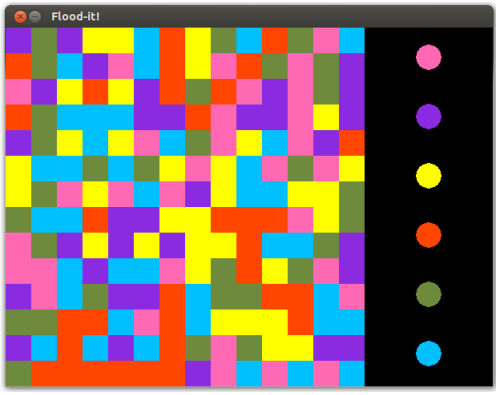
CS343 Homework 1 - Flood-It!

Due 11:59pm, August 30, 2013

1 The Game

The goal of the player is to *flood* the entire board with a single color within a given number of moves, e.g. for a 14x14 board, the player wins the game if they flood the board within 25 moves. The player can choose a new color for the flooded region by clicking one of the circles to the right of the board or by pressing the following keys:



- a: pink
- s: violet
- d: yellow
- z: red
- x: olive
- c: blue

2 Your Task

We have written most of the game but we need your help in finishing it. We would like you to write the `flood` function. The `flood` function takes two arguments:

- `color_of_tile` - a Python dictionary that maps a tile coordinate to its current color. A coordinate is represented as a pair (a 2-tuple) containing the x and y values, with (0,0) representing the upper left corner. The x coordinates increase as you go to the right and the y coordinates increase as you go down.
- `flooded_list` - a Python list (an array) of coordinates for the flooded area. These tiles will always have the same color.

We say that a tile is *adjacent* to another tile if it is directly above, below, left, or right, that is, if the two tiles share a side. You will find some helpful functions in a file named `utilities.py`: the functions named `up`, `down`, `left`, and `right` compute the coordinates of the adjacent tiles. There is also a function named `in_bounds` in `utilities.py` that tells you whether a coordinate is on the board.

An *X-colored region* is a set of tiles defined as follows:

- A tile of color X is an X-colored region.
- If tile T is color X and adjacent to a tile in an X-colored region R, then $T \cup R$ is an X-colored region.

Given a `flooded_list` whose tiles are color X, the flood function should add every X-colored region to the `flooded_list`, provided the region contains a tile that is adjacent to a tile in the `flooded_list`.

3 Analysis

After implementing and debugging your `flood` function, run `floodit` in batch mode, which produces a graph of the execution time (y-axis) versus the size of the board (x-axis, number of tiles). Look at the graph. What function (roughly) fits that graph? (Hint: possibilities to think about are $f(n) = n, f(n) = n^2, f(n) = n \lg n$.)

4 Logistics

There is a github@IU organization named C343-Fall2013. Within that organization there is a repository for you (your github user name followed by "-c343"). Fork that repository into your own github@IU account. You'll find the code for Flood-It! in the sub-directory named `hw1`. The `flood` function is in the file named `flood.py`. (Not to be confused with `floodit.py`, the main program.)

You need to download and install pygame version 1.9, which you can obtain at `www.pygame.org`. Note that there are different download files of pygame for different versions of Python. They do not have Python 3 versions for many operating systems (such as Mac), so everyone needs to use the version of pygame for Python 2.7 (look for `py2.7` in the name of the pygame download file). If Python 2.7 is not already installed on your computer, you will need to install that as well.

Place the answer to the analysis question and a one-paragraph description of your `flood` implementation in the `README.md` file within `hw1`. When you are finished with the assignment, initiate a pull request.