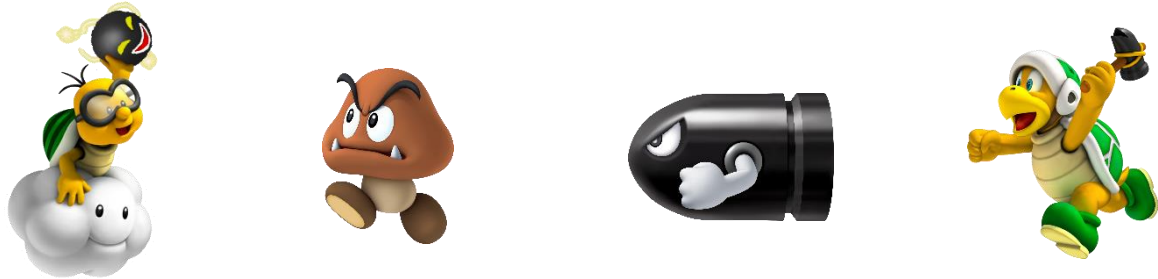


# Super Mario Matlab

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.

The purpose of the game will be to avoid several different enemies, as in the Mario franchise games. The enemies we are going to encounter are: “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 let them react by pressing a key on the keyboard. Each of the enemy can be “avoided” by pressing a specific key. This key can be pressed any time 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 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 participant will be shown a welcome screen for the game and will learn the associations between different enemies and arrow keys. During the testing phase, the participant will be shown a random selection of enemies, one at a time, and can press a key to avoid each enemy. A proposed solution for the tutorial phase will be available after the end of the first lecture, while a proposed solution for the testing phase will be available after the end of the second lecture.

## Testing phase

During the testing phase, participants will be given the chance to respond to presentations of each enemy image by pressing the required key. Each trial should proceed as follows:

1. A fixation cross at the center of the screen indicates the start of the trial
2. The fixation cross disappears after a fixed duration, a random image from the set of four potential enemies is displayed at the center of the screen, and keyboard monitoring begins
3. The image stays in place for a fixed duration, giving the participant a chance to respond by pressing a key
4. The image disappears, and auditory feedback is given depending on whether the participant pressed the correct key or not
5. Repeat from step 1 until all enemy images have been displayed the desired number of times

Here are also some additional requirements that should be implemented:

- Each trial always has the same duration regardless of whether the participant responds or not
- Only the first key press detected after the start of the trial should be recorded - all subsequent key presses should be ignored
- At the end of the testing phase, the screen should be safely closed and all collected data should be saved permanently in the 'Results' folder
- Before starting each trial, the participant should have released all keys on the keyboard
- In case of an error, we should be able to save the data collected up to that point

Audio files for auditory feedback can be found in the 'Assets' folder - 'pos\_feedback' should be used for positive feedback and 'neg\_feedback' for negative feedback

When structuring your code, it would be nice to be able to quickly change some values at the start of the script, including:

1. the duration for which the fixation cross will be displayed
2. the number of times each enemy will be presented
3. the duration for which each image will be displayed
4. the scale of each image
5. the orientation of each image

For each trial, you should also collect the name of the image being presented and the correct response key associated with it, the key the participant pressed (if any), reaction time, and a logical index indicating whether the response was correct or not. Below you can find a proposed structure for the collected data (in this example each image was presented twice):

1 Trial	2 Stimulus	3 Key	4 KeyPress	5 RT	6 Correct
1	'Bill'	'DownArrow'	'UpArrow'	0.7166	0
2	'Lakitu'	'RightArrow'	'DownArrow'	0.8064	0
3	'Super Ham...	'LeftArrow'	'LeftArrow'	1.4550	1
4	'Lakitu'	'RightArrow'	'RightArrow'	0.8764	1
5	'Super Ham...	'LeftArrow'	'LeftArrow'	0.6252	1
6	'Bill'	'DownArrow'	'DownArrow'	0.7063	1
7	'Goomba'	'UpArrow'	'UpArrow'	0.2807	1
8	'Goomba'	'UpArrow'	'UpArrow'	0.6393	1