

Project 16 (Minesweeper). Implement the game of Minesweeper.

Minesweeper is a single-player puzzle video game. The objective of the game is to clear a rectangular board containing hidden "mines" or bombs without detonating any of them, with help from clues about the number of neighboring mines in each field. The game originates from the 1960s, and has been written for many computing platforms in use today. It has many variations and offshoots.

You are asked to work on the following variation: upon start, the user is allowed to click anywhere in the field.

- If the user clicks on the square containing a mine, the mine is revealed, the player loses the game.
- If no mine is revealed, a digit is instead displayed in the square, indicating how many adjacent squares contain mines.
- If no mines are adjacent, the square becomes blank, and all adjacent squares will be recursively revealed.

The player uses this information to deduce the contents of other squares, and may either safely reveal each square or mark the square as containing a mine.

Here is the link to the online game: <http://minesweeperonline.com/>

This game requires the use of a graphics library. You can either use **cs1graphics** supplied with the textbook used in CSI 32 or you can use **tkinter**. You will have to be able to work with event handlers. Let me know if you didn't cover them in CSI 32.

You are asked to define and use the following classes to avoid grabbing the code from an online resource. The methods are suggested, but you can define your own if you don't like these. Just keep the idea in place.

class Cell: will represent each cell of the field.

Cells have a value associated with them:

'0' stands for 0 mines around

'1' stands for 1 mines around

'2' stands for 2 mines around

...

'8' stands for 8 mines around

'9' stands for the mine is placed in this cell

Cells can be visible/not visible, and can be locked (so that we don't open it by accident, - it means that we think there is a mine there) / unlocked.

class InputOutput: will represent the both, the input and the output in a graphics window. You can of course split it into two classes ... at the time I was working on it it seemed "easier" to merge them (so that both types of input/output operations have access to the graphics window).

class Game: will take care of the entire game

Finally, this is something you can have in a file Minesweeper.py to run the game:

```
from Game import *

# Left click of the mouse is to open a cell
# Mouse hovering over a cell + keyboard key(any) = lock the cell, it
# is a potential mine

def main():

    g = Game(10,10) # size, number of mines
    print(g)
    g.playGame()

main()
```

See the files with some “skeletons” of the classes and some code as well.

This is all done in Python 3. However, if you want to use C++ and know a good graphics library that can do event handling, feel free to do the project in C++.