# WSI laboratorium 6 – Sprawozdanie

Pryimak Andrii-Stepan 336173

## Wprowadzenie

Celem projektu było zaimplementowanie algorytmu Q-learning w środowisku FrozenLake, aby nauczyć agenta efektywnego poruszania się po planszy w kierunku celu, unikając pułapek. FrozenLake jest standardowym środowiskiem dostępnym w bibliotece Gymnasium i symuluje planszę, gdzie agent porusza się po zamarzniętym jeziorze.

## Realizacja

W projekcie zastosowano dwie różne konfiguracje środowiska:

1. **Środowisko niesliskie (No Slippery)**, gdzie agent ma pełną kontrolę nad ruchem.
2. **Środowisko śliskie (Slippery)**, w którym ruch agenta może być losowo zaburzony.  
   Do każdego środowiska zastosowano trzy różne systemy nagród: standardowy, dostosowany 1 oraz dostosowany 2, które kładły nacisk odpowiednio na unikanie ścian, unikanie pułapek oraz premiowanie poprawnych ruchów.

## Wyniki

|  |  |
| --- | --- |
| Bez poślizgu | Z poslizgiem |
| A graph with red lines  Description automatically generated | A graph of red lines  Description automatically generated |
| A graph of a graph of a graph  Description automatically generated with medium confidence | A graph of red and blue lines  Description automatically generated |
| A graph of a graph of a graph  Description automatically generated with medium confidence | A graph with red and blue lines  Description automatically generated |
|  | A graph with red and blue lines  Description automatically generated |

## Wnioski

1. **Wyniki dla wersji bez poślizgu**:
   * W przypadku używania algorytmu Q-learning z domyślnym systemem nagród (1 za dojście do celu, 0 za inne przypadki), agent wykazywał stabilny proces uczenia się, osiągając wyższe wartości średnich nagród w miarę upływu epizodów.
   * Kara za wpadnięcie do dziury wzmocniły proces uczenia się, co umożliwiło agentowi szybsze nauczanie optymalnej polityki.
   * Kara za pójście w ściane poprawiła wynik.
2. **Wersja z poślizgiem (is\_slippery=True)**:
   * Po włączeniu poślizgu i zwiększeniu liczby epizodów do 10000, wyniki uległy znacznemu pogorszeniu. Agent napotkał większe trudności w nauce skutecznej polityki decyzyjnej ze względu na losowość w zachowaniu środowiska.
   * Zmiana systemu nagród miała istotny wpływ na skuteczność algorytmu. Wersja z karą za wpadnięcie do dziury nadal przynosiła korzyści, ale funkcja przyznająca karę za ruch w ścianę nie poprawiła wyników w wersji z poślizgiem. To może sugerować, że w trudniejszych warunkach, takich jak wprowadzenie poślizgu, agent powinien skupić się bardziej na unikaniu nieprzewidywalnych zmian stanu niż na unikaniu ścian.
3. **Wpływ zmienionych funkcji nagrody**:
   * Funkcja nagrody z karą za wpadnięcie do dziury przyczyniła się do poprawy wydajności algorytmu w obu przypadkach, ponieważ agent uczył się szybciej unikać dziur.
   * Wprowadzenie kary za ruch w ścianę poprawiło wyniki w wersji bazowej, ponieważ agent, mając ograniczoną liczbę ruchów, bardziej starał się unikać niepotrzebnych, błędnych działań, co prowadziło do mniejszego marnowania ruchów. Wersja z poślizgiem przyniosła gorsze rezultaty, ponieważ agent unika ścian, ale w rezultacie zbliża się do dziur. Pójście w ścianę nie oznacza automatycznej przegranej, natomiast wpadnięcie do dziury już tak. Dlatego w przypadku środowiska z poślizgiem najlepiej, aby agent trzymał się jak najdalej od dziur, nawet jeśli wiąże się to z ryzykiem zbliżenia do ściany.