THE AMAZING PUZZLE

The final project for CSC 111
Mackie Zhou, Grace Li, Winnie Zong
Group 11

How to play the game?

1. Run the **main.py** file to play the game.
2. A python package, **pygame**, needs to be installed. See pygame official website to learn how to pip install pygame.

    https://www.pygame.org/wiki/GettingStarted#Pygame%20Installation

Game Description

Bored by problem sets and papers? Let our **Amazing Puzzle** sharp your brain! Choose a difficulty level and challenge yourself!

1. **How to win?**
   - Make the yellow and blue squares cover the two static grey squares.
2. **How to play?**
   - Choose a **difficulty level** from level 9, 99, and 999.
   - Control the movement of the yellow square and the blue square by hitting the **left, right, up, or down arrow keys** on your keyboard.
3. **Any rules?**
   - The two squares **move together** (i.e. if you hit the up arrow key, both squares will move up a grid)
   - **NO MOVEMENT when hitting a wall**. If only one of the square is stopped by the wall, the other one will move normally (this is a strategy you can use to move only one square).
   - The squares will be **transported by the numbers** around the game board. All the numbers appear in pairs and they are like channels between grids. If you move a colored square out of the board from one number, the colored square will reappear in the other grid surrounded by the same number.
   - **NO OVERLAP** of the two squares.
4. **Everything will be clear when you try it out!**

There are two functions in the Project.py file: 1. the main() function 2. the createsq() function

call the main() function

draw the components of the main page. There are four buttons drawn on the main page:
-- button 1 & 2 & 3 direct the user to the three different puzzles
-- button 4 directs the user to the guide page

detect event

if mouse click

if click button 1/2/3

if click button 4

if click the cross of the window

quit game

call the createsq() function. It creates:
-- 4 puzzle1 class objects if button 1 is clicked or
-- 4 puzzle2 class objects if button 2 is clicked or
-- 4 puzzle3 class objects if button 3 is clicked.
(the four newly created objects are returned by the creatsq() function)

draw the guide page components.
on the guide page are:
-- goal & guide & rules
-- a BACK button(directs the user back to the main page)

main game loop

detect event

detect event

if mouse click

if keyboard hit

if mouse click

if click the cross of
the window

if click the "BACK" button

left arrow key > move to the left
right arrow key > move to the right
up arrow key > move up
down arrow key > move down

if click the cross of
the window

quit game

call the main() function

quit game

check positions of the colored squares

if the two squares
cover the grey squares

else if the two
squares overlap

else

the user wins

back to their last positions

call the main() function

the loop continues