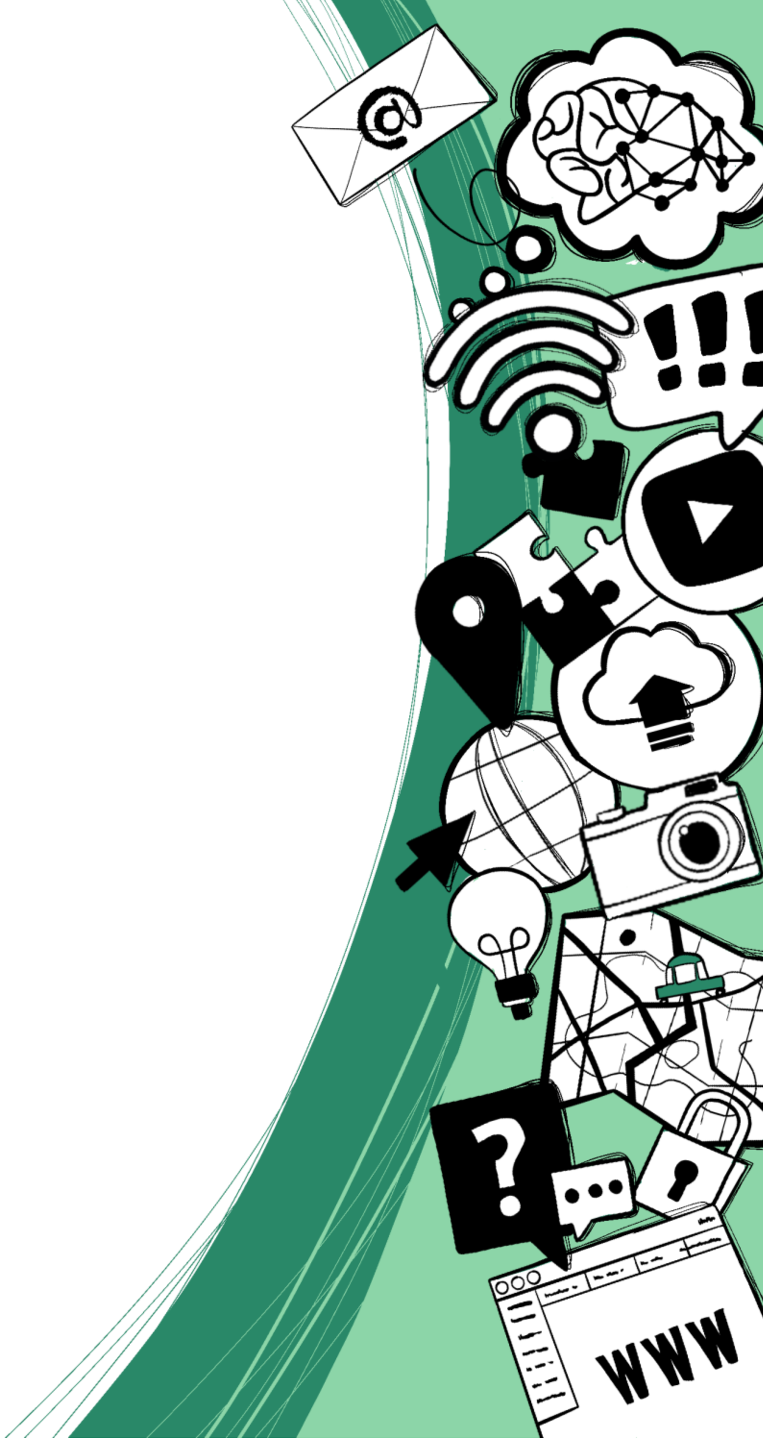
ENARIS



Maze

mobilis
interaktív kiállítási központ

TU Graz

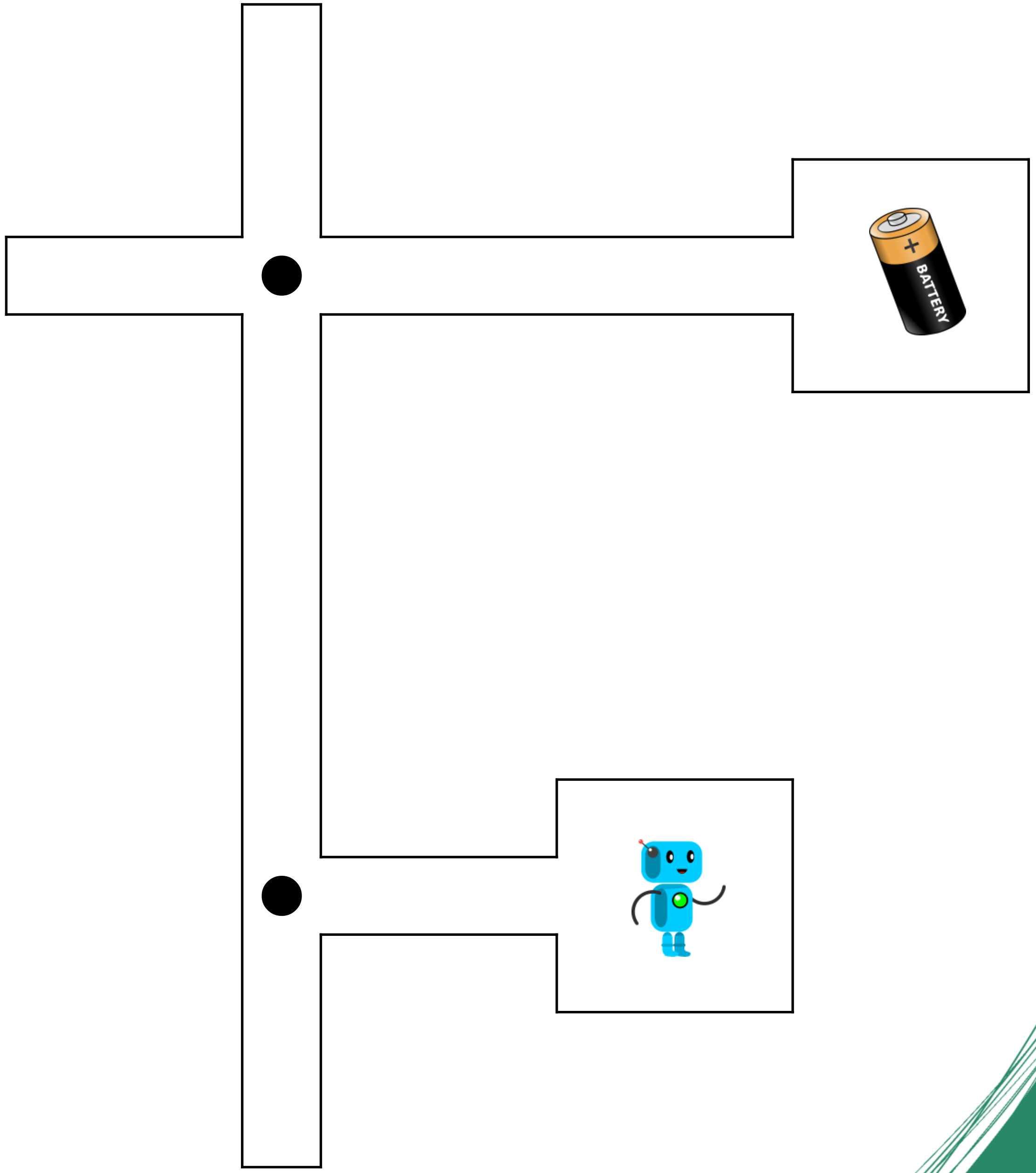
OESTERREICHISCHE COMPUTER GESELLSCHAFT ®
AUSTRIAN COMPUTER SOCIETY

Interreg
Austria-Hungary 2014-2020
European Union – European Regional Development Fund

Core Rules

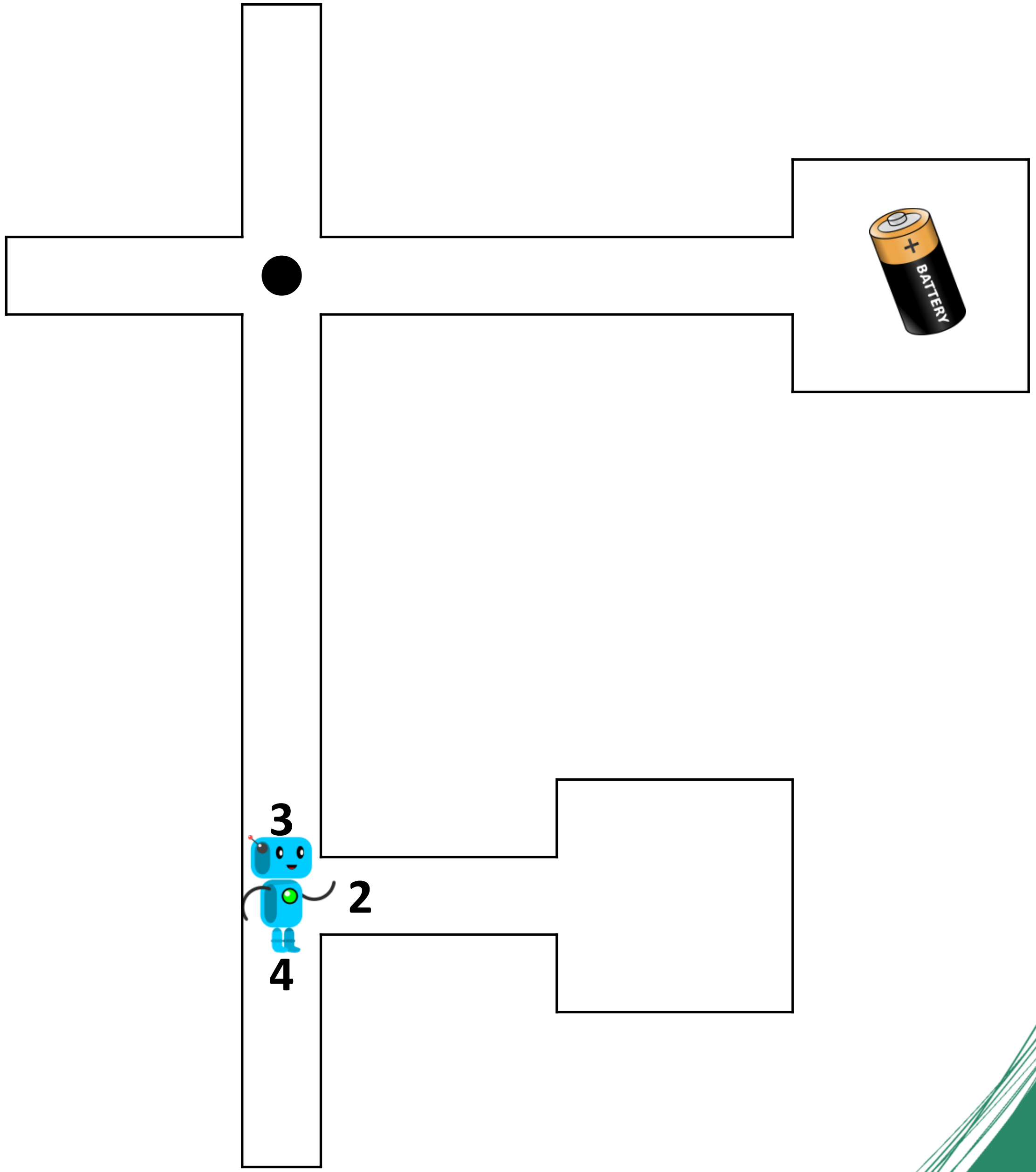
The Game

- **Agent**: Robot
- **Actions**: Take adjacent path
- **State**: Maze, robot Position, Q-values
- **Rewards**: Numbers written on pathways
- **Goal**: reach battery reliably



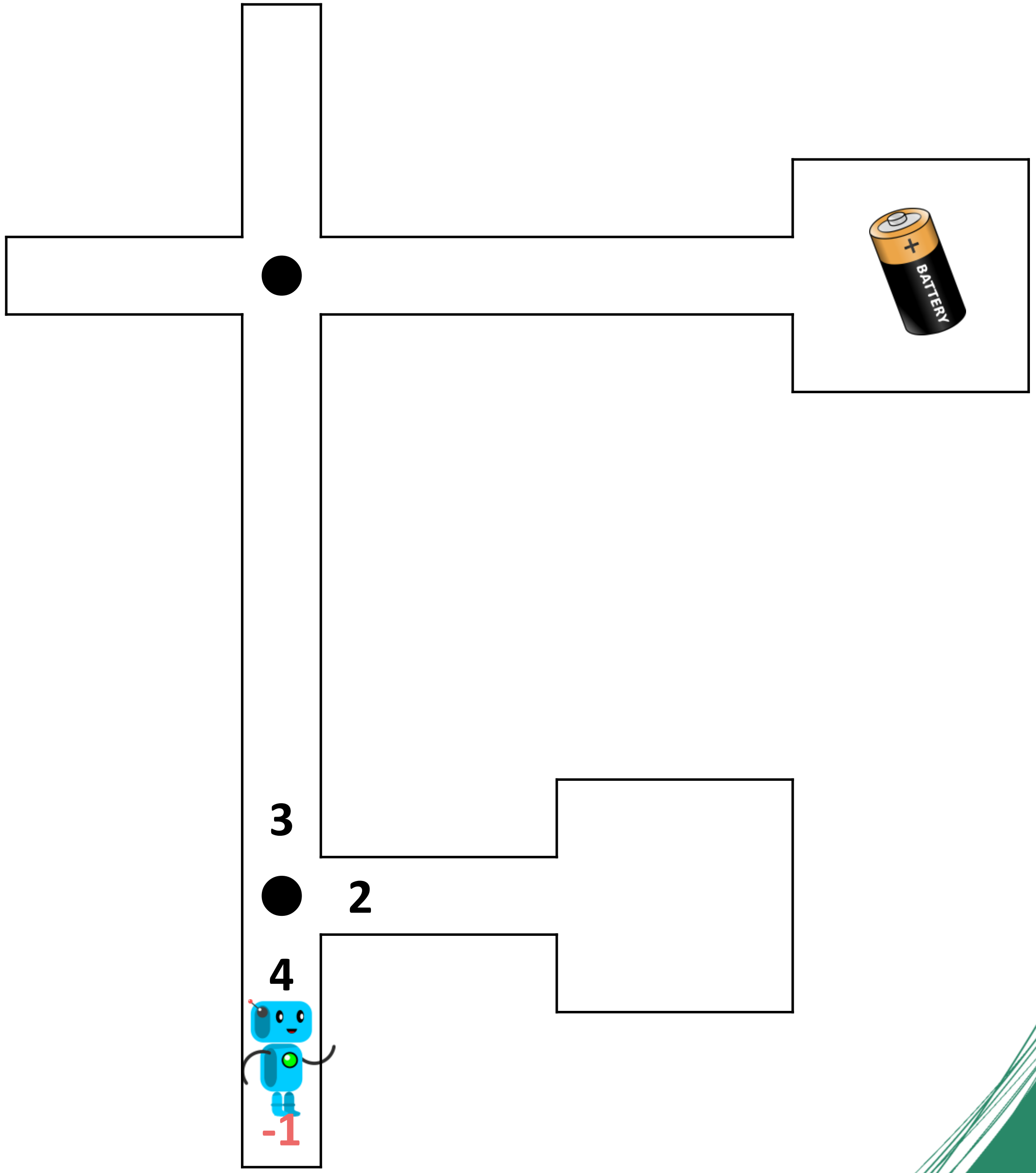
The Game

- Whenever the robot reaches a **new intersection**, it writes a **random number** (use **dice**) on **each path**
- The robot then takes the path with the **highest number**



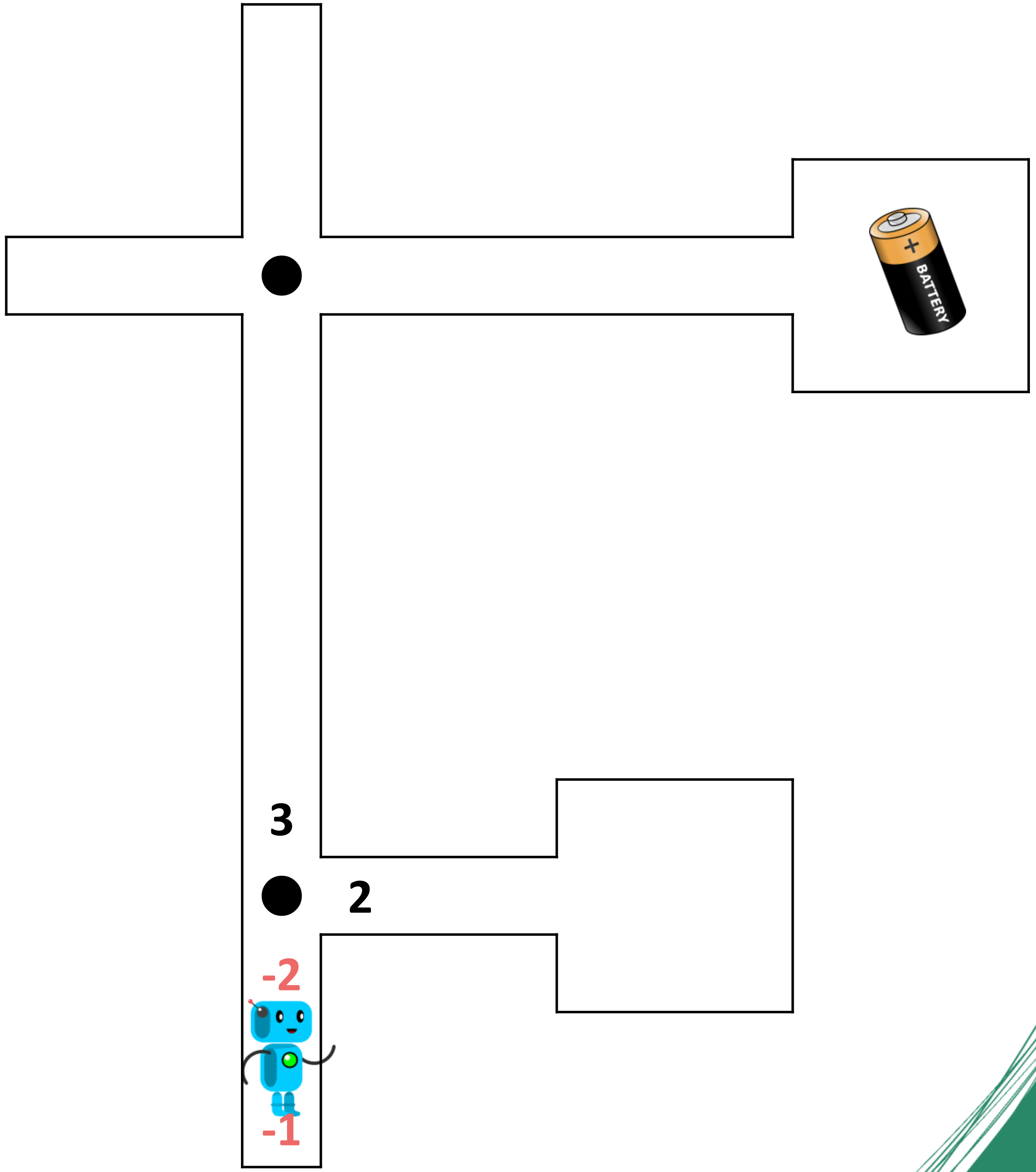
The Game

- Whenever the robot reaches a **dead end**, it writes –1



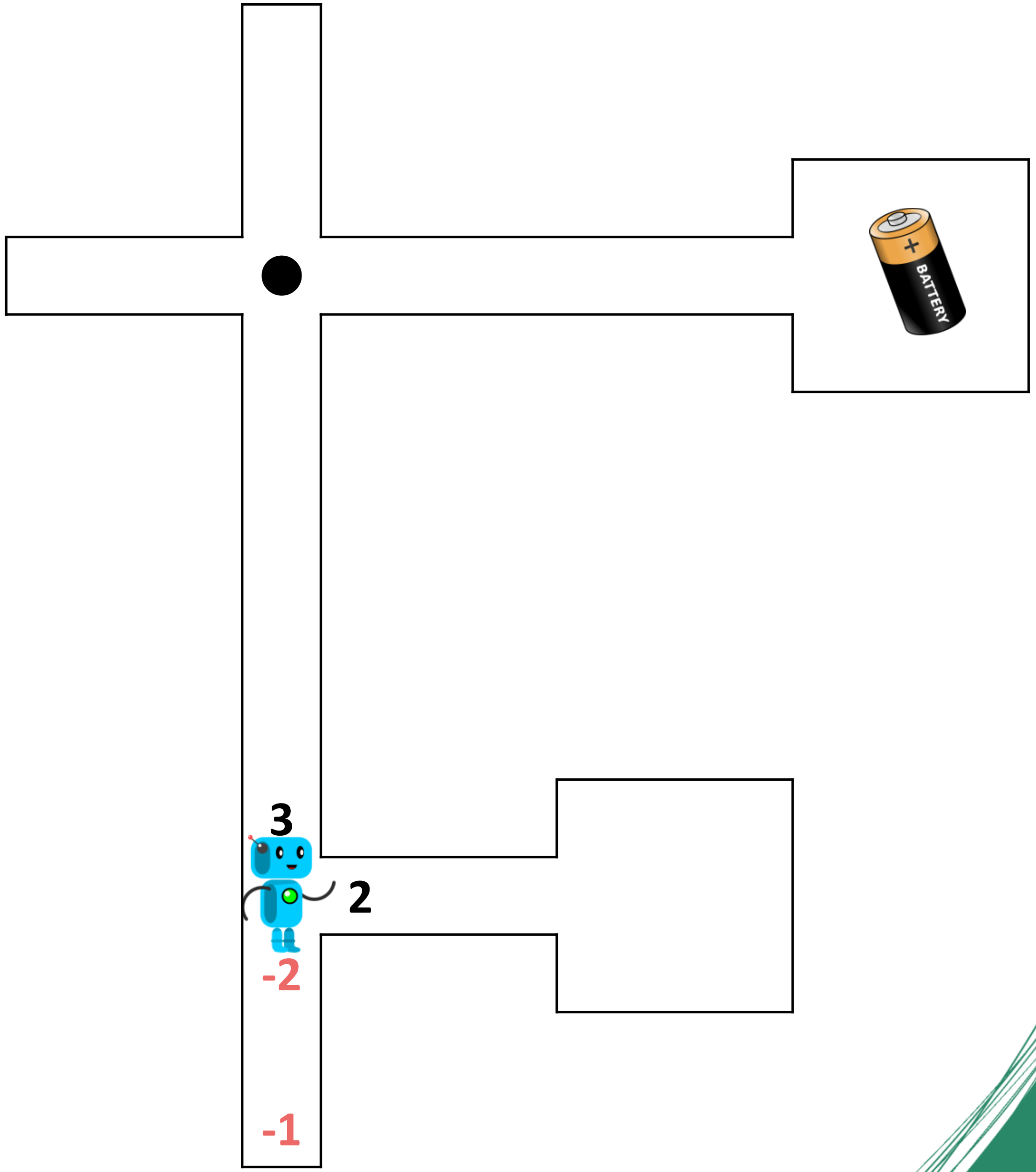
The Game

- Whenever the robot reaches a **dead end**, it writes **-1**
- Then the robot **updates** the **number** of the **path where it came from** to the new **highest value** (-1) **minus one** (-1-1=-2)



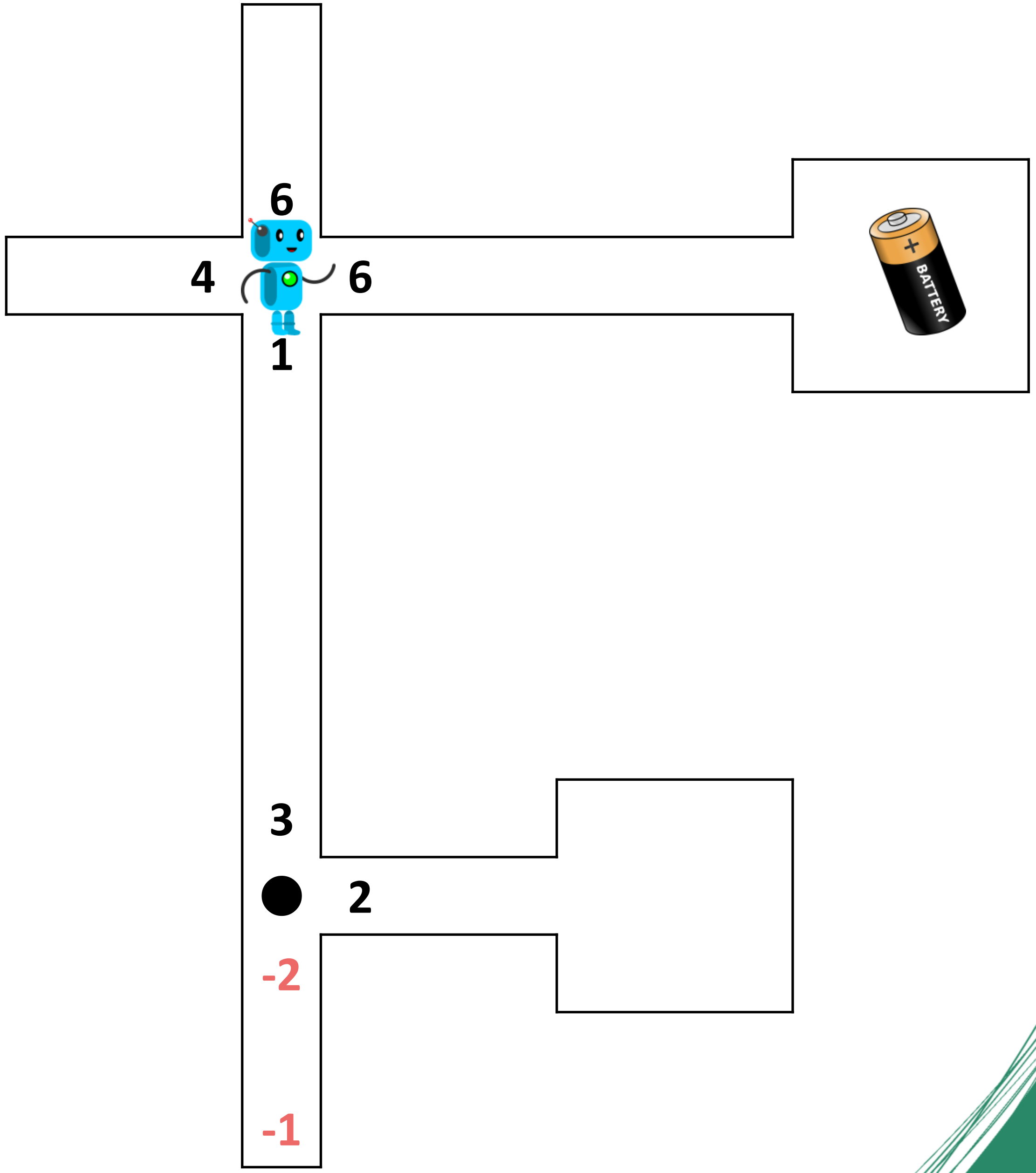
The Game

- In case of a **dead end**, the robot then returns to the **previous crossing** and continues by choosing the highest number



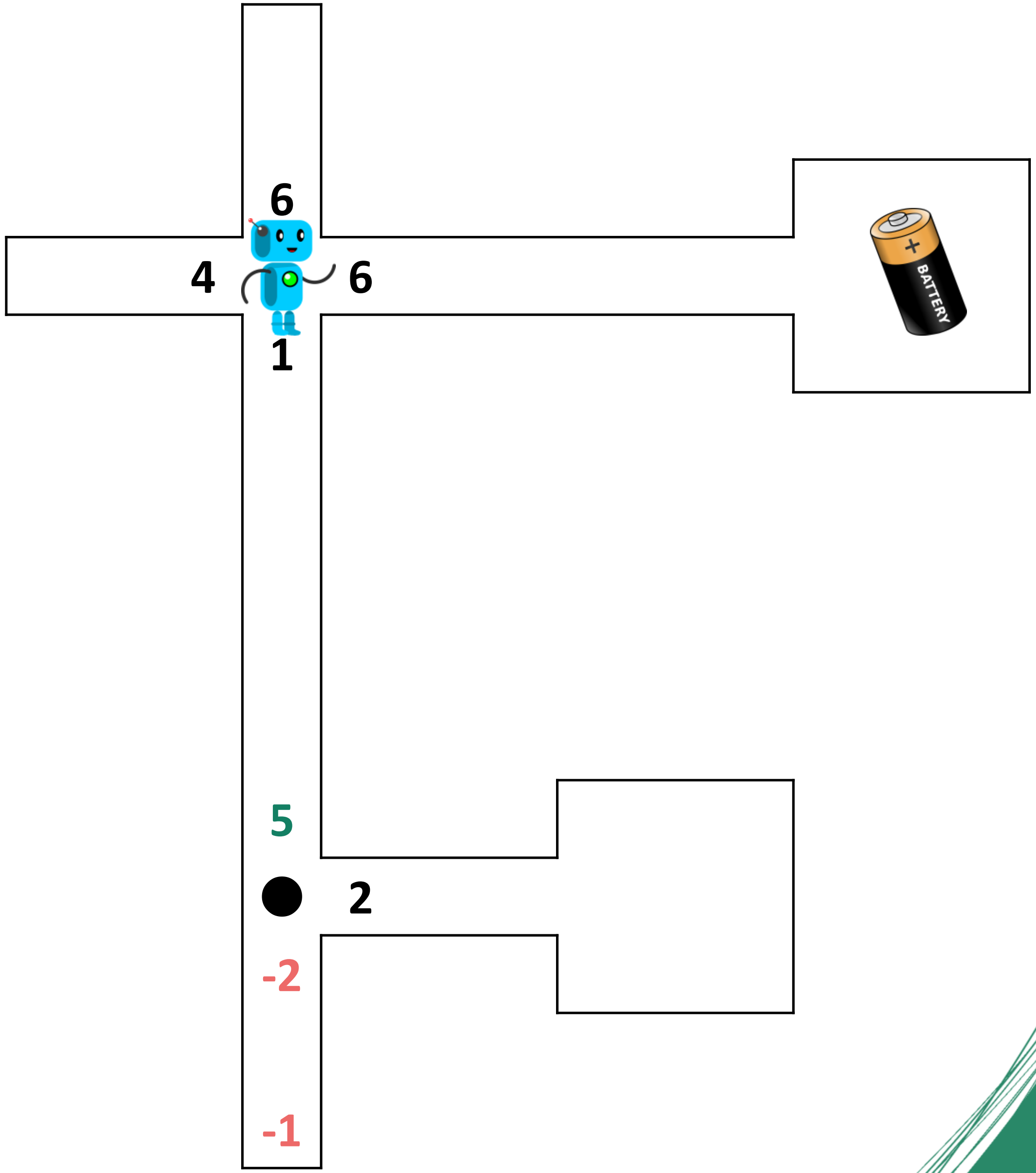
The Game

- **New crossing ->** random numbers



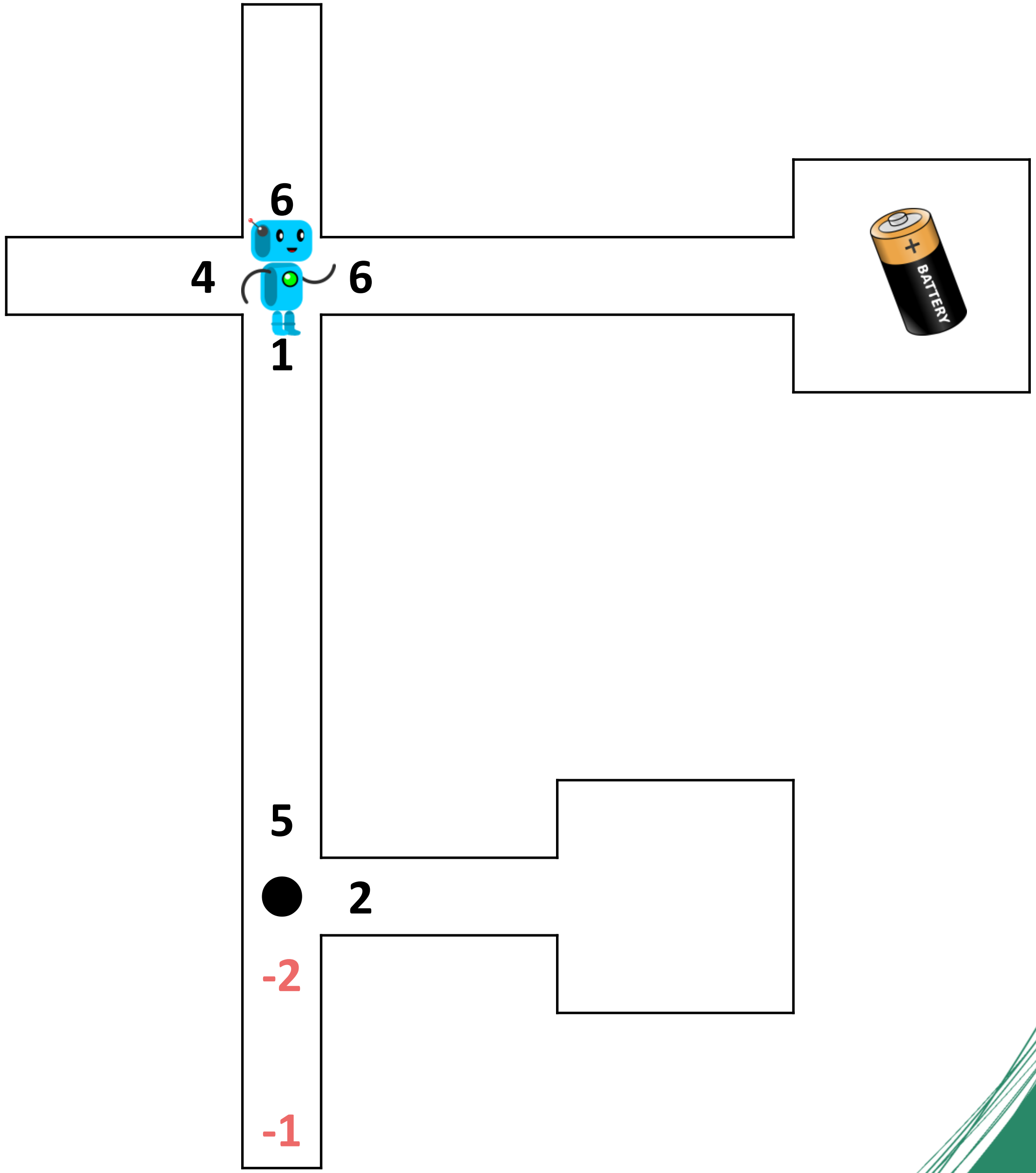
The Game

- **New crossing** -> random numbers
- **Update** pathway to new highest (6) minus one (6-1=5)



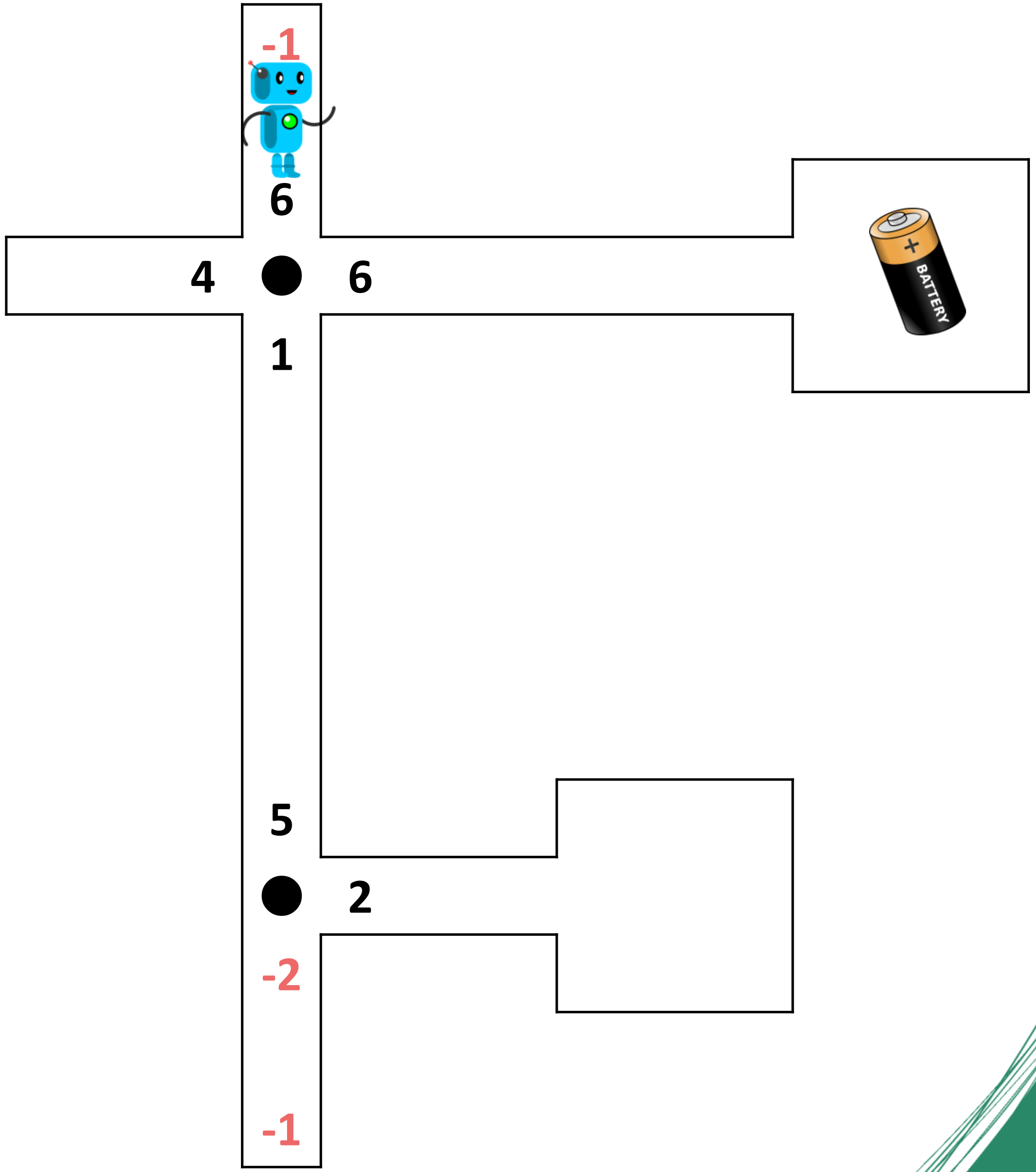
The Game

- **New crossing** -> random numbers
- **Update** pathway to new highest (6) minus one (6-1=5)
- If there are **two or more** biggest **numbers**, the robot choses a **random one** (in this case: up)



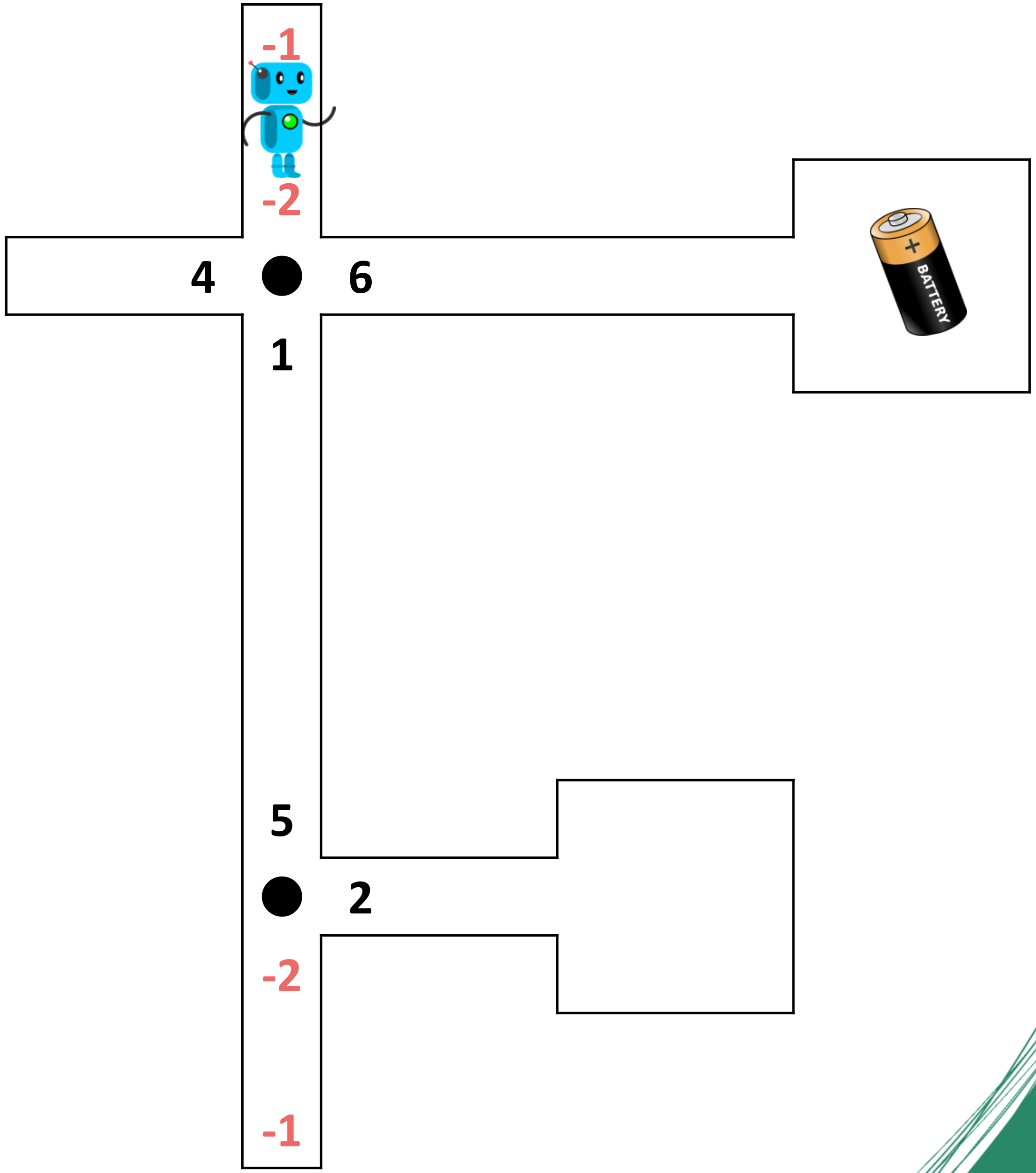
The Game

- The robot **continues** until it **reaches** it's **goal**



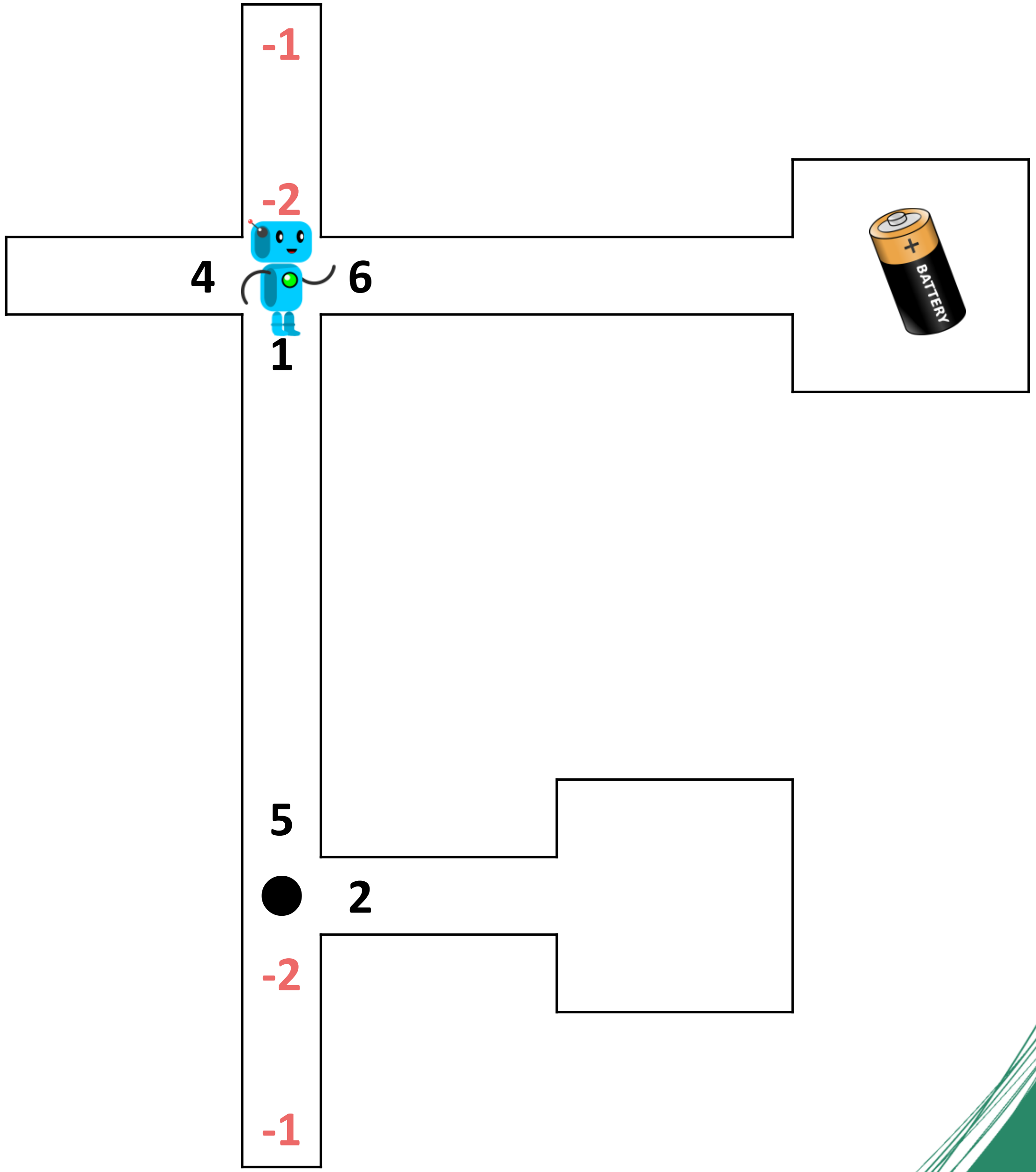
The Game

- The robot **continues** until it **reaches** it's **goal**



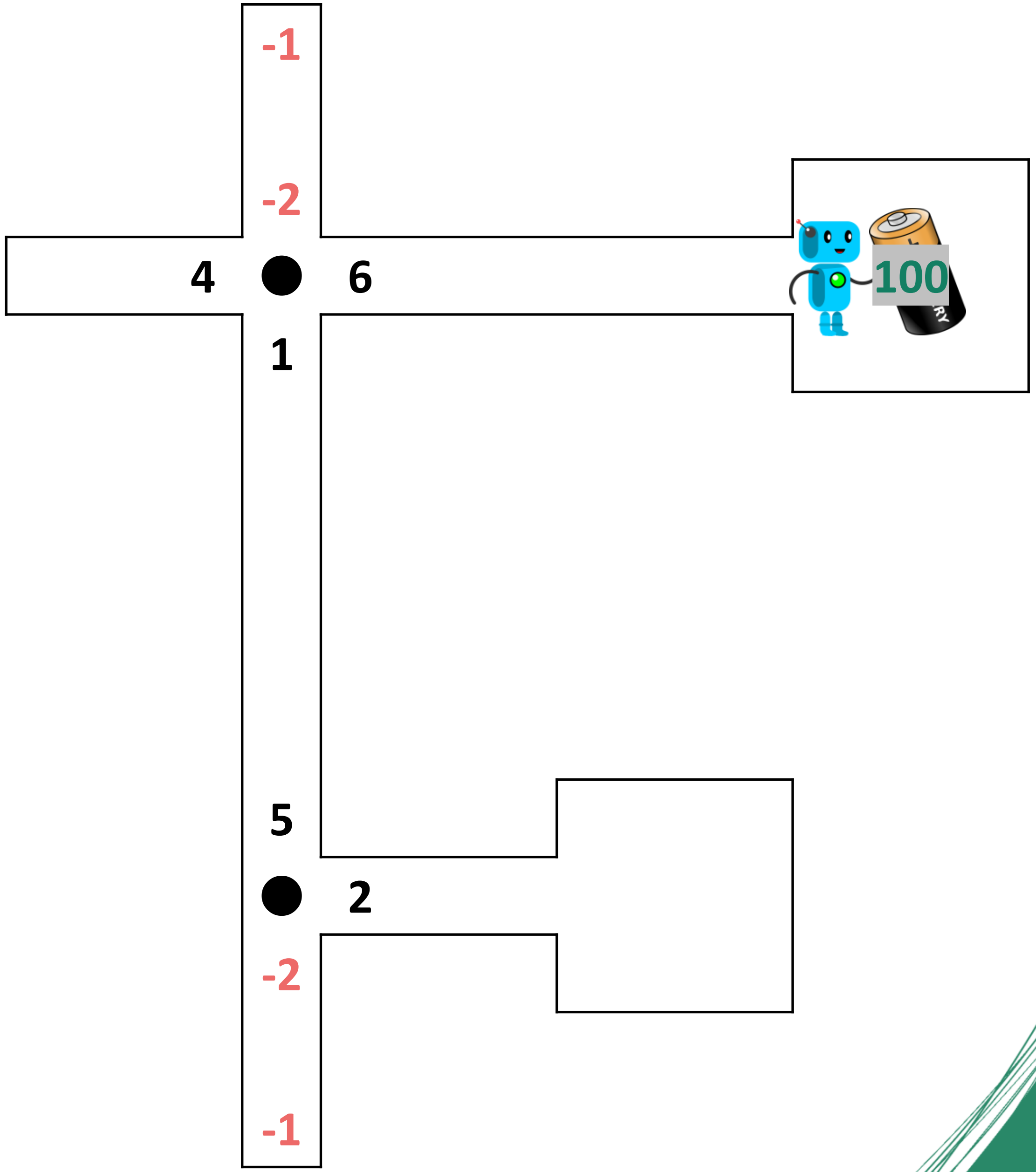
The Game

- The robot **continues** until it **reaches** it’s **goal**



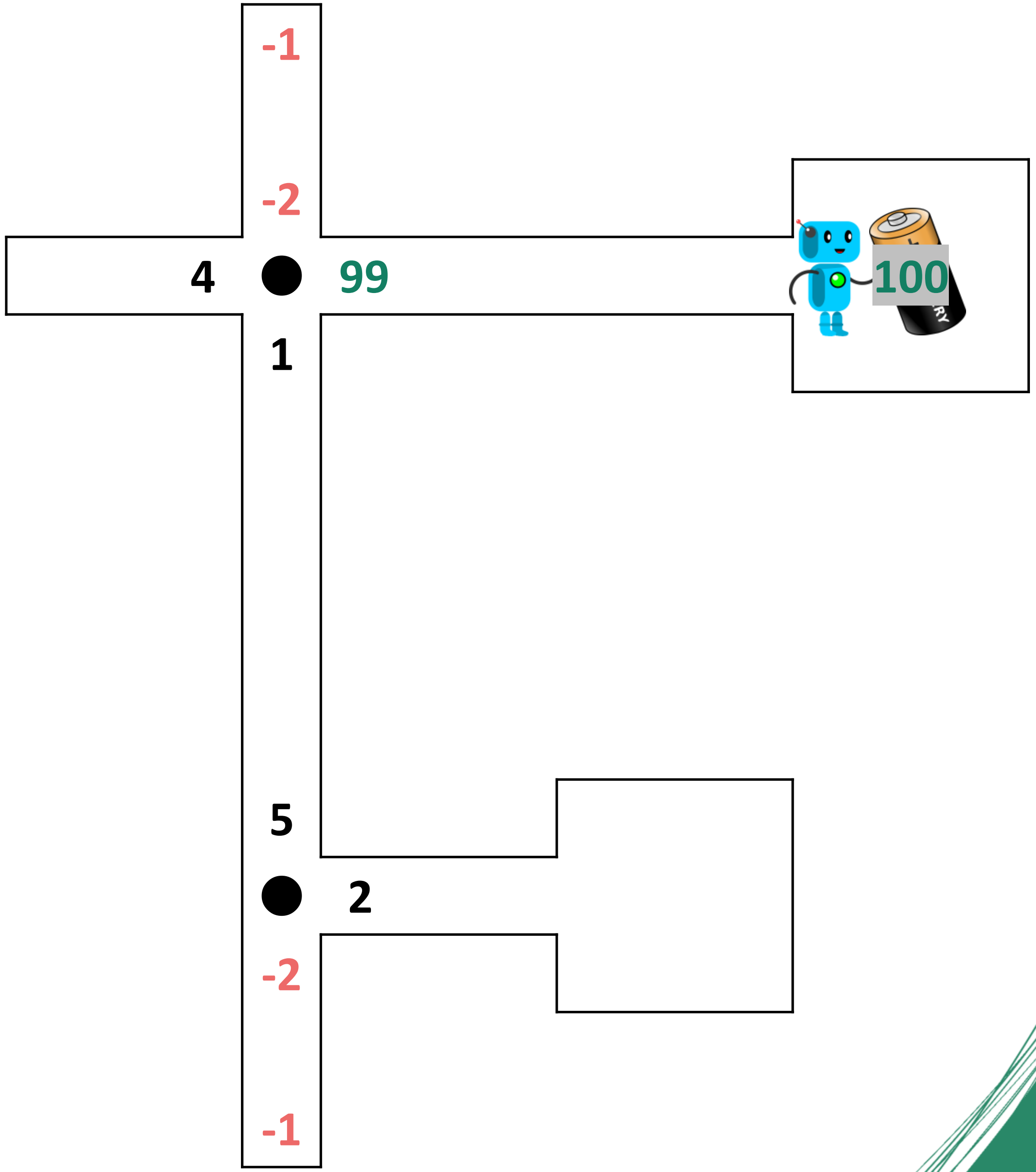
The Game

- The goal in this maze has a value of **100**



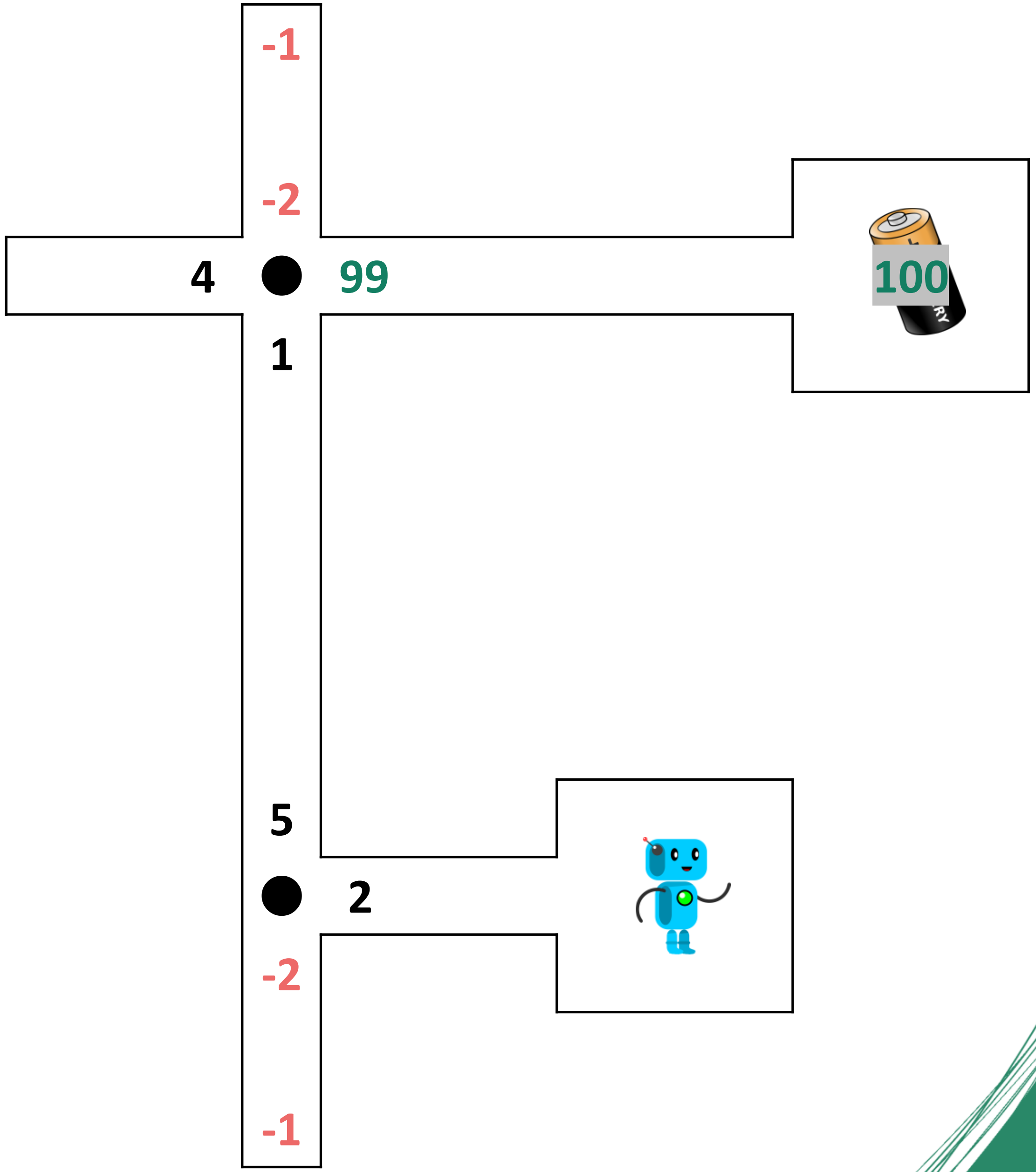
The Game

- The goal in this maze has a value of **100**
- Don’t forget to **update** the path!



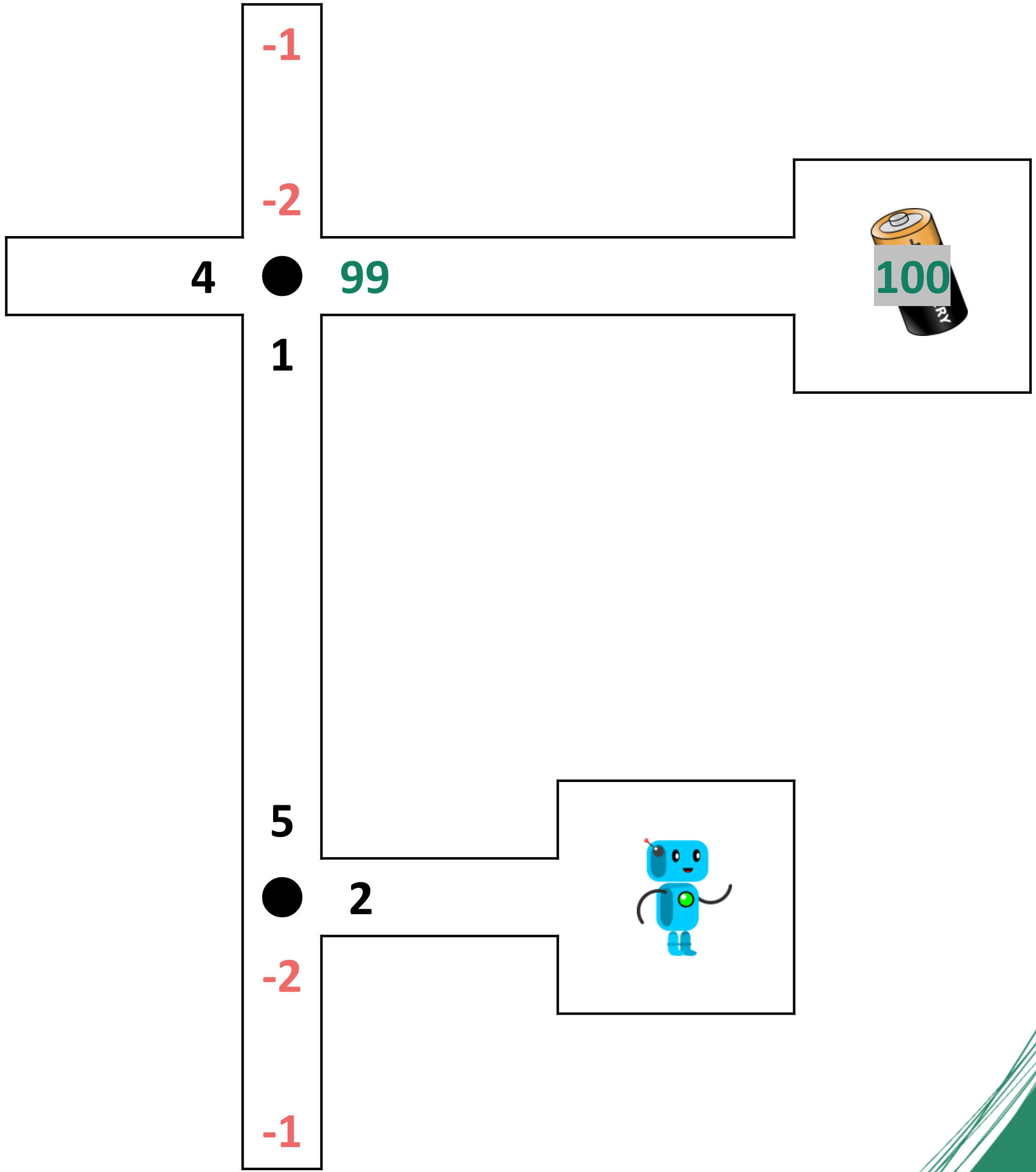
The Game

- The goal in this maze has a value of **100**
- Don't forget to **update** the path!
- Then the robot **returns** to the **first room** and starts another round



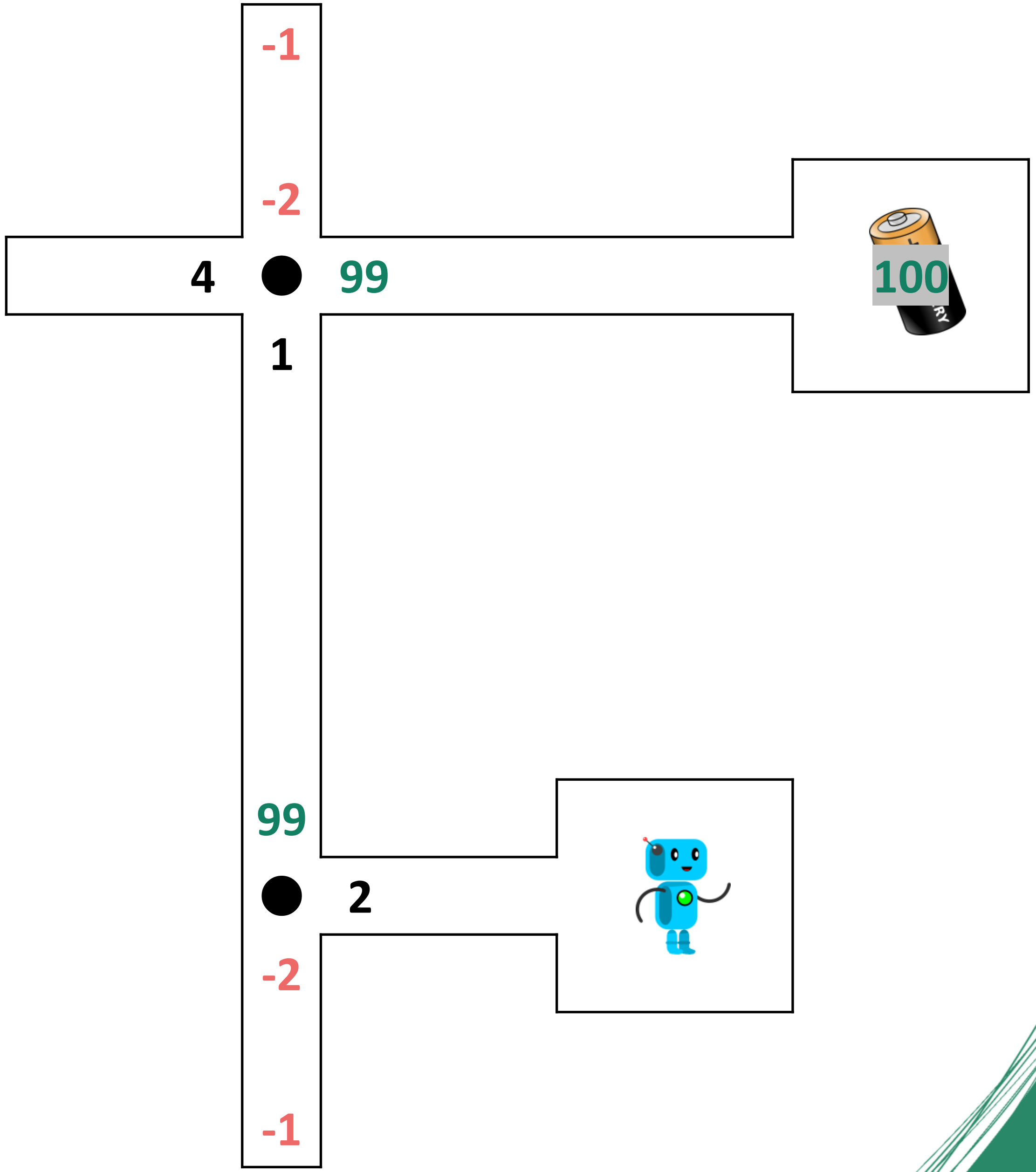
The Game

- **Continue** until the robot **doesn't learn anything new** (write or correct numbers)



The Game

- Finally, the robot has **learned** a reliable **path to its goal**



Exploration vs Exploitation

The Problem

- Some **actions** can have an **immediate positive** effect but a **long term negative effect**

The Problem

- Some **actions** can have an **immediate positive** effect but a **long term negative effect**
    - E.g. the robot takes a path and continues walking, but many steps later it turns out to be a dead end.

The Problem

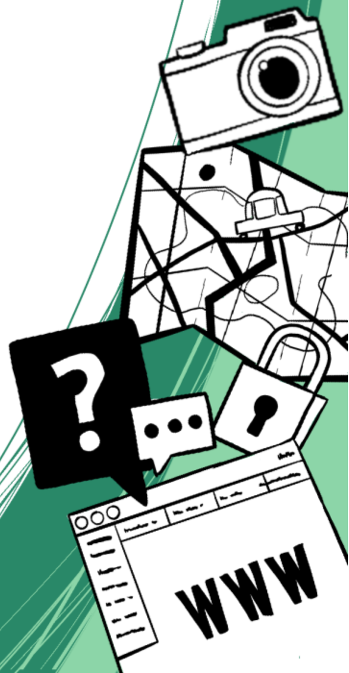
- Some **actions** can have an **immediate positive** effect but a **long term negative effect**
  - E.g. the robot takes a path and continues walking, but many steps later it turns out to be a dead end.
- Some **actions** might only have a **small immediate positive** effect but a **big long term positive effect**

The Problem

- Some **actions** can have an **immediate positive** effect but a **long term negative effect**
  - E.g. the robot takes a path and continues walking, but many steps later it turns out to be a dead end.
- Some **actions** might only have a **small immediate positive** effect but a **big long term positive effect**
  - E.g. the robot finds a rewarding battery, but there would be an even more rewarding one along the next path

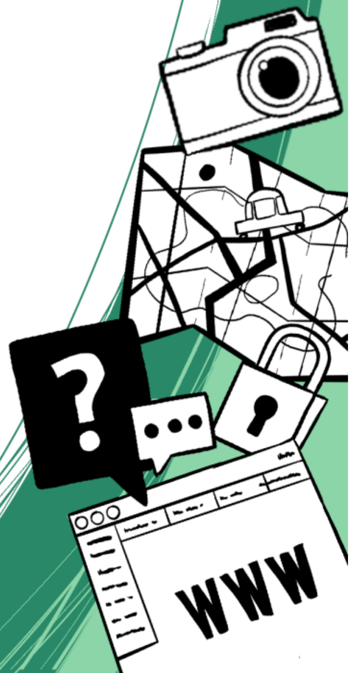
The Solution

- Find a **balance** between **exploiting** already learned information and **exploring** new possibilities



The Solution

- Find a **balance** between **exploiting** already learned information and **exploring** new possibilities
  - E.g. instead of always choosing the path with the **highest number**, the robot could have a **25% chance of taking a random path**



The Solution

- Find a **balance** between **exploiting** already learned information and **exploring** new possibilities
  - E.g. instead of always choosing the path with the **highest number**, the robot could have a **25% chance of taking a random path**
  - This **exploration rate** can also be dynamic, so that it is **high in the beginning** but **gets lower** as the robot improves its understanding of the environment