ lists are hashable

How to model the actions?

Given a puzzle we can move only the tiles that are adjacent to the empty spot.

In the example we have 4 possible actions: MOVE 4, MOVE 8, MOVE 7, MOVE 6.

with this formulation we have 8 possible actions: MOVE 1/2/3/4/5/6/7/8

there is a more compact formulation! If we focus on the empty tile we can say that the empty tiles moves UP/DOWN/LEFT/RIGHT

! WARNING:

a random eight tiles puzzle is NOT ALWAYS SOLVABLE.

Before running a search verify that your puzzle is solvable (for example generate it online)

in this way we can represent any action with only 4 kind of moves!

MOVE UP

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

MOVE DOWN

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

MOVE RIGHT

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

MOVE LEFT

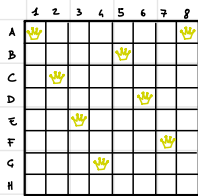
| | | |
|---|---|---|
| 1 | 4 | 3 |
| | 7 | 6 |
| 5 | 8 | 2 |

THE EIGHT QUEENS PROBLEM

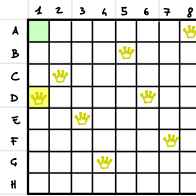
How the problem is formulated?

There is a 8×8 chess with 8 queens. Each queen can be moved in her column

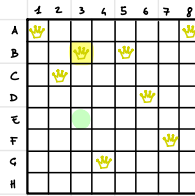
if we move a queen in the column of another queen they will be in conflict for sure



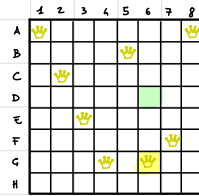
SOME POSSIBLE ACTIONS



MOVE A1 TO D4



MOVE E3 TO B3

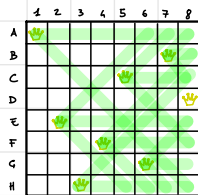


MOVE D6 TO G6

Which is the objective?

The OBJECTIVE is to obtain a chess with 0 conflict between queens

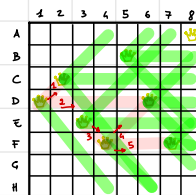
THE OBJECTIVE



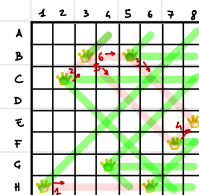
0 QUEENS IN CONFLICT



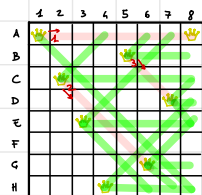
EXAMPLES WITH CONFLICTS



5 CONFLICTS



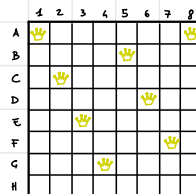
6 CONFLICTS



3 CONFLICTS

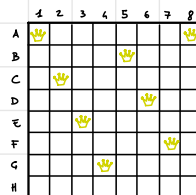
How to represent it in python?

We can represent the puzzle as \rightarrow a MATRIX (list of lists)
 \rightarrow a TUPLE (a series of number)



represented as $\left[\begin{matrix} A1 \\ 1, 0, 0, 0, 0, 0, 0, 1 \end{matrix} \right], \left[\begin{matrix} B5 \\ 0, 0, 0, 0, 1, 0, 0, 0 \end{matrix} \right], \dots$

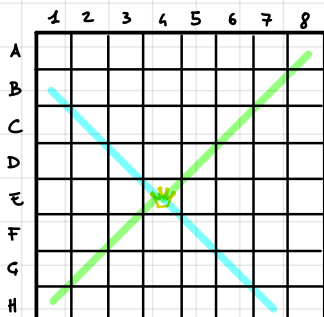
\times lists are NOT HASHABLE OBJECTS so we cannot use sets to store them



represented as $\left(\begin{matrix} A1 & C2 & E3 & G4 & B5 & D6 & F7 & H8 \\ 0, 2, 4, 6, 1, 3, 5, 0 \end{matrix} \right)$
 column 1

How to compute the conflicts?

Let's consider a chess with only one queen



If the queen is in position E4 \rightarrow this means that the queen is the 5th row and 4th column.

\rightarrow we can notice that the conflict along the diagonals can be described as follows:

\rightarrow N/E are D5, C6, B7 \Rightarrow the row decreases \downarrow
 the column increases \uparrow
 \rightarrow S/W are F3, G2, H1 \Rightarrow the row increases \uparrow
 the column decreases \downarrow

\rightarrow the sum between rows and columns indexes is constant!

\rightarrow S/E are F5, G6, H7 \Rightarrow both row and columns increase $\uparrow\uparrow$
 \rightarrow N/W are D3, C2, B1 \Rightarrow both row and columns decrease $\downarrow\downarrow$

\rightarrow the absolute difference between rows and columns indexes is constant!