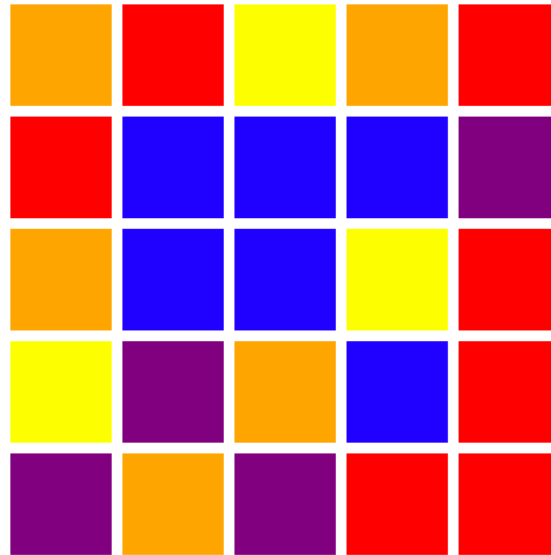
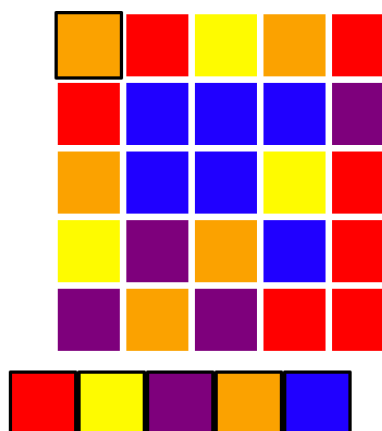


# Color Flipping – 2022

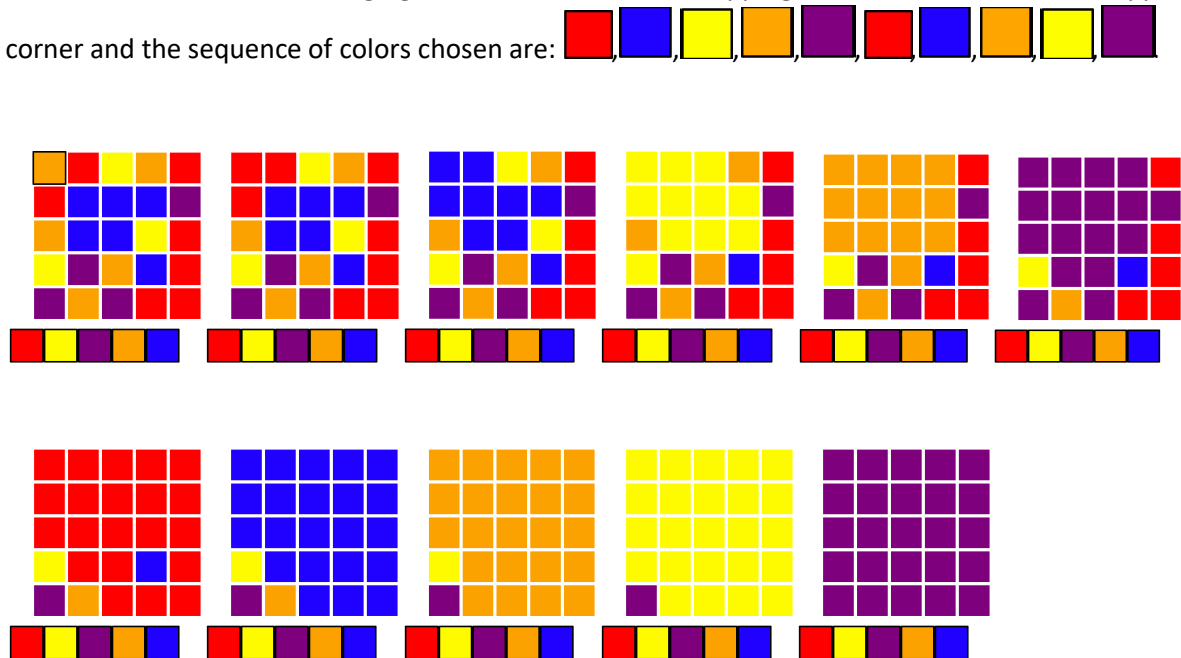


## OVERVIEW

In this assignment, you are going to design and develop a GUI (Graphical User Interface) based color-flipping game in python. The game is composed of a rectangular board divided into a number of flat colored tiles of various colors as illustrated in the figure above. The goal of the game is to turn all tiles to the same color by choosing one tile on the board and then changing its color to another color. The newly selected color will spread to all its neighboring tiles which match the original color of the chosen tile. The player continues selecting another tile and changing its color to another until all the tiles within the board flipped to the same color. In addition to the rectangular board, an array of colors is shown below the game board. These are the colors to which the player can choose to flip the color of the selected tile, as shown below.

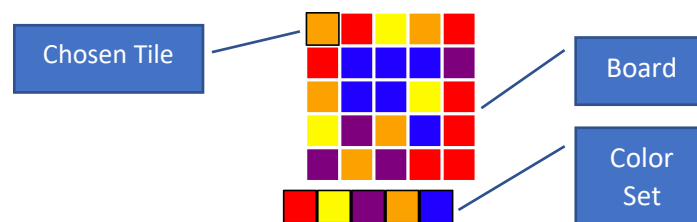


As an illustration, the following figures show the results of flipping the color of the tile at the upper left corner and the sequence of colors chosen are:



## SCOPE

- Design the color-flipping game using the standard python module “turtle”, including the following components:
  - Board
  - Flat-colored Tiles
  - GUI (Mouse Click, Tile and Color Selection)
- When the game starts, it creates the rectangular board with a 5x5 array of colored tiles, each filled with a randomly chosen color from a fixed set of 5 different colors. You are free to choose your own set of colors, or use the ones from this example. The color set is displayed as a color bar under the game board as shown below.



- During the game, the player uses the mouse to select one tile from the board and the player can also change to another tile by freely clicking another tile, the last selected tile will be framed with a border (black in this example), finally the player picks one color from the color set to begin flipping the color of the selected tiles, plus all its neighboring tiles matching the original color of the selected tile. The selected tile will be reset (border removed) as soon as the color-

flipping logic begins. After the color-flipping logic is completed the player is then allowed to choose another tile and another color to flip.

- Choose a reasonable size for the tiles including the color bar, minimum 60x60 pixels.
- Set the width of the border to about 5 pixels
- Do not terminate your program when the player successfully flipped all tiles to one color.
  - a. Simply keep your program up and running and let the player to choose different colors as desired.
- Coding Styles
  - a. Ensure that your program follows the proper layout structure as discussed in class.
  - b. You might declare global variables used for this assignment, ensure that a consistent naming convention is in place to differentiate various variable scopes.

**NOTE:**

- Keep your entire source code in ONE SINGLE file.
- Use only standard python modules
- In your design stick ONLY to functions, in other words, no class objects of your own.

# STARTUP OPTIONS

Not applicable

# SKILLS

In this assignment, you will be trained on the use of the followings:

- Understand differences between console-based vs GUI based application
- Use built-in turtle module to design the program as per scope
- Use standard objects (strings, numbers & lists)
- GUI interaction
- Variable Scope
- Functions for program structure and decomposition

# DELIVERABLES

1. Program source code (A1\_School\_StudentID.py)

where School is SSE, SME, HSS, FE or LHS and StudentID is your 9-digit student ID.

Submit the python file by due date to the corresponding assignment folder under “Assignment (submission)”

For instances, a SME student with student ID “119010001” will name the program as follows:

- A1\_SME\_119010001.py:

5% will be deducted if file is incorrectly named!!!

# LATE SUBMISSION

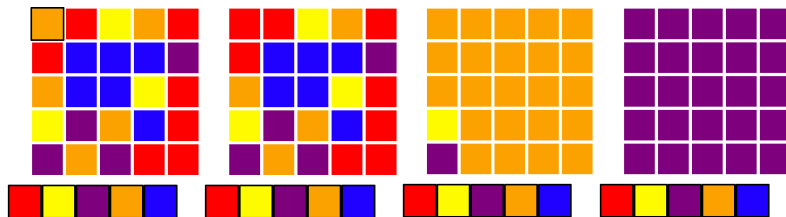
For late submission, a daily penalty amounting to 10 points will be deducted against the final grade, up to maximum 3 days (30 points).

No further submission will be accepted 3 days after the set deadline.

## TIPS & HINTS

- Follow the layout structure as mentioned in class (import, declarations, functions, main process).
- Clearly name and comment your global variables.
- Refer to python website for program styles and naming convention (PEP 8).
- Use Turtle() as objects for the color tiles and color set
- Use the shape() function to set the shape for your objects, or pass the shape as string to the Turtle(). Note: use the simple, built-in shape such as “square”.
- Use clone() to duplicate copy of the turtle objects.
- Use write() to display a text on the screen.
- Remember to set the pen in “up” position to avoid line drawing.
- Use Screen() to configure the game area and use onclick() to capture mouse-click event and the mouse’s current x,y position.
- Use tracer(0) to disable auto screen refresh and call update() to manually refresh the game area.
- Refer to <https://docs.python.org/3/library/turtle.html> for more information on Turtle Graphics.

## SAMPLE OUTPUT



## MARKING CRITERIA

- Coding Styles – overall program structure including layout, comments, white spaces, naming convention, variables, indentation, functions with appropriate parameters and return.
- Program Correctness – whether or the program works 100% as per Scope.
- User Interaction – how informative and accurate information is exchanged between your program and the player.
- Readability counts – programs that are well structured and easy-to-follow using functions to breakdown complex problems into smaller cleaner generalized functions are preferred over a function embracing a complex logic with nested conditions and sub-functions! In other words, a design with clean architecture with high readability is the predilection for the course objectives over efficiency.
- KISS approach – Keep It Simple and Straightforward.
- Balance approach – you are not required to come up with a very optimized solution. However, take a balance between readability and efficiency with good use of program constructs.

ITEMS	PERCENTAGE	REMARKS
CODING STYLES	10%	0% IF PROGRAM DOESN'T RUN
USER INTERFACE	10%	0% IF PROGRAM DOESN'T RUN
FUNCTIONALITY	80%	REFER TO SCOPE

## DUE DATE

April 22<sup>nd</sup>, 2022, 11:59:59PM