

# **ISP PLAN**

By Peter Ye

December 13th, 2019

# Animation Storyline

---

You are a master electrician named Megavolt. You have been captured by a group of masked gang members and thrown into a secret base hidden in the heart of Toronto. The gang has locked you up behind multiple layers of electrical security equipment. You must escape undiscovered - and not electrocuted, of course!

## Maze Design

---

The game now goes back in time to when you were still an apprentice working for Toronto Hydro. You are guided through a series of challenges in a maze. The maze goes through clever ways you can use short circuits and switches. The maze includes 4 stations of learning which are initially red and turn green when completed. After the player has completed all of the stations of learning, they can exit the maze of learning.

## Game of Testing

---

It is now time to test the player! They must manipulate the electrical switches using mouse input to turn off the motion detector and also use keyboard input to cross the high voltage electrical wire.

## List of Methods

---

Method	Purpose
introAnimation	A 30-40 second long animation about Megavolt, the main character of the game
setup	Sets window size and calls animatedIntroduction() and mainMenu()
goodbye	Shows the goodbye screen before the program exits

---

<b>Method</b>	<b>Purpose</b>
mainMenu	A menu for the player to select “Instructions”, “Maze of Learning”, or “Exit”
instructions	Shows detailed instructions on the game’s controls
mazeOfLearning	Displays the maze of learning
mazeStation1	Station 1 of the maze
mazeStation2	Station 2 of the maze
mazeStation3	Station 3 of the maze
mazeStation4	Station 4 of the maze
exitMaze	Checks that the player has completed all of the stations, displays a success message, and returns to main menu
gameOfTesting	Displays the game of testing
escapeFailed	Called when player fails the game of testing
escapeSuccessful	Called when player beats the game of testing
drawCharacter	Draws Megavolt on the graphics window
drawLightBulb	Draws a light bulb on the graphics window
drawSwitch	Draws a switch on the graphics window
drawBattery	Draws a battery on the graphics window
drawDetector	Draws a motion detector on the graphics window
drawWire	Draws a wire on the graphics window
drawStation	Draws a station of learning on the graphics window

Method	Purpose
draw	Updates the screen
mouseClicked	Detects mouse clicks (toggling electrical switches and choosing options in menus)
keyPressed	Detects when a key is pressed (controlling the movement of the character)

## List of Variables

Variable	Type	Purpose	Scope
screenID	int	ID of the current screen (animation, maze, station, game of testing, etc.)	Global
switchX	int	x coordinate of an electrical switch	Local
switchY	int	y coordinate of an electrical switch	Local
lightBulbX	int	x coordinate of a light bulb	Local
lightBulbY	int	y coordinate of a light bulb	Local
loadingBarLength	int	length of loading bar	Local
wireStartX	int	x coordinate of start of wire	Local
wireStartY	int	y coordinate of start of wire	Local
wireEndX	int	x coordinate of end of wire	Local
wireEndY	int	y coordinate of end of wire	Local
characterX	int	x coordinate of character	Global

Variable	Type	Purpose	Scope
characterY	int	y coordinate of character	Global
lightColor	color	color of a light bulb	Local
stationX	int	x coordinate of a station	Local
stationY	int	y coordinate of a station	Local
score	int	player's score	Global

## List of Errortraps

---

- Makes sure character does not go off the screen in any of the games
- Makes sure character does not go through the walls of the maze
- Makes sure player completes all of the maze stations before exiting the maze

## List of Colors

---

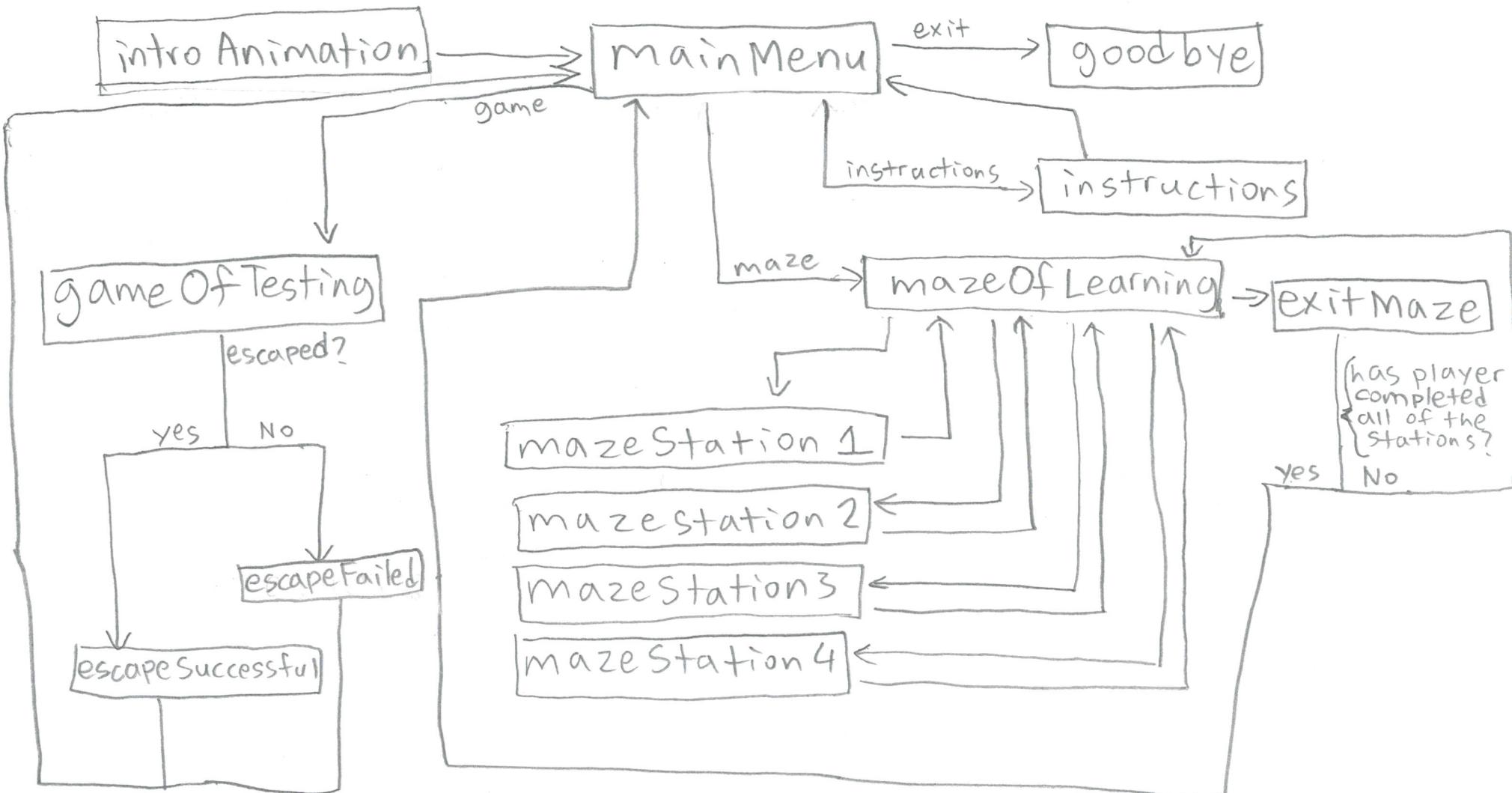
- Orange -> character's hard hat
- Gray -> several different shades, including black
- Red -> lights
- Green -> lights
- Yellow -> lights

## Tools I Will Use

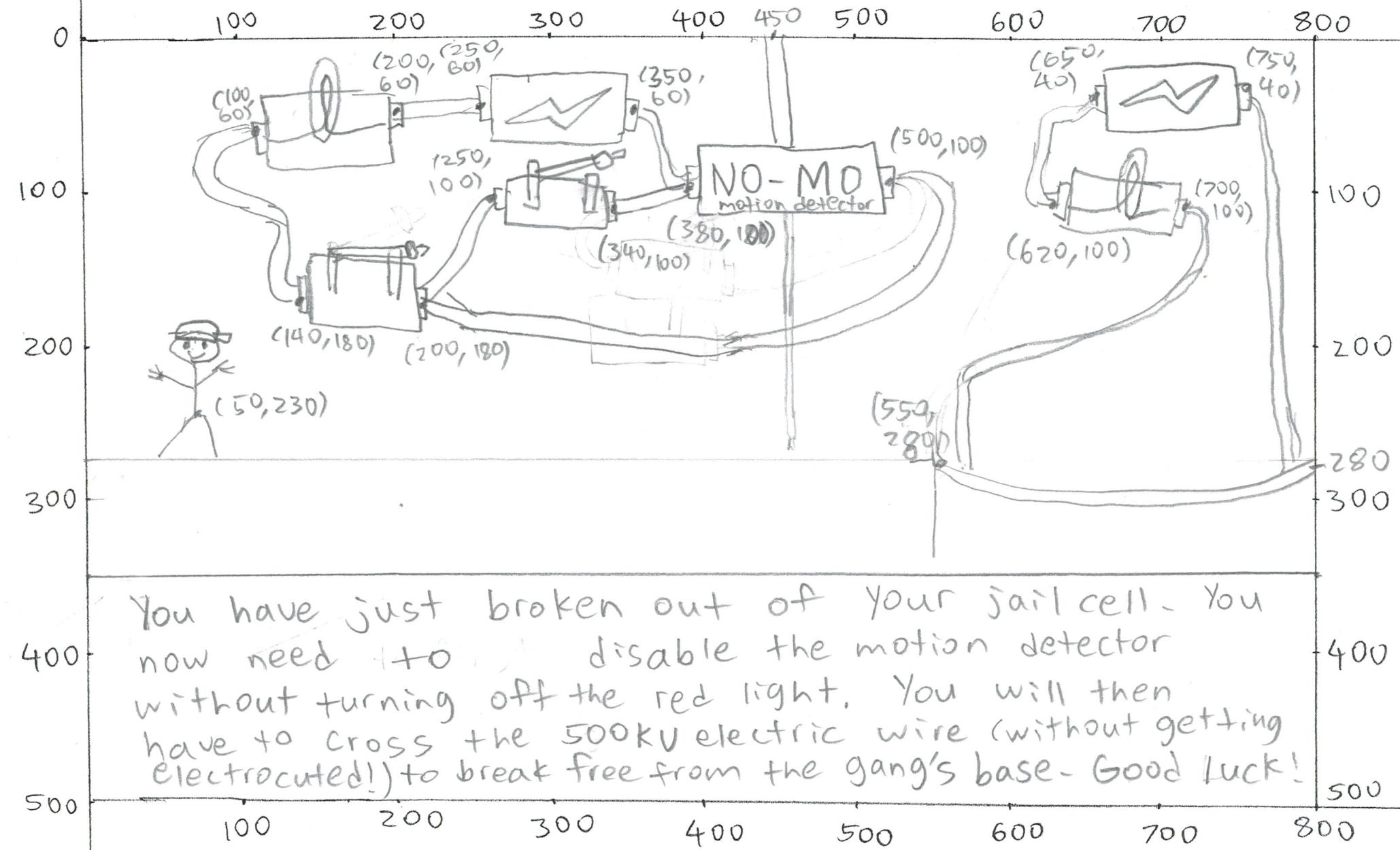
---

- mouseClicked() to process mouse clicks
- keyPressed() to process key presses
- methods such as **drawLightBulb**, **drawSwitch**, and **drawBattery** to draw the different components of a circuit
- specific coordinates assigned to each box in maze grid (0 to 7 for x, 0 to 4 for y)

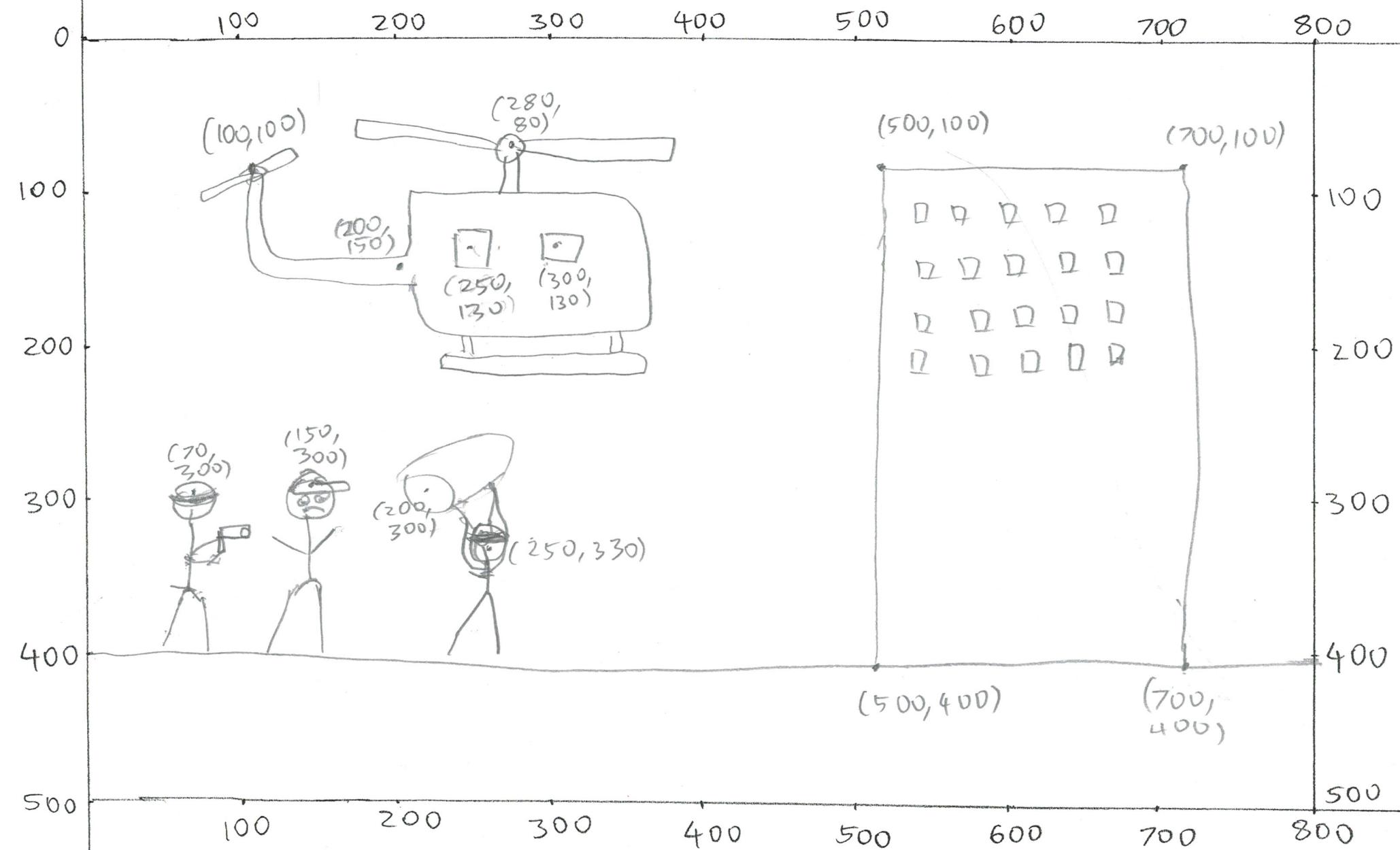
# Program Execution Flowchart



# Game of Testing



# Animation Scene



Goodbye

0 100 200 300 400 500 600 700 800

100  
Thanks For Playing The  
Adventure of Megavolt!

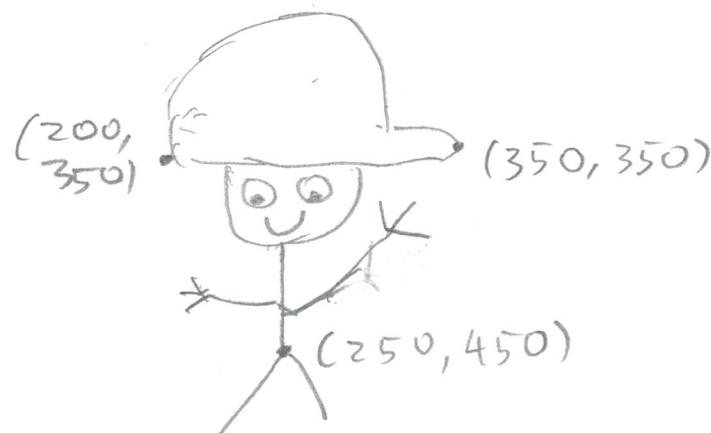
100

200

Made by Peter Ye with Processing

200

300



400

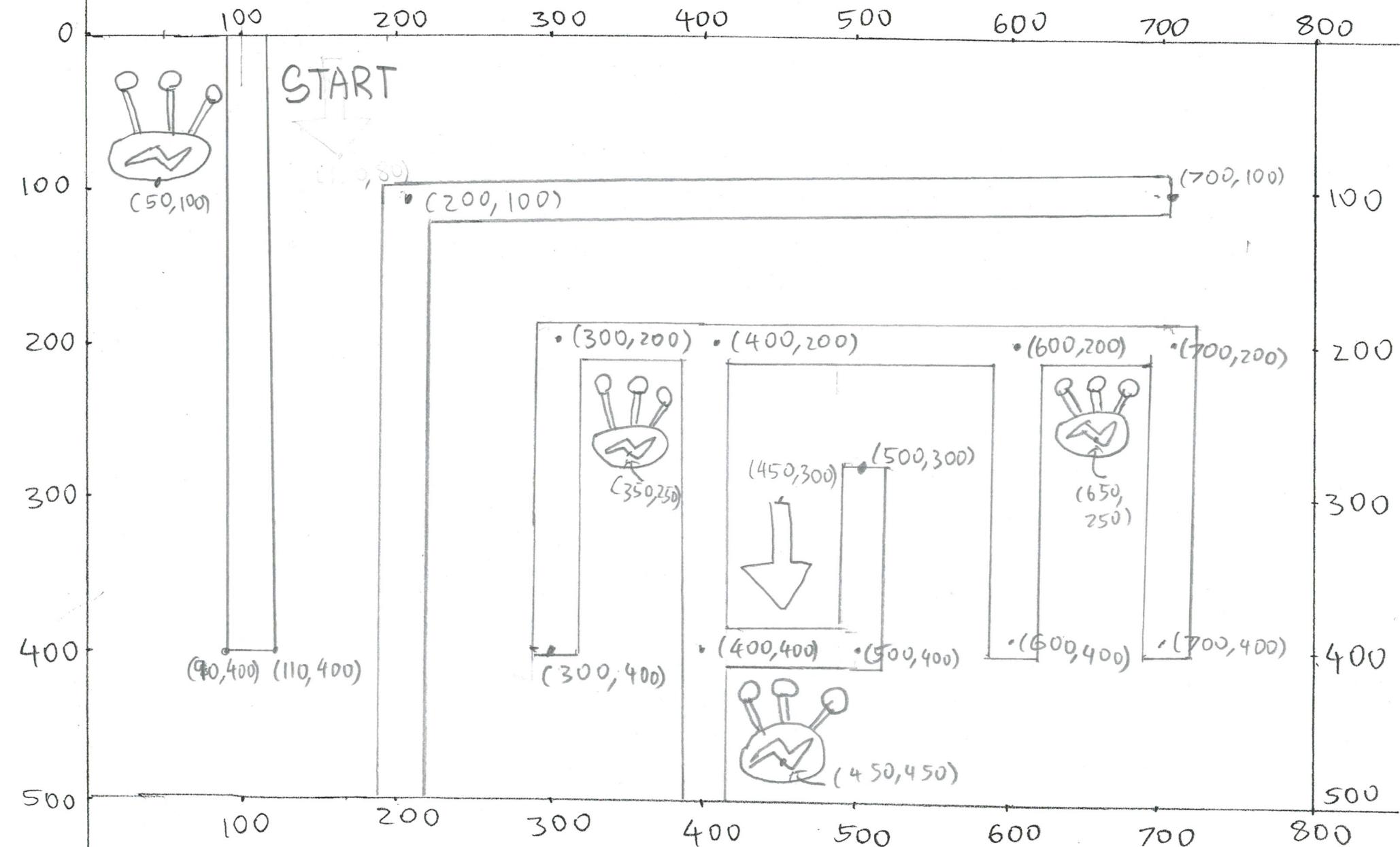
400

500

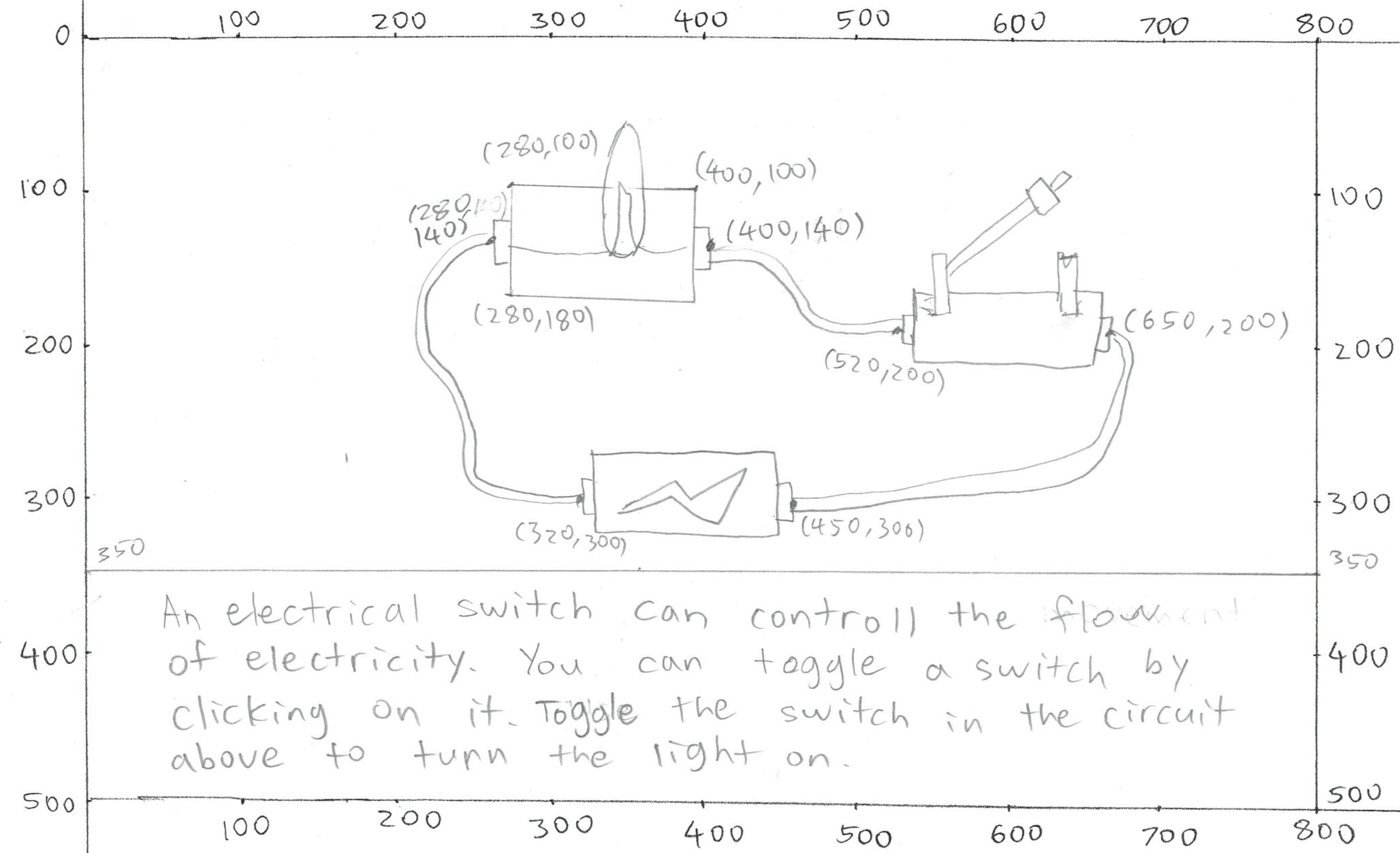
500

100 200 300 400 500 600 700 800

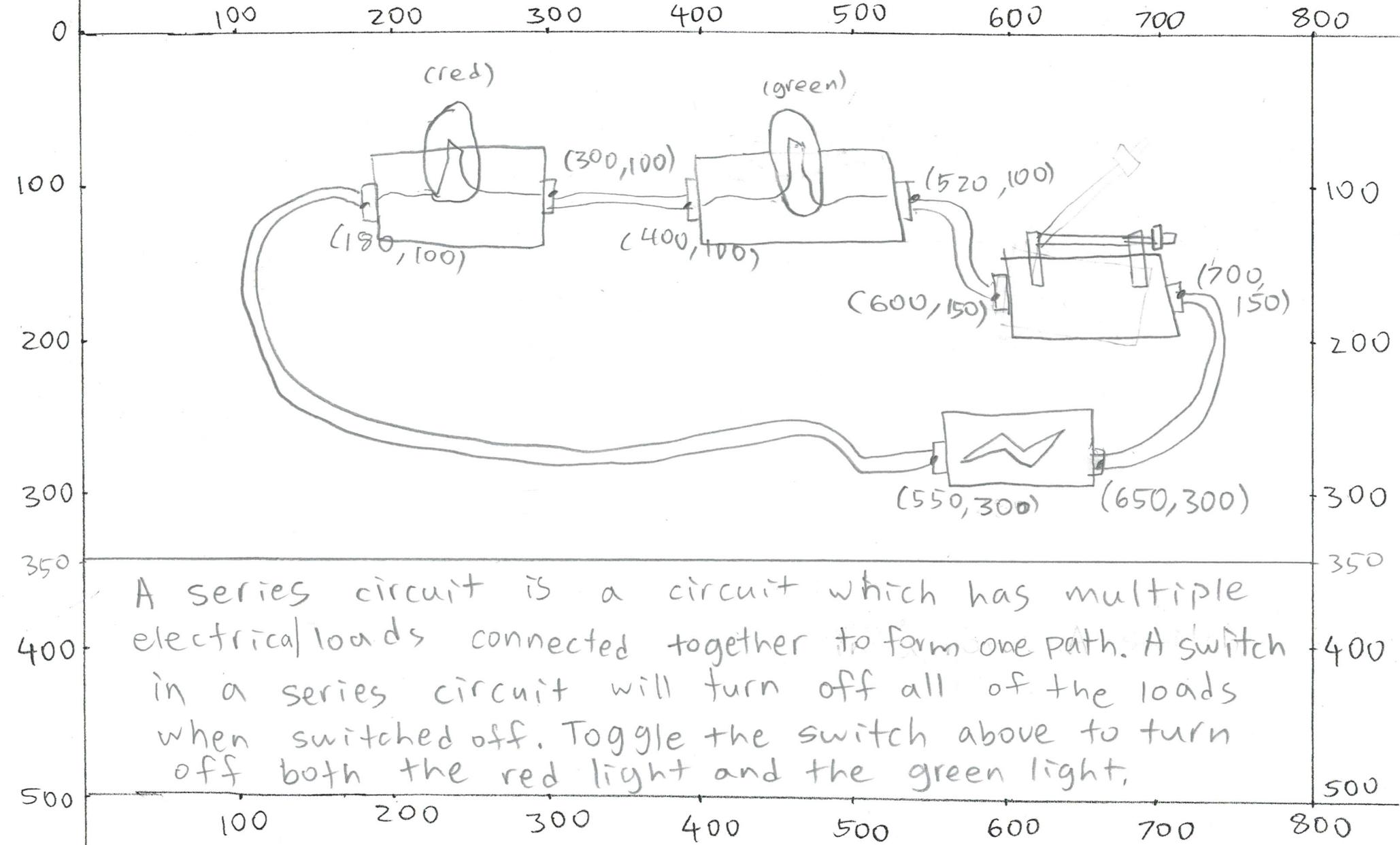
# Maze of Learning



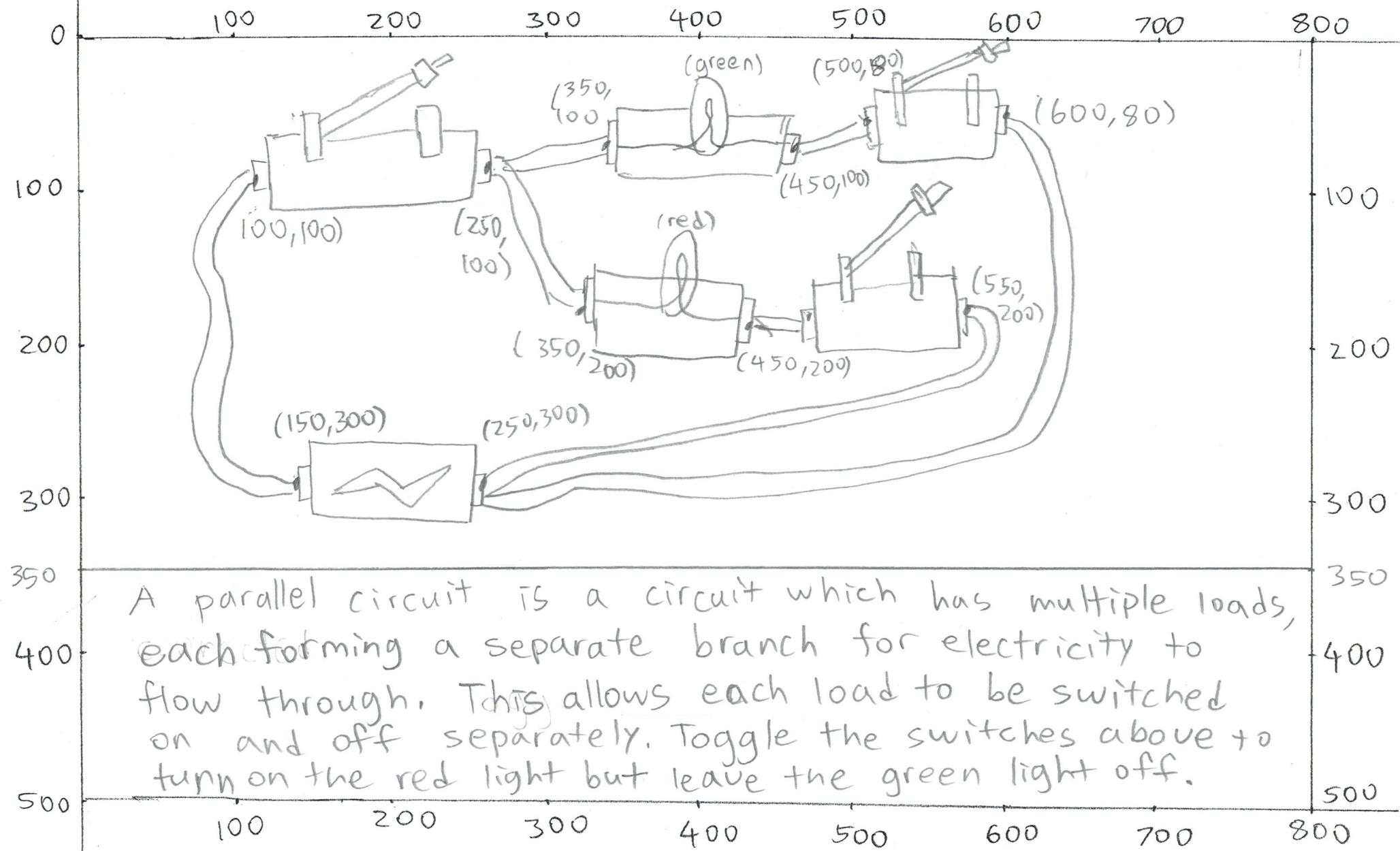
# Maze of Learning Challenge # 1



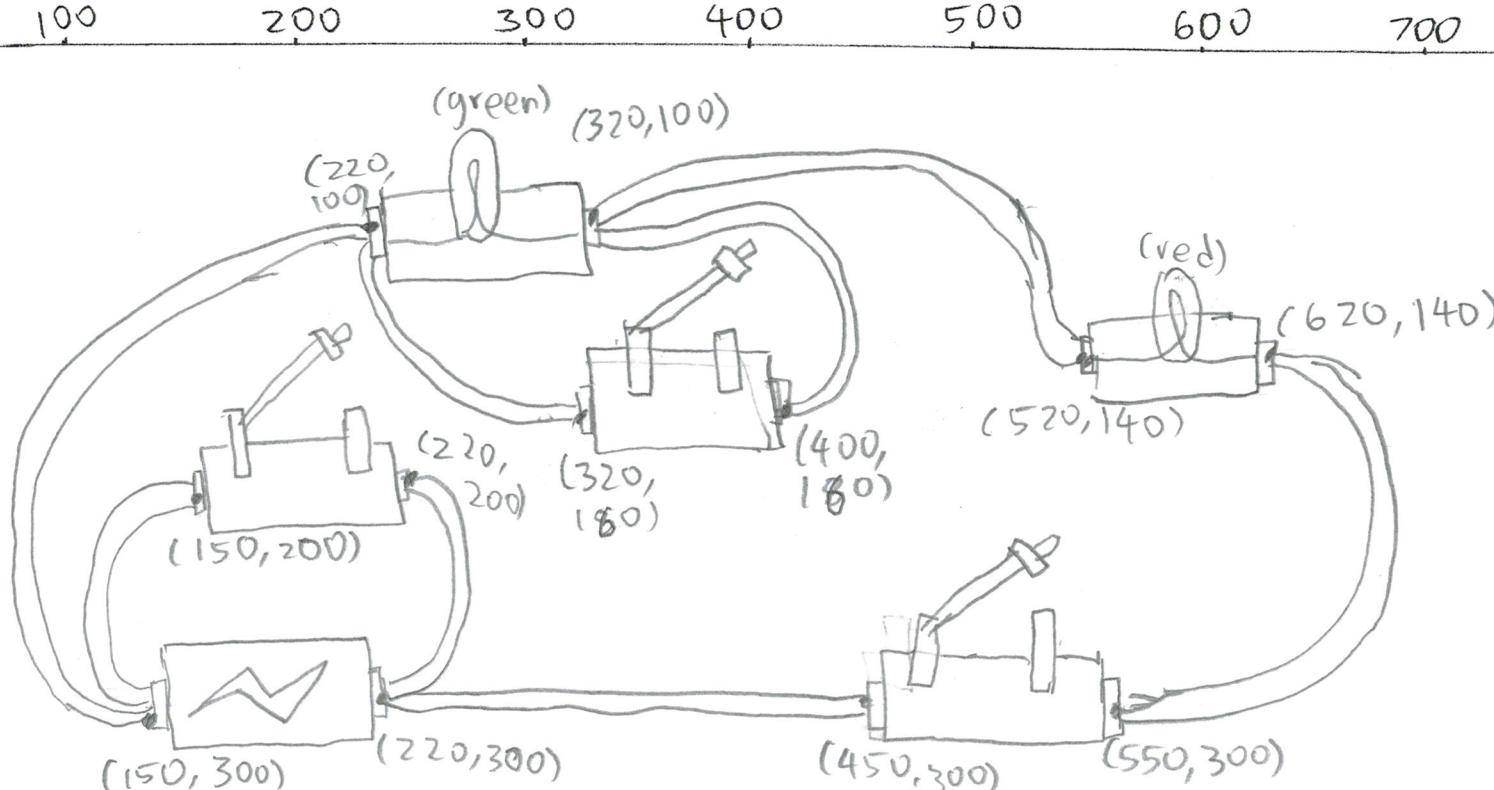
# Maze of Learning Challenge #2



# Maze of Learning challenge #3



# Maze of Learning Challenge #4



Electricity always takes the path of least resistance. A short circuit is a special type of parallel circuit where the two ends of one or more loads are connected by a wire. This directs all of the electricity through the wire, causing none of it to go through the loads. Short circuiting an entire battery is very dangerous. Use a short circuit

to turn off the green light above but leave the red light on (without blowing up the battery).