

# Ball Sort Puzzle

Reinforcement Learning

Artificial Intelligence

Master in Informatics and Computing Engineering



Gonalo Alves – up201806451  
Gustavo Mendes – up201806078  
Pedro Seixas – up201806227

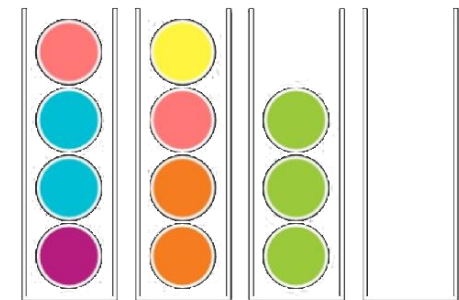
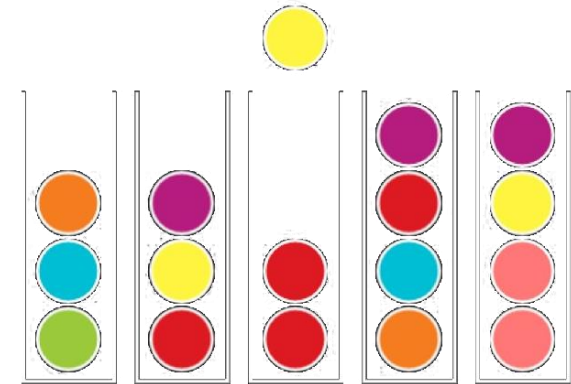
# Formulation of the problem as an ML problem

The objective of the Ball Sort Puzzle is to sort the colored balls in the tubes until all balls with the same color stay in the same tube.

The player can only move a ball at the top of a tube to:

- an empty tube;
- another tube that has a ball with the same color on top and has enough space;

The player can also undo his moves if he finds himself without moves.



# Formulation of the problem as an ML problem

## State Representation:

- Ball: [1..n] (n = the number of colors)
- Tube: [Ball, Ball, Ball, Ball] or []
- Game: [Tube, Tube, ...]

## Actions:

- Move(X,Y): move Ball from Tube X to Tube Y

## Policies:

- Q-Learning
- SARSA
- Monte Carlo

## Rewards:

Three ways to assign a reward based on verifications:

- Verify if the game is finished (+2) or the tube that received a ball is completed (+1)
- Verify if the game is invalid (-5) or finished (+20)
- Verify if the game is invalid (-1000), finished (+50) or count the number of consecutive balls with the same colour

# Related Work

Mainly 2D games:

- [Retro](#)
- [Slime Volley Gym](#)
- [Demon Attack](#)

On OpenAI:

- [Spaces](#)
- [Environments](#)

On Policies:

- [Q-Learning](#)
- [Jupyter Notebook about Q-Learning](#)
- [SARSA](#)

# Description of the tools

**Jupyter Notebook** – Allows for interactive computing

**OpenAI gym** - Allows for modeling of a reinforced learning environment

**Algorithms** - Since the action and the observation space of the puzzle are both discrete, the policies to be implemented are: Monte Carlo, Q-Learning and SARSA

# Implementation work already carried out

**Programming language:** [Python](#), [OpenAI](#), [Jupyter Notebook](#) (with [Anaconda](#))

**Development environment:** [VSCode](#)/[IntelliJ](#)

**Data Structures:**

- Lists, for representing the Tubes and the Game

**File Structure:** The Project's Repository is available at [Github](#)

**Implemented Work:** Modeled Environment and Agent, Q-Learning Policy