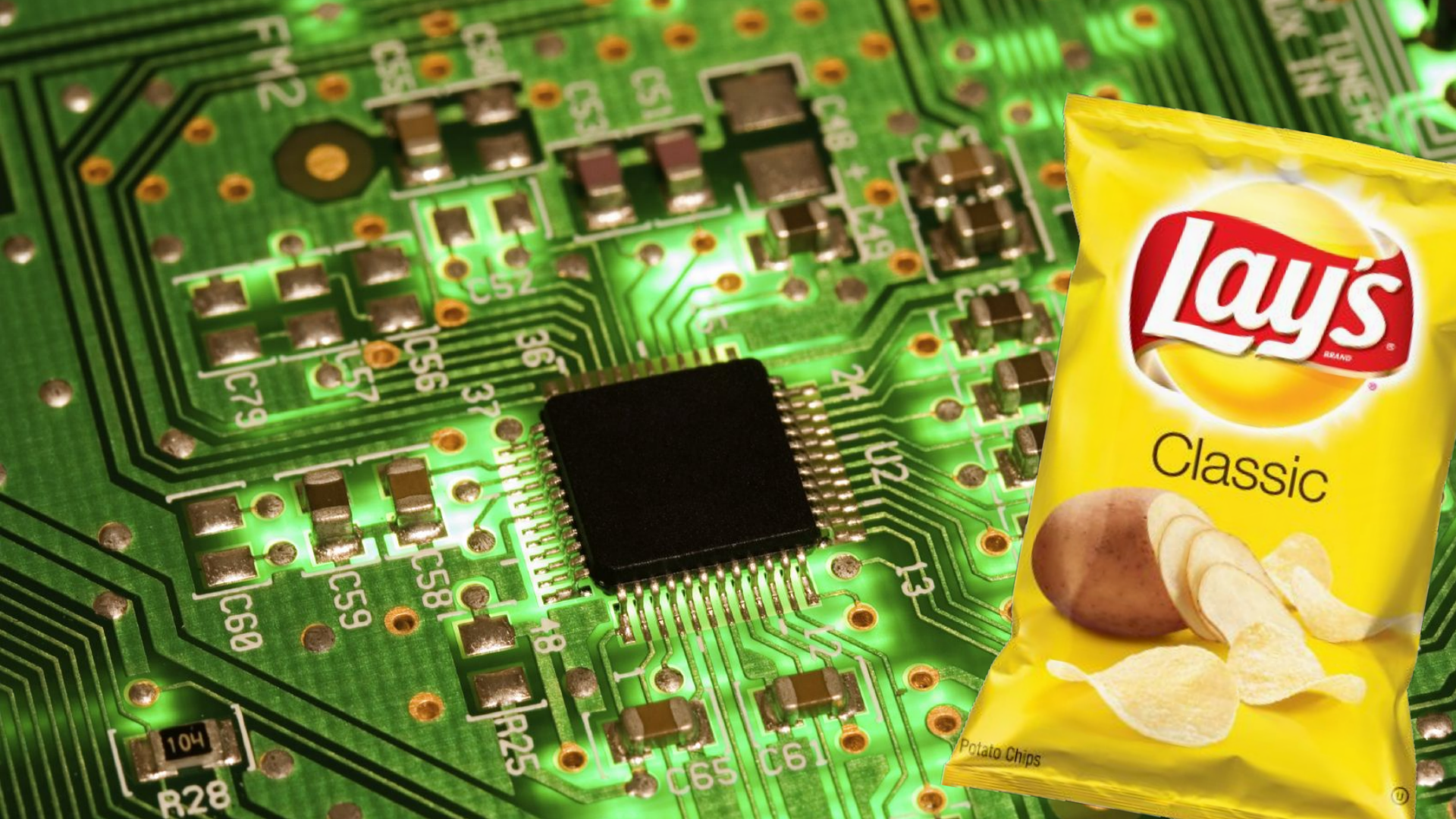
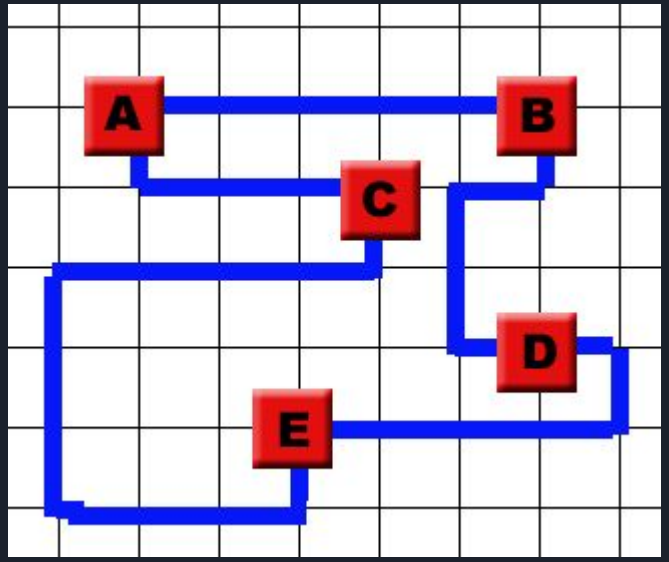
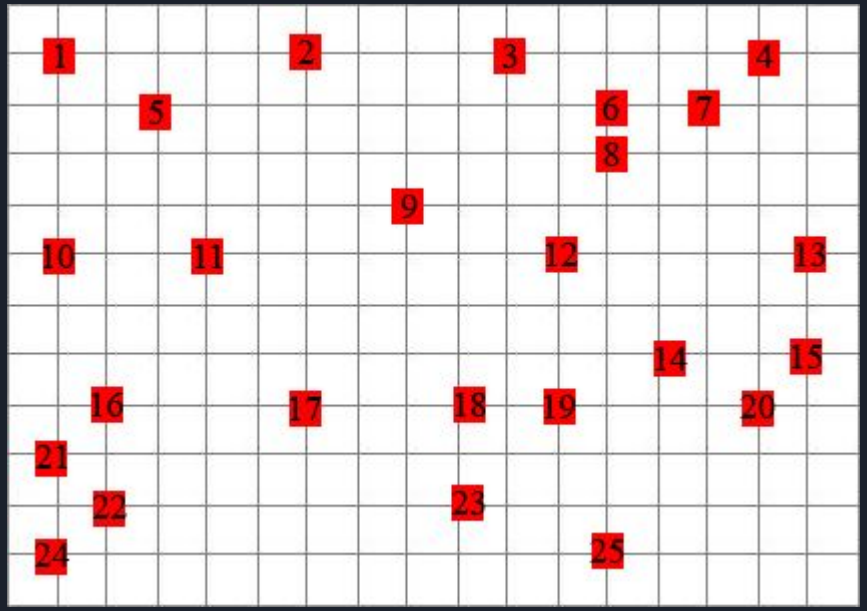
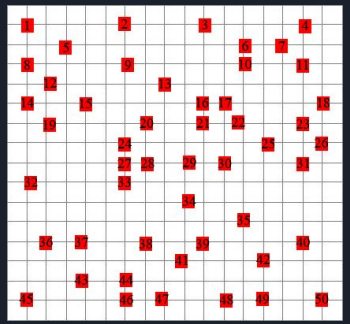
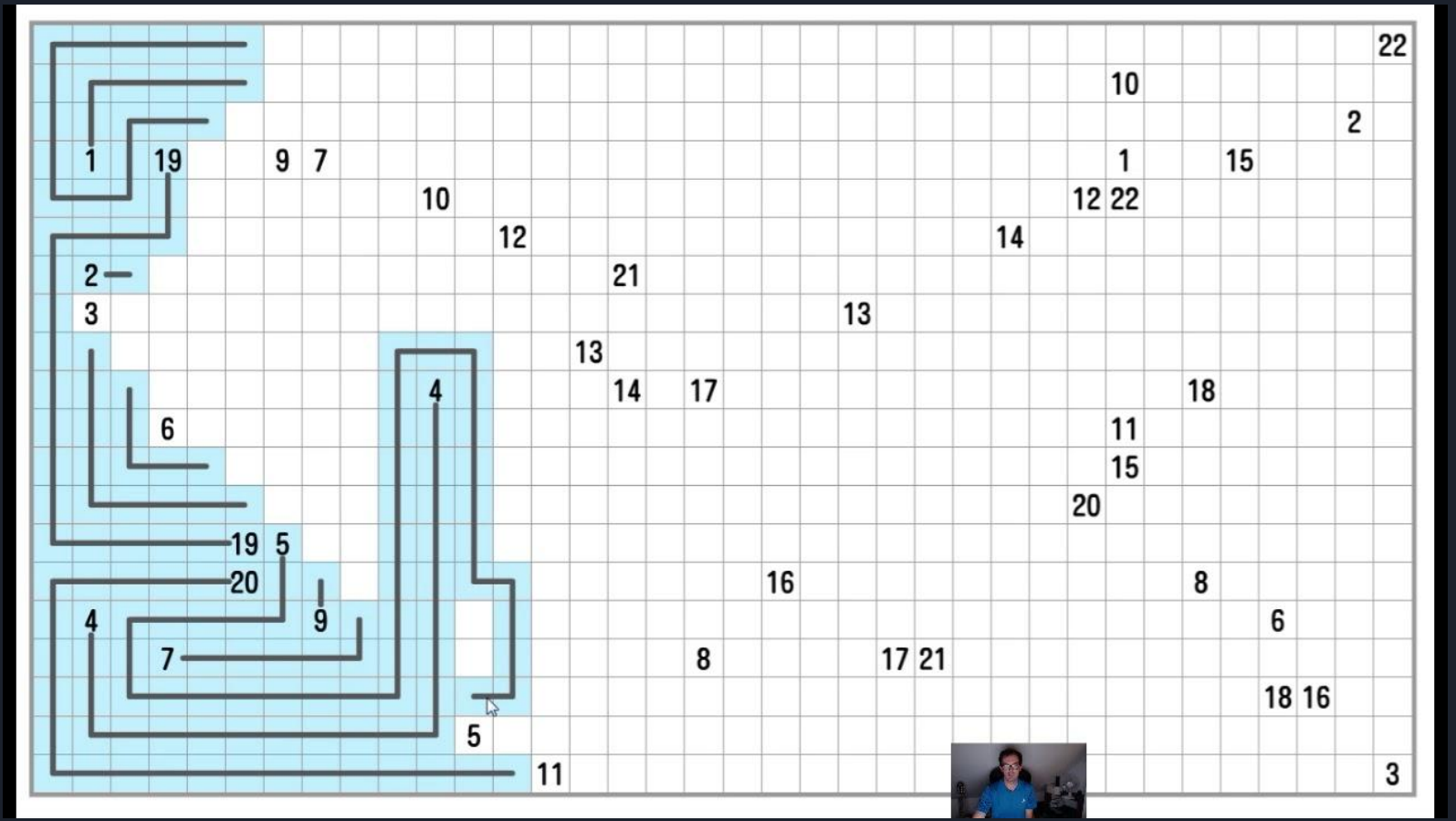

Lay's
Classic
Potato Chips



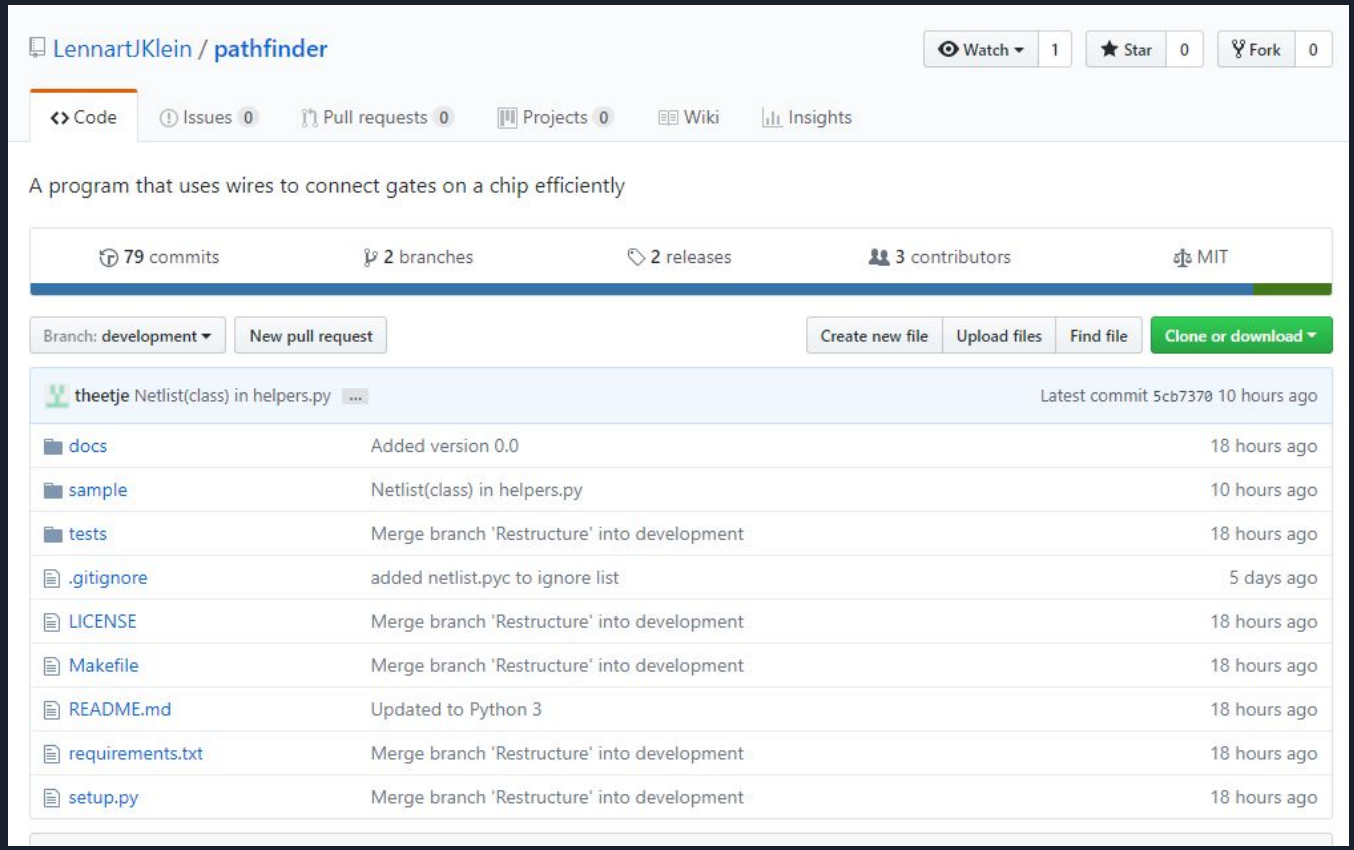




OPDRACHT


22
10
2
1
19
9
7
1
15
10
12
22
12
14
2
21
3
13
13
4
14
17
18
6
11
15
20
19
5
20
16
8
4
9
6
7
8
17
21
18
16
5
11
3

NUMBERLINK


LennartJKlein / pathfinder
Watch 1
Star 0
Fork 0
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Insights
A program that uses wires to connect gates on a chip efficiently
79 commits
2 branches
2 releases
3 contributors
MIT
Branch: development
New pull request
Create new file
Upload files
Find file
Clone or download
theetje Netlist(class) in helpers.py ...
Latest commit 5cb7370 10 hours ago
docs Added version 0.0 18 hours ago
sample Netlist(class) in helpers.py 10 hours ago
tests Merge branch 'Restructure' into development 18 hours ago
.gitignore added netlist.pyc to ignore list 5 days ago
LICENSE Merge branch 'Restructure' into development 18 hours ago
Makefile Merge branch 'Restructure' into development 18 hours ago
README.md Updated to Python 3 18 hours ago
requirements.txt Merge branch 'Restructure' into development 18 hours ago
setup.py Merge branch 'Restructure' into development 18 hours ago

GITHUB

PLAN VAN AANPAK

✓ v0.0 Set up architecture for the files, codes and folder

✓ v0.1 Write a program to print a two-dimensional grid
- ✓ v0.1.1 Update labels to start at 0
- ✓ v0.1.2 Align vertical labels to the left.

✓ v0.2 Calculate the most efficient route between point A and B on a grid.

v0.3 Create a data structure for gates on the grid (with simple data)
- ✓ v0.3.1 Load the ID of a tuple in set_gate()
- ✓ v0.3.2 Don't load the header of the CSV files
- ✓ v0.3.3 Refer to Westly White in the comments of the board-class
- ✓ v0.3.4 Call calculatePath in pathfinder.py
- v0.3.5 Read gates.csv and save the data in an accessable dict (creating a class that reads gates.csv)

v0.4 Load netlists from a .txt file to a data structure (instead of hard coded data in the helpers file)

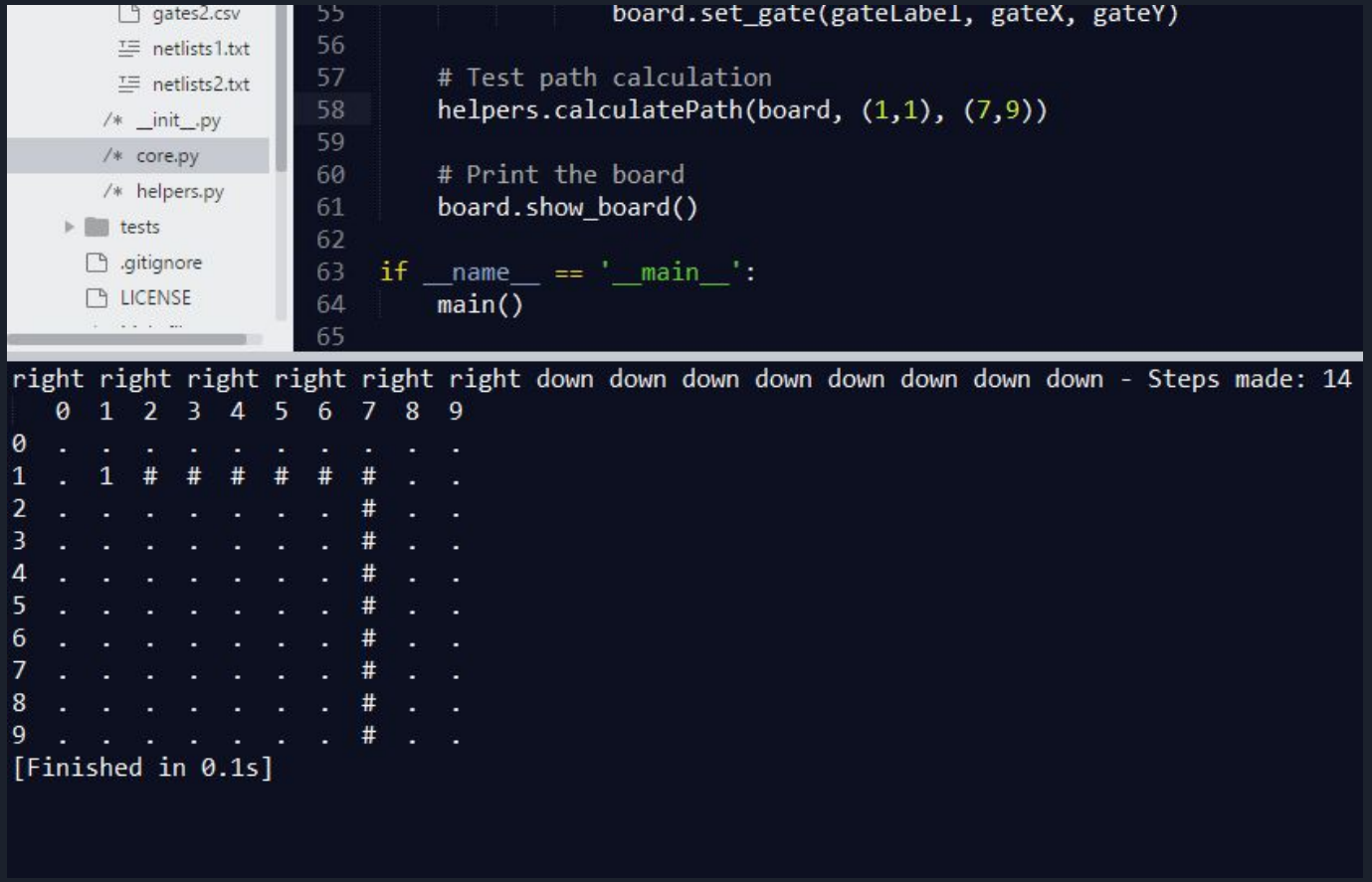
v0.5 Calculate a route between point A and B on a grid (based on a simple netlist and gateslist)

v0.6 Evade obstacles on a grid while calculating a route

v0.7 Remember the walked path in a data structure

v0.8 Calculate 2 routes, which evade obstacles and each other.

v0.9 Make the program modulair and installable (by using setup.py)


gates2.csv
netlists1.txt
netlists2.txt
__init__.py
core.py
helpers.py
tests
.gitignore
LICENSE
board.set_gate(gateLabel, gateX, gateY)
# Test path calculation
helpers.calculatePath(board, (1,1), (7,9))
# Print the board
board.show_board()
if __name__ == '__main__':
main()
right right right right right right down down down down down down down down - Steps made: 14
0 1 2 3 4 5 6 7 8 9
0 . . . .
1 . 1 # # # # # . .
2 . . . . . . # . .
3 . . . . . . # . .
4 . . . . . . # . .
5 . . . . . . # . .
6 . . . . . . # . .
7 . . . . . . # . .
8 . . . . . . # . .
9 . . . . . . # . .
[Finished in 0.1s]

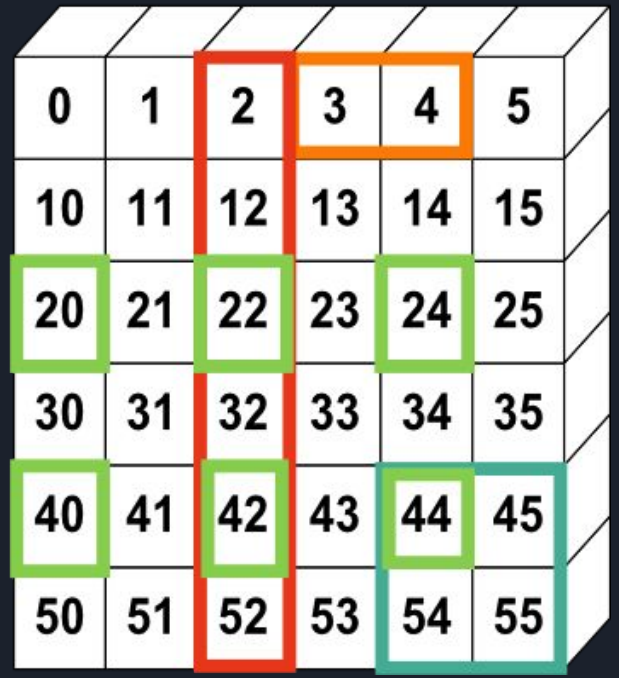
PREVIEW

```
>>> a[0,3:5]
array([3,4])

>>> a[4:,4:]
array([[44, 45],
       [54, 55]])

>>> a[:,2]
array([2,12,22,32,42,52])

>>> a[2::2,::2]
array([[20,22,24]
       [40,42,44]])
```



1. Bekend met Numpy? Andere suggesties voor een 3D-board?

2. Waar zit volgens jou de moeilijkheid van deze opdracht?

VRAGEN