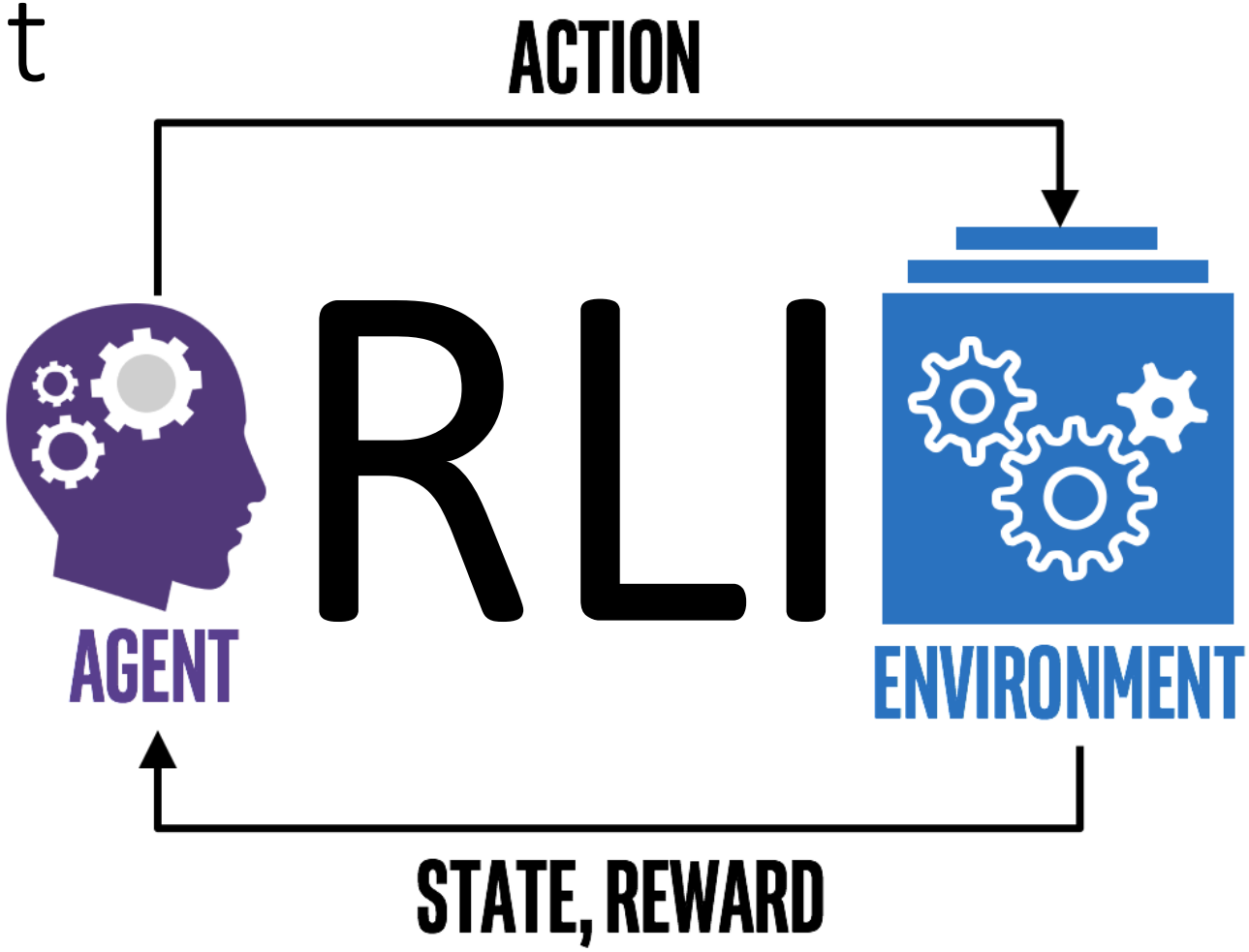
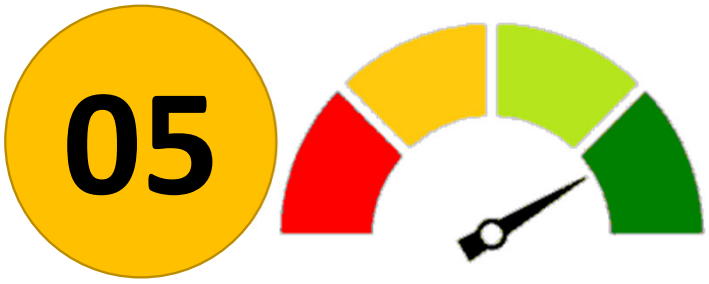
# Reinforcement Learning Introduction




José Manuel Rey, IE

# Reinforcement Learning Introduction Assignment 05.00

Date: Jan/18 2023

Due Date: Jan/25 2023 [SESSION 06]

Ongoing grading (maximum)

30 Individual Assignment [mini-groups]

# Assignment Description Overview

This assignment consists of 4 parts/questions:

**Assignment 05.00.01 → "Dice and chupitos" [12 points]**

**Assignment 05.00.02 → "Spin until you win" [9 points]**

**Assignment 05.00.03 → "Cat and Mouse Hide and Seek (MC, probability)" [6 points]**

**Assignment 05.00.04 → "Cat and Mouse Hide and Seek (MDP, strategy)" [3 points]**

For each part/question you should provide a self-contained ".ipynb" file, providing a clear concise explanation of the conceptual approach, the diagrams or models representing the problem, and the derivation of the solution (whatever it could be the path you chose: analytical, algebraic, procedural or algorithmic)

The solution/answer itself "alone", will not be considered valid enough

(*) Note: to help you with the "diagrams" part, an additional helper tool (markovchains.py) is provided, but feel free to use any other one of your choice if it can help you better to organize and present you approach and ideas

**SUBMIT YOUR WORK BY ZIPPING TOGETHER THE 4 NOTEBOOKS (COMPLETED WITH YOUR ANSWERS) IN A SINGLE SUBMISSION FILE NAMED:**

**"RLI_05_00 – mini-group number.zip" (see the attached .txt for the mini-groups)**

# Administrative and additional notes

- Find attached (in the zip file) the configuration of the mini-groups for the assignment:

  → RLI_2022-23 - ASSIGNMENT 0 RLI_05_00 TEAMS.txt

  Only submit ONE zip file per group. Choose ONE member of each group for that submission. In case you send more than one, ensure that all the submissions by the different members are identical, as the system will choose just ONLY one of them (randomly) for the review and evaluation of the whole group.

- Try to use the practice example "MarkovChainDraw-Example (dice 6-number sequence).ipynb" reviewed during the sessions as a template, reference or guideline for the structure and approach of your work and submissions. [find it attached additionally in this assignment zip file]
- As usual, the code should run without warnings or errors. All additional required files should be included in the ZIP file, and if you were using any "novel" additional external library for your work, clearly indicate a brief description of its purpose, the version used and the **"`!pip install <package>`"** required for its installation

# Dice and “chupitos”

**Question**: How long (expected average of throws) will the game last and how many "chupitos" would you have drunk?

Start with 6 empty glasses → Start game

Roll the dice…

DICE = 2
You drink it…
and continue

DICE = 5
You fill it…
and continue

Repeat until all glassed are filled → End game

# A simple example: Spin until you win…

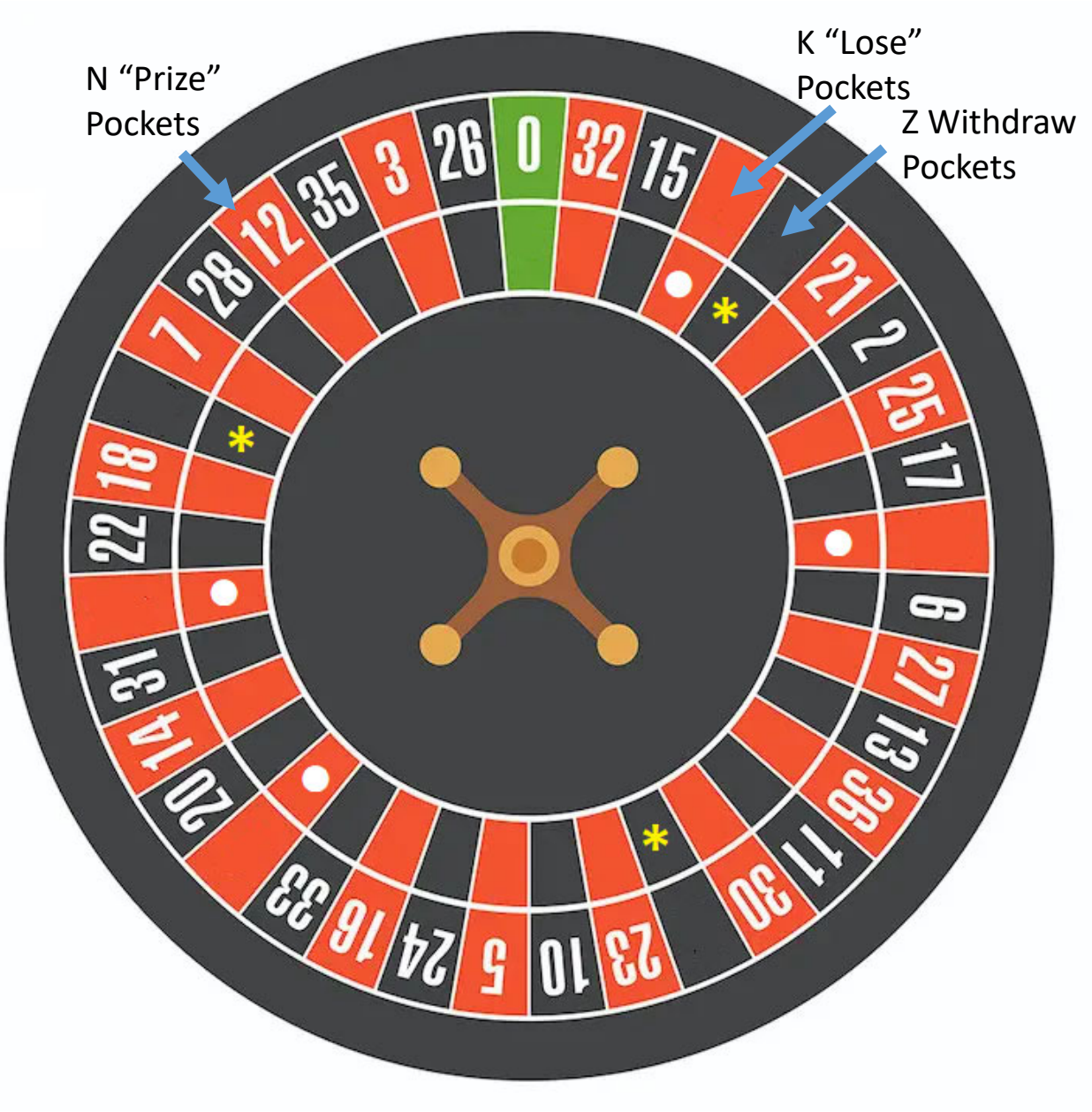
## (Recurrence)

Suppose a roulette game with ***N*** cash prize pockets of various amounts (whose total sum is ***TR*** = Total Rewards) and with ***K*** "lose all" (and end) pockets and ***Z*** "end game" (and withdraw winnings) pockets.

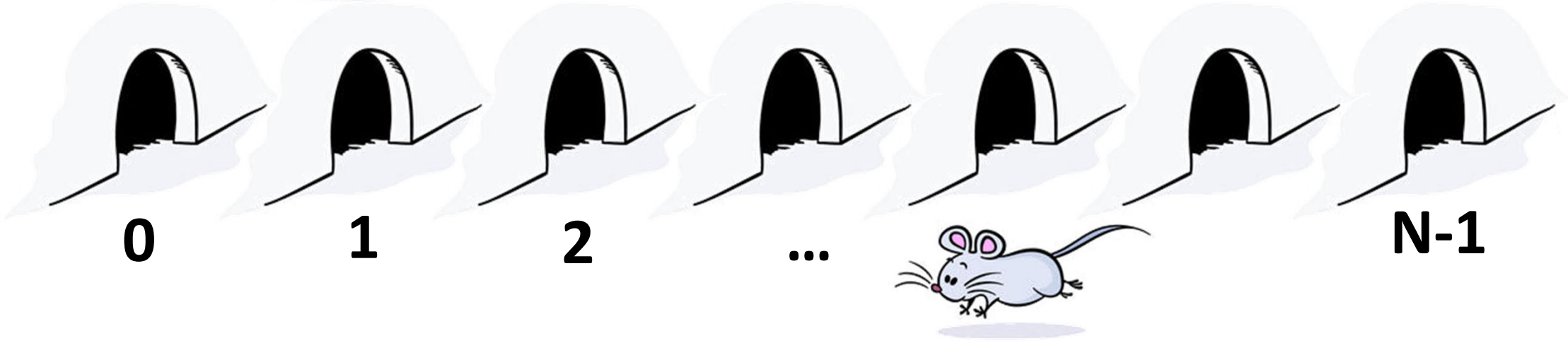
You spin the wheel, bank money and eventually spin again until game over, when you can cash out the banked amount (Z pockets) or simply finish empty (K pockets)

**What would be the mathematical expectation of winnings? i.e. How much would you win, on average?**

**How long would be the average "episode" (number of spins until game over)?**



# Cat and Mouse Hide and Seek (MC's, Probability)



# Cat and Mouse – Hide and Seek (MC's, Probability)

- A mouse is hiding in one of N holes (let's assume N = 7 and label them from 0 to 6) in an old abandoned mansion.
- Each night the mouse moves randomly to an adjacent hole, exactly one number away,
- A cat plans to catch the mouse and plans to sit down at a given hole (let's say, number = k) and hunt in that hole every morning.
- **Question:** For how many days (on average) will it have to stay at a given hole ("k") until it may catch the mouse?
- Note (1): solve the problem for N=7
- Note (2): As an optional additional exercise, you could also think about the problem in the case of a "circular" arrangement of the holes (i.e.: the mouse could also move from hole 0 to N-1 back and forth)

# Cat and Mouse Hide and Seek (MDP's, Strategy)

# Cat and Mouse – Hide and Seek (MDP's, Strategy)

- A mouse is hiding in one of N holes (let's assume N = 7 and label them from 0 to 6) in an old abandoned mansion.
- Each night the mouse moves randomly to an adjacent hole, exactly one number away,
- A cat plans to catch the mouse ("START"). For that purpose it chooses an initial hole to hunt in, and if not successful (not "END", terminal state and reward), each morning may decide whether to "Stay" (repeat in the same hole) or move to the hole on the "Left" or on the "Right" to keep trying to find its prey next morning.
- Define and model:
  - The Transition probability matrix for the position of the mouse
  - The matrix of probabilities of success/failure (conditioned to State)
  - The basic MDP diagram for a generic State
  - The whole MDP diagram of the problem
- IMPORTANT NOTE: It's not necessary to solve the problem or to design either a finite or an optimal strategy for the problem; just formulate the MDP model diagram for it (graph of Sates, Actions and probabilities)