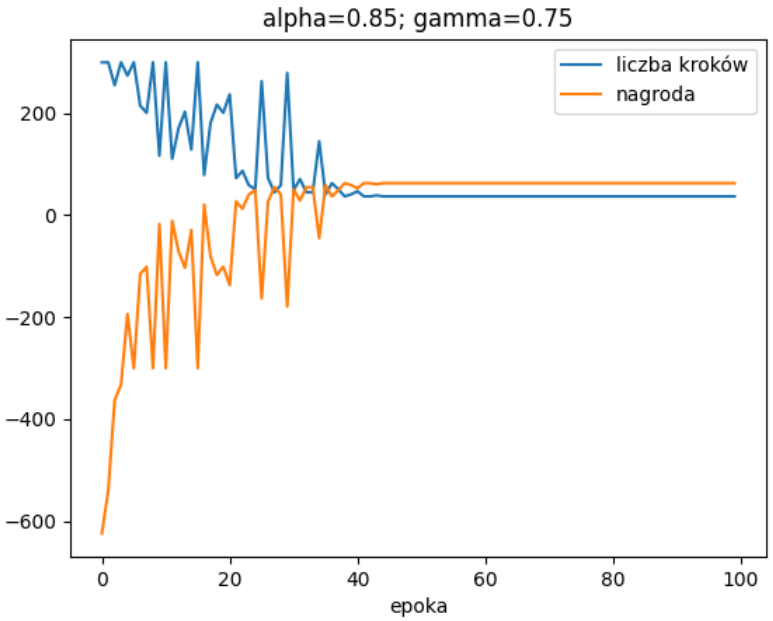
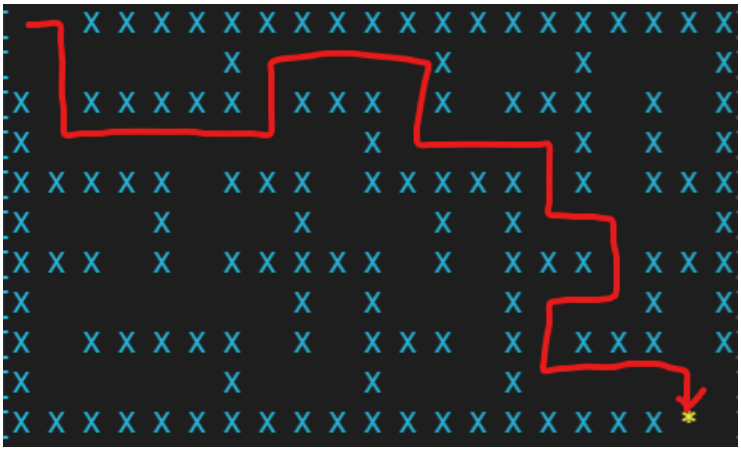
WSI21Z Piotr Szmurło (303785)

Zaimplementować algorytm Q-Learning.
Zebrać i przedstawić na wykresie liczbę wykonanych kroków i naliczoną karę/nagrodę w kolejnych epokach.
Problem do rozwiązania to znalezienie drogi z punktu 'S' do punktu 'F' w "labiryncie" / świecie z przeszkodami.
Rezultatem działania algorytmu powinna być ścieżka w postaci: (1,1)->(0,1)->...->(2,3) oraz ww. wykres.

Labirynt jest ładowany z pliku .txt, gdzie 0 – wolne pole, 1 – zajęte, 3 – start, 9 – meta.

Wynik działania programu:





[0 0]->[1 0]->[1 1]->[2 1]->[3 1]->[3 2]->[3 3]->[3 4]->[3 5]->[3 6]->[3 7]->[2 7]->[1 7]->[1 8]->[1 9]->[1 10]->[1 11]->[2 11]->[3 11]->[3 12]->[3 13]->[3 14]->[3 15]->[4 15]->[5 15]->[5 16]->[5 17]->[6 17]->[7 17]->[7 16]->[7 15]->[8 15]->[9 15]->[9 16]->[9 17]->[9 18]->[9 19]->[10 19]

Kara za próbę ruchu w ścianę: -5; za każdy możliwy ruch: -1; nagroda za osiągnięcie celu: 10.

Najlepsze rezultaty otrzymałem dla alpha(learn rate) = 0.85 i gamma(dyskonto) = 0.75. Przy zbyt małych wartościach gamma algorytm nie jest w stanie rozwiązać labiryntu, gdyż potencjalna nagroda staje się znikoma. Dla mniejszych wartości współczynnika alpha uczenie trwa dłużej.

Występujące na wykresie piki świadczą o losowości ruchów na początku algorytmu (strategia epsilon-zachłanna). Wpływa ona na eksplorację, np. możliwość znalezienia krótszej drogi niż ta już odkryta.