Super Mario Matlab

During the two lectures of the introduction to Psychophysics Toolbox (PTB) we are going to develop a simple game which will show how to structure the code for creating a task using the most common functions available in PTB.

A close up of a toy

Description automatically generated with medium confidenceA picture containing light

Description automatically generatedA close-up of a toy

Description automatically generated with medium confidenceA picture containing indoor, decorated

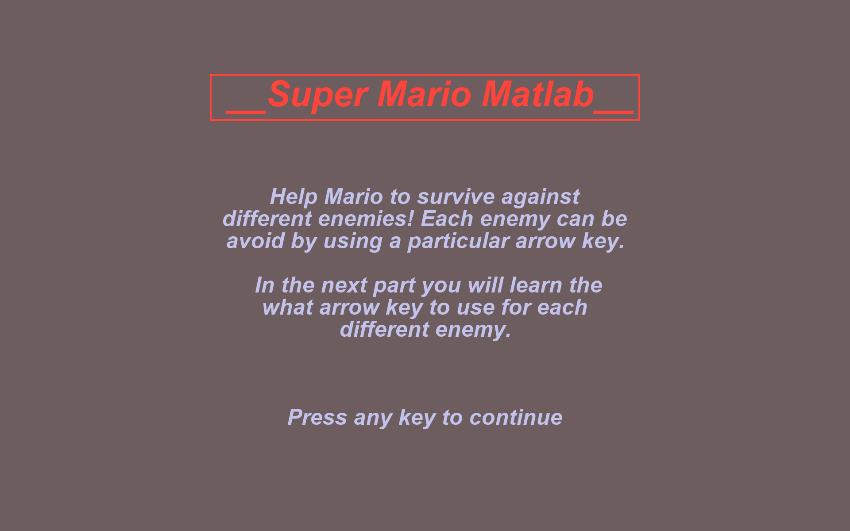
Description automatically generatedThe purpose of the game will be to avoid, like in the Mario franchise games, several different enemies. The enemies we are going to encounter are the following: “Lakitu”, “Goomba”, “Bill” and “Super Hammer”

The purpose of the game is to present each of these enemies, one at a time, to the user and the let the user react to each one of them by pressing a key on the keyboard. Each of the enemy can be “avoided” (consider it as a good trial) by pressing a specific key on the keyboard. The key can be pressed anytime after the enemy is being presented, but the user will have only a specific amount of time to press the correct key. The keys to avoid the enemies are the following: the right arrow key for “Lakitu”, the up-arrow key for “Goomba”, the down arrow key for “Bill” and the left arrow key for “Super Hammer”.

The game is divided in two parts, a tutorial and a testing phase. During the tutorial the user will see a welcome screen for the game, and it will learn the associations between different enemies and arrow keys. During the testing phase the participant will be show with a random number of enemies, each one at a time, and will have the possibility to press a key to avoid the enemy. A proposed solution for the tutorial will be given at the end of the first lecture, while a proposed solution for the testing phase will be given at the end of the second PTB lecture.

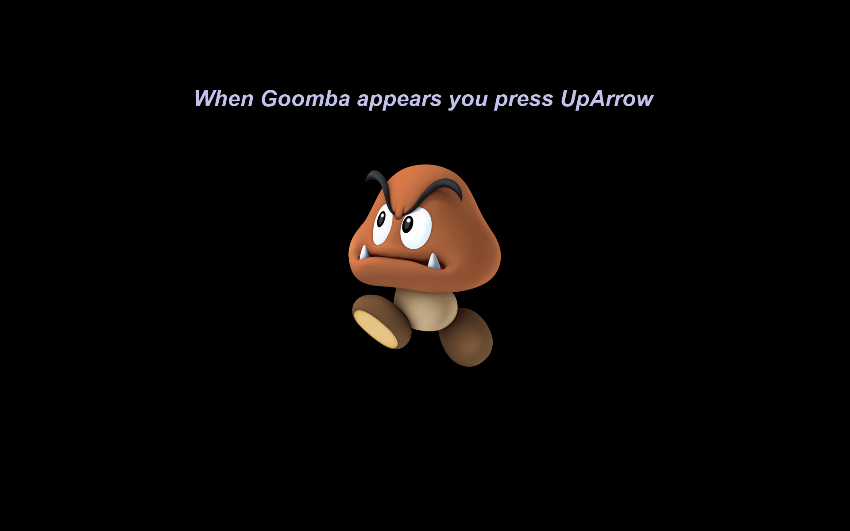
# Tutorial part

In this section a description of the tutorial with some example screenshots will be provided. The tutorial will start after initializing the PTB window.

1. The first screen the user will see is a welcome screen filled with a brief game description. This screen will stay on the PTB window up until the user press any key to acknowledge he has read the instruction and he can proceed further. An example of the screen is provided below.

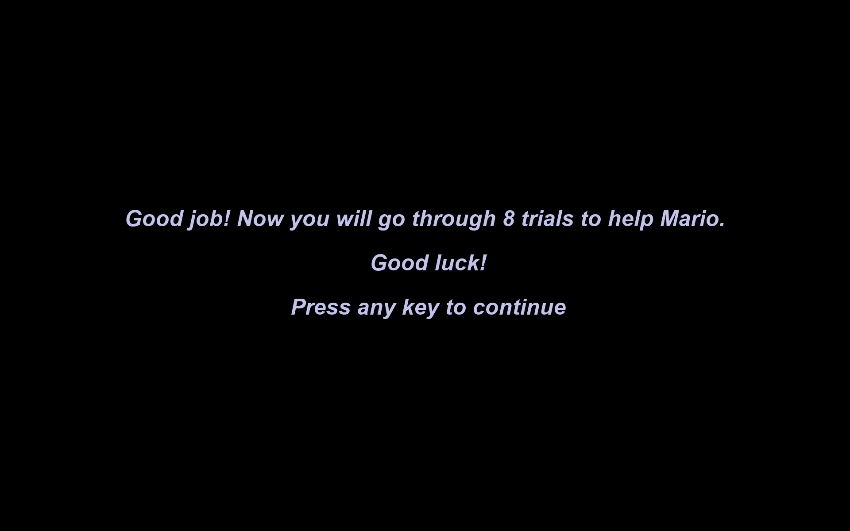
[BONUS] Notice how the title is enclosed into a frame. You can try using the sub-function Screen(‘FrameRect’) to draw it. Type in *Screen FrameRect?* In the command window to see case uses.

1. After the welcome screen, the user will be presented with one of the images from the set. The image will be placed at the center of the screen. A text will be displayed on the upper part of the screen stating what key is expected from the user to press. After presenting the image the screen will stay in place until the user press a key. See the image below as a proposed example.

The process will repeat until all the enemy images will be shown once.

Images you can load should be found into the “Assets” folder contained in the package relative to PTB lecture one.

[BONUS] A user should only be allowed to be presented with the next image only when the correct key has been pressed. Can you think of a way to check for the correct keycode?

1. After the last image of the enemy set has been displayed on the screen, an ending tutorial screen will remind the user that a testing phase is about to start. The screen will remain in place until the user presses any key. See the image below as a proposed example.
2. After step 3, you can safely close the PTB window