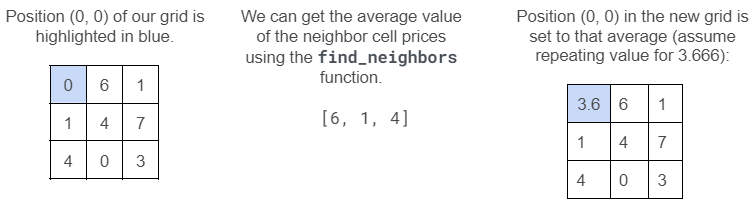
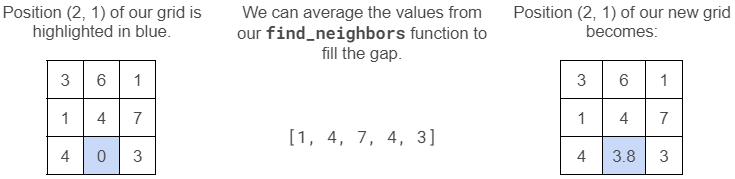
# **Sim City Land Value Calculator Version 3: Estimate missing land values**

Some of the land values are missing! Let’s estimate the missing value of a cell with value 0, using the values of its neighbors.

Once you have calculated the missing land values, display the updated grid.

**IMPORTANT: Do not modify the original grid, but create a new grid where no cell has missing values.**

Study the examples below to see how the fill\_gaps() function should work:

You can safely assume that if a cell value is missing (the value is given as 0), then none of the neighboring cell values are also missing. So you will always take averages over non-zero values.

## **What to do**

Implement the fill\_gaps() function that estimates the missing land values within the grid and updates the grid.

You must use the following template:

| **def** **create\_grid**(filename: str) -> list[list[int]]:  """  Create a grid of land values from a file  """  *# Implemented in Version 1*  **pass**   **def** **display\_grid**(grid: list[list[int]]) -> **None**:  """  Display a grid of land values  """  *# Implemented in Version 1*  **pass**    **def** **find\_neighbor\_values**(grid: list[list[int]], row: int, col: int) -> list[int]:  """  Find the neighbors of a cell  """  *# Implemented in Version 2*  **pass**    **def** **fill\_gaps**(grid: list[list[int]]) -> list[list[int]]:  """  Fill the gaps in the grid  Creates a new grid with the same dimensions as the original grid  Calls find\_neighbor\_values() to find the neighbors of each cell  Do NOT modify the original grid!  """  *# TODO: Implement this function*   **def** **main**() -> **None**:  """  Main program.  """  grid = create\_grid("data\_0.txt")  print("Sim City land values:")  display\_grid(grid)  print("\nCalculated Sim City land values:")  new\_grid = fill\_gaps(grid)  display\_grid(new\_grid) |
| --- |

## **Hints**

* Remember that lists in Python are mutable. It’s important to create a new list to store the new values.
* You can use the deepcopy() function to make a copy of a list.

## **Program name**

Save your program as simcity3.py.

## **Demo**

In this demo, data\_1.txt is used.

<https://asciinema.org/a/YXxXjyRiVAiVXd853d2P9hVsI>

## **Testing**

To make sure your program works correctly, you should test it.

Good news: we wrote the unit tests for this version as well: [test\_simcity3.py](https://drive.google.com/file/d/1CPfKOVhhpHoTIdSG0exLkSJJmYXb3Fzt/view?usp=sharing)

To test your fill\_gaps() function, simply run the unit tests:

| $ python -m pytest test\_simcity3.py |
| --- |

All tests should pass.

You should also manually test your program with different input files.

* Run your program with python simcity3.py with data\_0.txt Your program should print:

| Sim City land values:  1 0 3 4   5 6 7 8   9 10 11 12   13 14 15 16   Calculated Sim City land values:  1 4 3 4   5 6 7 8   9 10 11 12   13 14 15 16 |
| --- |

* Run your program with python simcity3.py with data\_1.txt Your program should print:

| Sim City land values:  76000 0 54000 16000 83000   27000 49000 62000 0 31000   0 48000 53000 22000 19000   71000 37000 63000 41000 0   83000 25000 0 16000 59000   Calculated Sim City land values:  76000 53600 54000 16000 83000   27000 49000 62000 42500 31000   46400 48000 53000 22000 19000   71000 37000 63000 41000 31400   83000 25000 36400 16000 59000 |
| --- |

* Run your program with python simcity3.py with data\_2.txt Your program should print:

| Sim City land values:  94000 64000 30000 0 14000 92000   37000 49000 50000 29000 35000 0   0 88000 85000 96000 60000 22000   13000 44000 73000 0 45000 53000   20000 33000 67000 71000 82000 0   36000 0 62000 55000 44000 75000   Calculated Sim City land values:  94000 64000 30000 31600 14000 92000   37000 49000 50000 29000 35000 44600   46200 88000 85000 96000 60000 22000   13000 44000 73000 72375 45000 53000   20000 33000 67000 71000 82000 59800   36000 43600 62000 55000 44000 75000 |
| --- |

* Run your program with python simcity3.py with data\_3.txt Your program should print:

| Sim City land values:  24000 57000 50000 43000   38000 0 16000 62000   51000 25000 49000 0   0 76000 19000 34000   Calculated Sim City land values:  24000 57000 50000 43000   38000 38750 16000 62000   51000 25000 49000 36000   50667 76000 19000 34000 |
| --- |

## **Submitting**

Submit simcity3.py via eClass.

**Copyright**

I. Akhmetov, J. Schaeffer, M. Morris and S. Ahmed, Department of Computing Science, Faculty of Science, University of Alberta (2022).