

Aleksander Prus, 218404
Rafał Szyszka, 218207

Wrocław, d. 4 kwietnia 2018

Monte Carlo Tree Search

Rok akad. 2017/2018, DAN

Mgr inż. Jan Jakubik

Spis treści

1	Cel ćwiczenia	3
2	Realizacja	3
2.1	Naiwni gracze	3
2.1.1	Gracz losowy	3
2.1.2	Gracz agresywny	3
2.1.3	Gracz kontrolujący	3
2.2	Gracz Monte Carlo	4
3	Badania	4
4	Wnioski	8

1 Cel ćwiczenia

Celem ćwiczenia była implementacja uproszczonej wersji popularnej gry Hearthstone, z graczem komputerowym, którego zachowanie zdefiniowane będzie algorytmem Monte Carlo Tree Search. Dodatkowo należało zaimplementować trzech graczy naiwnych:

- losowy,
- agresywny,
- kontrolujący.

Gracz agresywny w pierwszej kolejności skupia się na zadaniu jak największej ilości obrażeń bohaterowi gracza. Natomiast gracz kontrolujący skupia się na wyeliminowaniu minionów przeciwnika.

2 Realizacja

Do realizacji zadania wybrano język Python, ze względu na mnogość zewnętrznych bibliotek oraz wysoki poziom abstrakcji języka.

2.1 Naiwni gracze

Logika naiwnych graczy została zimplementowana w pakiecie **naive-bots**. Ich zachowanie określają dwie metody **card-rating** oraz **choose-targets**. Pierwsza z nich odpowiada za dobór kart do zagrania z ręki, w turze wybrangeo bota. Natomiast druga metoda, odpowiada za wybór celów do zaatakowania po stronie przeciwnika.

2.1.1 Gracz losowy

Ten gracz w sposób losowy wybiera karty do zagrania oraz cele, które zaatakuje w danej turze.

2.1.2 Gracz agresywny

Ten gracz preferuje karty z dużą ilością ataku. Za swoje cele, w pierwszej kolejności, bierze karty przeciwnika z właściwością **[TAUNT]**, następnie wybiera karty z wysoką wartością życia (większy od 4), jeżeli takich kart przeciwnik nie posiada na stole, bot agresywny atakuje jego bohatera.

2.1.3 Gracz kontrolujący

Ten gracz preferuje karty defensywne, z dużą ilością życia. Za swoje cele, w pierwszej kolejności, bierze karty przeciwnika z właściwością **[TAUNT]**, następnie wybiera karty z wysoką wartością życia (większy od 4), następnie wybiera pozostałe karty przeciwnika na stole. Bohater przeciwnika jest ostatnim wybieranym celem gracza kontrolującego.

2.2 Gracz Monte Carlo

Jego działanie opiera się na czterech krokach budowy drzewa decyzyjnego:

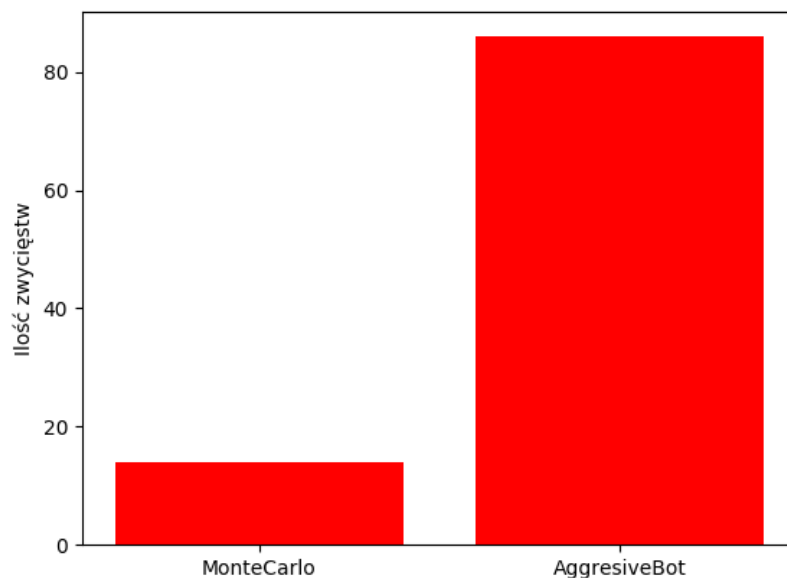
- wybór,
- rozrost,
- symulacja,
- propagacja wstecz.

Te cztery kroki są wykonywane dopóki czas przydzielony na wykonanie ruchu nie upłynie. Następnie wybierany jest ruch z korzenia drzewa, dla którego została wykonana największa liczba symulacji.

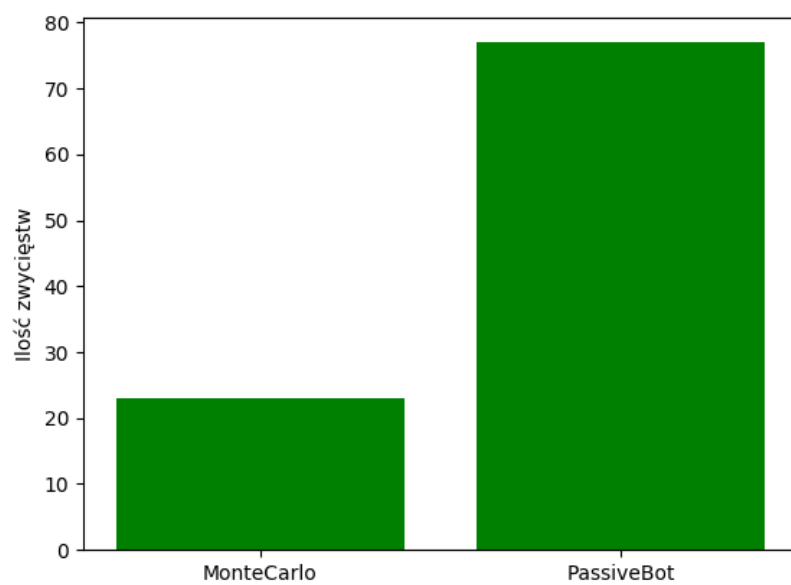
3 Badania

Łącznie rozegrano sześć potyczek, trzy dla gracza Monte Carlo z maksymalnym czasem na ruch wynoszącym 20 sekund i trzy dla gracza Monte Carlo z maksymalnym czasem na ruch wynoszącym 60 sekund.

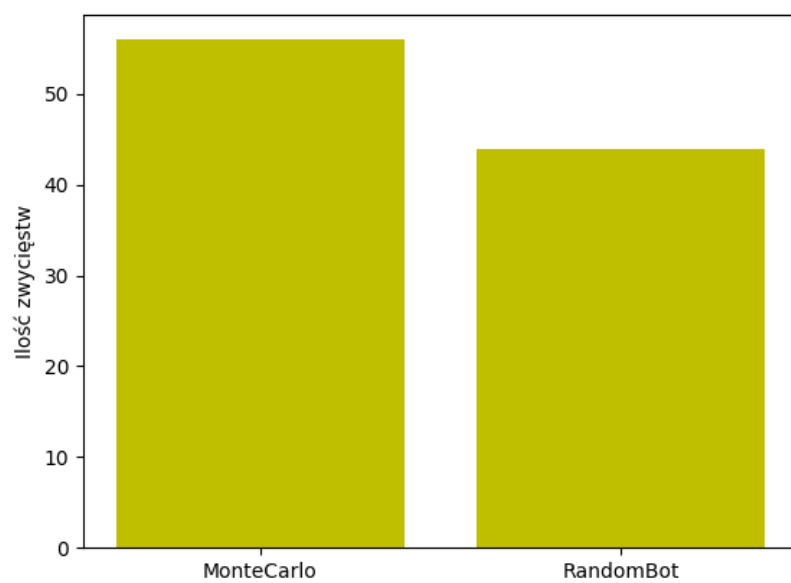
Gracz Monte Carlo grał z każdym z naiwnych graczy stukrotnie dla obu ustawień czasowych. Wyniki zaprezentowano poniżej.



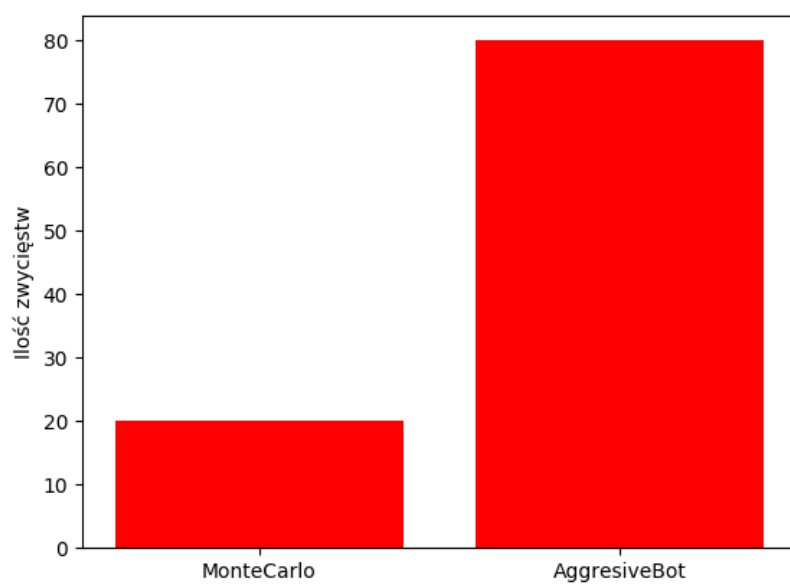
Rysunek 1: Monte Carlo versus Gracz Agresywny, 100 rozgrywek, 20 sekund na ruch



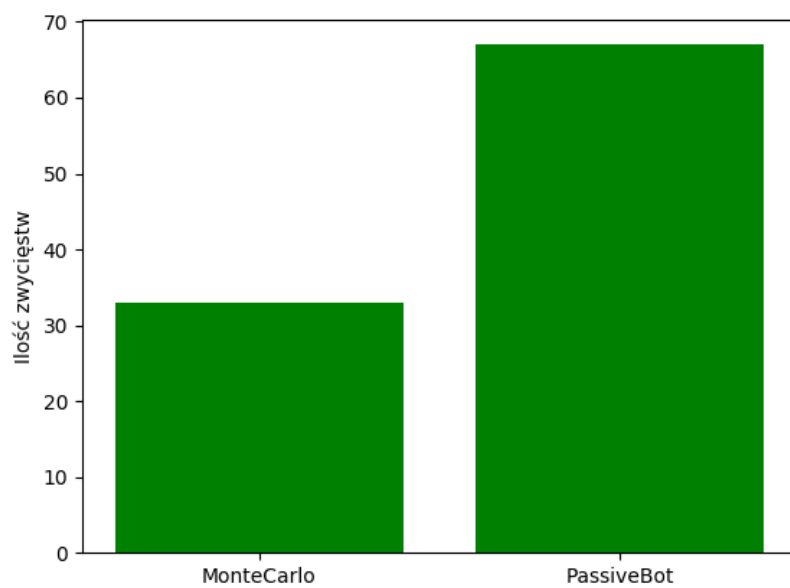
Rysunek 2: Monte Carlo versus Gracz Kontrolujący, 100 rozgrywek, 20 sekund na ruch



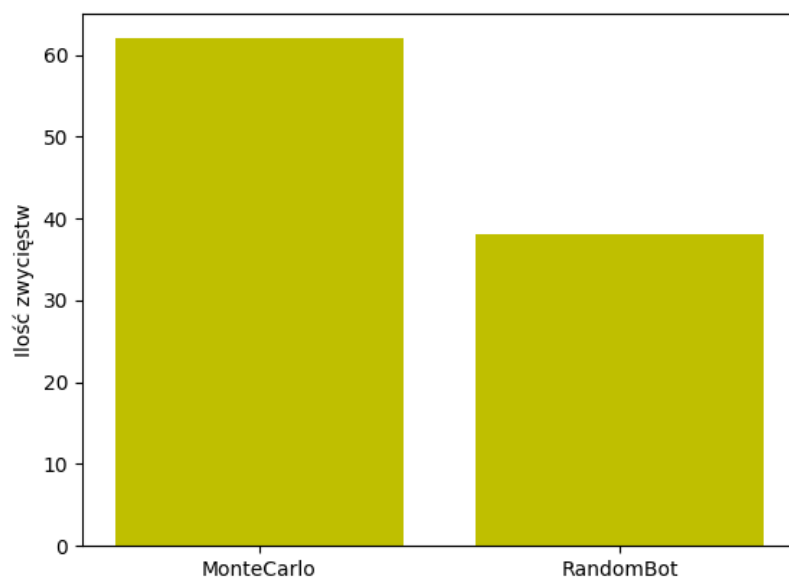
Rysunek 3: Monte Carlo versus Gracz Losowy, 100 rozgrywek, 20 sekund na ruch



Rysunek 4: Monte Carlo versus Gracz Agresywny, 100 rozgrywek, 60 sekund na ruch



Rysunek 5: Monte Carlo versus Gracz Kontrolujący, 100 rozgrywek, 60 sekund na ruch

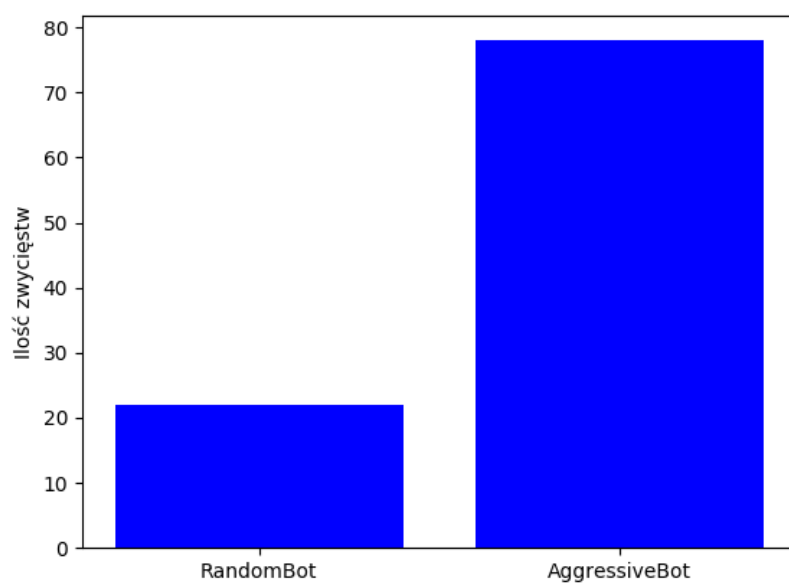


Rysunek 6: Monte Carlo versus Gracz Losowy, 100 rozgrywek, 60 sekund na ruch

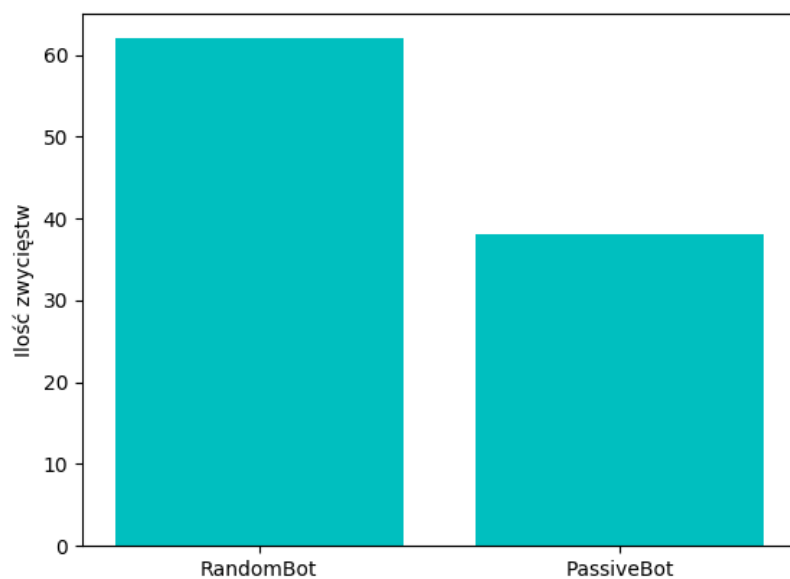
Dodatkowo, dla porównania, rozegrano potyczki:

- Gracz Losowy versus Gracz Agresywny,
- Gracz Losowy versus Gracz Pasywny,
- Gracz Pasywny versus Gracz Agresywny.

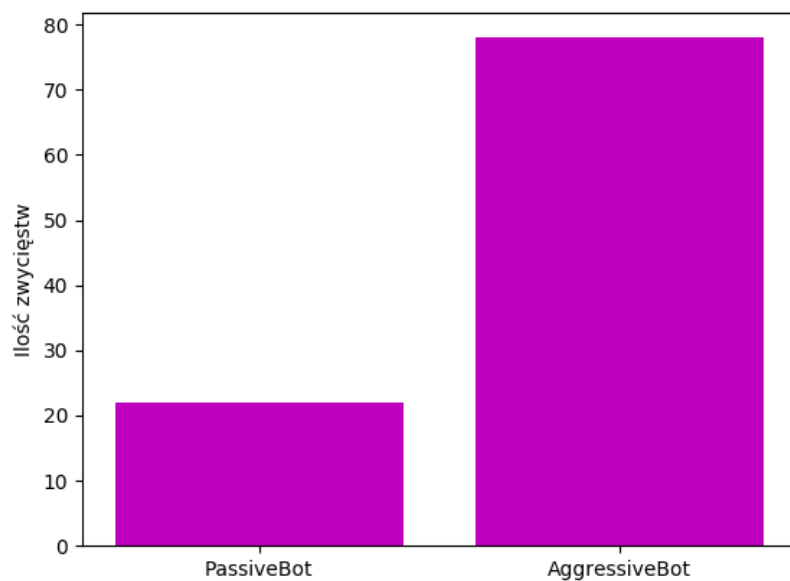
Wyniki rozgrywek zaprezentowano na wykresach poniżej.



Rysunek 7: Gracz Losowy versus Gracz Agresywny, 100 rozgrywek



Rysunek 8: Gracz Losowy versus Gracz Kontrolujący, 100 rozgrywek



Rysunek 9: Gracz Kontrolujący versus Gracz Agresywny, 100 rozgrywek

4 Wnioski

Gracz Monte Carlo nie okazał się najlepszym graczem spośród tych zaimplementowanych. Jego 'jakość' uzależniona jest od czasu, jaki zostanie mu przydzielony na wykonanie ruchu. Dzieje się tak, ponieważ drzewo gry Hearthstone, nawet w wersji uproszczonej,

jest bardzo duże i przejrzanie go w całości zajmuje wiele czasu. Znacznie prostszy w implementacji i szybciej działający gracz Agresywny okazał się być najlepiej grającym przeciwnikiem.