**Klasa GameOfLife**

**\_\_init\_\_(self, rules, height, width, canvas)**

* Inicjalizuje klasę LifeGame.
* Parametry:
  + **rules** (str): Zasady gry w życie, określone jako ciąg znaków w formacie "stay\_alive/become\_alive", gdzie każda cyfra reprezentuje liczbę sąsiadów wymaganą do utrzymania komórki przy życiu lub do jej ożywienia.
  + **height** (int): Wysokość planszy gry.
  + **width** (int): Szerokość planszy gry.
  + **canvas** (tkinter.Canvas): Obiekt płótna, na którym rysowana jest plansza gry.

**clickingCell(self, event)**

* Obsługuje zdarzenie kliknięcia na komórkę na planszy gry.
* Parametry:
  + **event** (tkinter.Event): Obiekt zdarzenia zawierający informacje o zdarzeniu kliknięcia.

**drawBoard(self)**

* Rysuje aktualny stan planszy gry na płótnie.

**startGame(self)**

* Rozpoczyna grę, uruchamiając pętlę gry.

**stopGame(self)**

* Zatrzymuje grę, kończąc pętlę gry.

**nextStep(self)**

* Oblicza i wyświetla następny krok w grze.
* Ta metoda jest wywoływana rekurencyjnie z opóźnieniem określonym przez prędkość gry.

**calculateNextStep(self)**

* Oblicza następny krok w grze na podstawie aktualnego stanu planszy gry i określonych zasad.

**countNeighbours(self, a, b)**

* Liczy liczbę żywych sąsiadów wokół konkretnej komórki.
* Parametry:
  + **a** (int): Indeks wiersza komórki.
  + **b** (int): Indeks kolumny komórki.
* Zwraca:
  + **neighbours** (int): Liczba żywych sąsiadów.

**pointAlive(self, a, b)**

* Sprawdza, czy określona komórka jest żywa.
* Parametry:
  + **a** (int): Indeks wiersza komórki.
  + **b** (int): Indeks kolumny komórki.
* Zwraca:
  + **alive** (bool): True, jeśli komórka jest żywa, False w przeciwnym przypadku.

**updateSpeed(self, value)**

* Aktualizuje prędkość gry na podstawie wybranej opcji z menu prędkości.
* Parametry:
  + **value** (str): Wybrana wartość prędkości.

**Funkcja start()**

* Inicjalizuje grę i konfiguruje interfejs użytkownika (GUI).
* Zwraca:
  + **canvas** (tkinter.Canvas): Obiekt płótna, na którym rysowana jest plansza gry.

**Inicjalizacja GUI i główna pętla**

* Główne okno interfejsu użytkownika jest tworzone za pomocą biblioteki tkinter.
* Dodawane są etykiety, pola tekstowe i przycisk umożliwiające wprowadzenie parametrów gry.
* Funkcja **start()** jest wywoływana po kliknięciu przycisku "Start Game".
* Okno interfejsu użytkownika jest uruchamiane w głównej pętli za pomocą **root.mainloop()**.