

# Specyfikacja Funkcjonalna

Daniel Ślusarczyk i Jakub Łaba

10.03.2021

## **Streszczenie**

Dokument stanowi opis szczegółowy projektu *The Game of Life* oraz związany z nim zakres prac, funkcje programu, interakcje zachodzące między system a użytkownikiem i instrukcje użytkowania.

# Spis treści

1	Cel Projektu . . . . .	1
2	Dane Wejściowe . . . . .	1
3	Argumenty wywołania programu . . . . .	2
4	Uruchomienie programu . . . . .	2
5	Wynik działania programu . . . . .	3
6	Scenariusz działania programu: . . . . .	3
6.1	Tryb podstawowy: . . . . .	3
6.2	Tryb step by step: . . . . .	3
7	Rozszerzony scenariusz działania programu . . . . .	4
7.1	Tryb podstawowy . . . . .	4
7.2	Tryb step by step . . . . .	5
7.3	Tryb save . . . . .	5

---

## 1 Cel Projektu

Program emuluje działanie automatu komórkowego Johna Conwaya - Grę w Życie (ang. *The Game of Life*). Na prostokątnej planszy znajdują się komórki żywe (zazwyczaj oznaczane kolorem czarnym) oraz martwe (zazwyczaj oznaczane kolorem białym, lub brakiem koloru). "Życie" na planszy toczy się przez określoną liczbę generacji, które są rozdzielone dyskretnymi odstępami czasowymi. W każdej generacji komórki ożywają/obumierają na następujących zasadach:

- **Komórki żywe** – pozostają żywe w kolejnej generacji jeżeli posiadają dokładnie 2 lub 3 żywych sąsiadów, w każdym innym przypadku umierają
- **Komórki martwe** – ożywają w kolejnej generacji jeżeli posiadają dokładnie trzech żywych sąsiadów, w przeciwnym przypadku pozostają martwe

Komórki znajdujące się poza planszą są uznawane za martwe.

## 2 Dane Wejściowe

Program potrzebuje danych z informacjami o stanie wyjściowej planszy (generacji 0), oraz o liczbie generacji, którą użytkownik chce wygenerować. Plik wejściowy powinien zawierać trzy linie, sformatowane następująco:

```
1: row col
2: x1 x2 ... xn
3: y1 y2 ... yn
Gdzie:
```

- **row** – liczba rzędów planszy
- **col** – liczba kolumn planszy
- **x1, x2, ..., xn** – współrzędne x-owe kolejnych żywych komórek
- **y1, y2, ..., yn** – współrzędne y-owe kolejnych żywych komórek

(x1, y1) są współrzędnymi pierwszej żywej komórki, (x2, y2) – drugiej, (xn, yn) – n-tej, itd. Współrzędne określa się względem klasycznego 2-wymiarowego kartezjańskiego układu współrzędnych.

### Przykład

Chcąc wczytać planszę 2x3 z jedną żywą komórką w lewym górnym rogu, oraz jedną w prawym dolnym rogu, plik powinien wyglądać w następujący sposób:

```
1: 2 3
2: 1 3
3: 2 1
```

---

### 3 Argumenty wywołania programu

#### Niezbędne do uruchomienia programu:

plik wejściowy – ścieżka do pliku z informacjami o stanie planszy wyjściowej

liczba generacji – liczba naturalna większa od 0, określająca pożądaną przez użytkownika

liczbę generacji

#### Flagi opcjonalne:

Flagi należy umieścić jako ostatnie argumenty wywołania, poza nielicznymi wyjątkami opisanymi poniżej, ich kolejność względem siebie nawzajem nie jest istotna

- **-sbs** – tryb *step by step*, tzn. program będzie pokazywał wszystkie generacje, czekając na kolejne instrukcje
- **-save** – program wywołany z tą flagą zapisze do pliku wynikowego ostatnią przeprowadzoną generację, sformatowaną identycznie jak plik wejściowy (dzięki temu można użyć pliku wynikowego jako wejścia do kolejnego uruchomienia programu), nazwę pliku wynikowego należy podać bezpośrednio po flagie -save, w przypadku chęci zapisania pliku w innym folderze, należy podać jego nazwę razem z pożądaną ścieżką
- **-overwrite** – przy wywołaniu z tą flagą, ostatnia generacja zostanie nadpisana w pliku podanym jako wejściowy, nie można używać w połączeniu z flagą -save
- **-refresh** – program będzie odświeżał ekran, pokazując kolejne generacje, przy wywołaniu z tą flagą można od razu po niej podać wartość w milisekundach, którą program ma odczekać pomiędzy odświeżeniami ekranu (1s = 1000ms), jeżeli wartość nie zostanie podana, domyślna zostanie ustawiona na 100ms (0,1s), jeżeli zamiast wartości zostanie użyta flaga -sbs, program będzie odświeżał ekran po wciśnięciu klawisza przez użytkownika

### 4 Uruchomienie programu

Program został przygotowany do uruchomienia w systemie Linux (zalecana dystrybucja: Ubuntu 20.04).

Uruchomienie programu odbywa się poprzez wpisanie niezbędnych argumentów wywołania programu w terminalu systemu Linux:

```
./TGOL [plik wejściowy] [liczba generacji]
```

#### Przykład:

Znajdując się w folderze projektu i mając odpowiednio przygotowany plik wejściowy (zgodnie z punktem "dane wejściowe") pod nazwą "dane1" w folderze "generacje" i chcąc przeprowadzić 10 generacji oraz zapisać wynik w pliku "wynik" powinniśmy wpisać następujące polecenie:

```
./TGOL generacje/dane1 10 -save wynik
```

---

## 5 Wynik działania programu

Wynikiem działania programu jest przeanalizowanie przez program wszystkich następujących generacji i wyświetlenie finalnej (zadeklarowanej w argumencie [liczba generacji]) w oknie terminala. Jeśli program został uruchomiony w trybie *step by step* to wynikiem działania pierwszego etapu będzie wyświetlenie kolejnej zachodzącej generacji i oczekiwania na kolejne komendy. Po osiągnięciu finalnej generacji w obu przypadkach program kończy działanie. Jeśli program został uruchomiony z flagą - save to wszystkie przeanalizowane generacje zostają zapisane w pliku podanym przez użytkownika w formacie zgodnym z danymi wejściowymi. Dodatkowo flaga -overwrite powoduje nadpisanie pliku.

## 6 Scenariusz działania programu:

### 6.1 Tryb podstawowy:

**Wywołanie:**

```
./TGOL [plik wejściowy] [liczba generacji]
```

**Wynik:**

Przeprowadzenie przez program wszystkich generacji. Wyświetlenie informacji o ostatniej generacji w oknie terminala.

### 6.2 Tryb step by step:

**Wywołanie:**

```
./TGOL [plik wejściowy] [liczba generacji] -sbs
```

**Wynik:**

Przeprowadzenie przez program pierwszej generacji. Wyświetlenie informacji na jej temat w oknie terminala. Czekanie na instrukcje od użytkownika.

**Możliwe kroki:**

- Wpisanie polecenia *n*

Wynik:

Przeprowadzenie przez program kolejnej generacji. Wyświetlenie informacji na jej temat w oknie terminala. Czekanie na instrukcje od użytkownika.

- Wpisanie polecenia *con*

Wynik:

Wyłączenie przez program trybu step by step. Przeprowadzenie przez program wszystkich pozostałych generacji. Wyświetlenie informacji o ostatniej generacji w oknie terminala.

- 
- Wpisanie polecenia *out*  
Wynik:  
Zakończenie działania programu.

## 7 Rozszerzony scenariusz działania programu

### 7.1 Tryb podstawowy:

**Możliwe wyniki kończące działanie programu:**

Komunikat	Przyczyna
INCORRECT_NUMBER_OF_ARGS	Niepoprawna ilość argumentów uruchomienia programu
UNKNOWN_FLAG	Użycie nieznanej flagi
INPUT_ERR	Nie można otworzyć podanego pliku wejściowego
INCORRECT_GENS	Niewłaściwa liczba generacji
INPUT_NOT_INT	W pliku wejściowym znajdują się wartości niebędące liczbami naturalnymi
INPUT_SHORT	W pliku wejściowym znajduje się za mało danych określających wymiary planszy
INPUT_XY	W pliku wejściowym liczba współrzędnych x-owych jest niezgodna z liczbą współrzędnych y-owych
AMBIGUOUS_OUT	Użyto niedozwolonego połączenia flag "-save" oraz "-overwrite"
NO_OUT	Wywołano z flagą "-save", ale nie podano nazwy pliku wynikowego

**Możliwe wyniki kontnuujące działanie programu:**

Komunikat	Przyczyna
FILENAME_TAKEN	Wywołano z flagą "-save" i podano nazwę pliku wynikowego, ale w danej lokalizacji już istnieje plik o tej nazwie. Program pyta o chęć nadpisania pliku.

## 7.2 Tryb step by step:

**Możliwe wyniki kończące działanie programu:**

Komunikat	Przyczyna
INCORRECT_NUMBER_OF_ARGS	Niepoprawna ilość argumentów uruchomienia programu
UNKNOWN_FLAG	Użycie nieznanego flagi
INPUT_ERR	Nie można otworzyć podanego pliku wejściowego
INCORRECT_GENS	Niewłaściwa liczba generacji
INPUT_NOT_INT	W pliku wejściowym znajdują się wartości niebędące liczbami naturalnymi
INPUT_SHORT	W pliku wejściowym znajduje się za mało danych określających wymiary planszy
INPUT_XY	W pliku wejściowym liczba współrzędnych x-owych jest niezgodna z liczbą współrzędnych y-owych
NO_OUT	Wywołano z flagą "-save", ale nie podano nazwy pliku wynikowego

**Możliwe wyniki kontynuujące działanie programu:**

Komunikat	Przyczyna
INCORRECT_ORDER	Wpisano złe polecenie. Program kontynuuje czekanie na polecenie

## 7.3 Tryb save:

**Możliwe wyniki kończące działanie programu:**

Komunikat	Przyczyna
INCORRECT_NUMBER_OF_ARGS	Niepoprawna ilość argumentów uruchomienia programu
UNKNOWN_FLAG	Użycie nieznanego flagi
INPUT_ERR	Nie można otworzyć podanego pliku wejściowego
INCORRECT_GENS	Niewłaściwa liczba generacji
INPUT_NOT_INT	W pliku wejściowym znajdują się wartości niebędące liczbami naturalnymi
INPUT_SHORT	W pliku wejściowym znajduje się za mało danych określających wymiary planszy
INPUT_XY	W pliku wejściowym liczba współrzędnych x-owych jest niezgodna z liczbą współrzędnych y-owych
NO_OUT	Wywołano z flagą "-save", ale nie podano nazwy pliku wynikowego



---

**Możliwe wyniki kontuujące działanie programu:**

Komunikat	Przyczyna
FILENAME_TAKEN	Wywołano z flagą "-save" i podano nazwę pliku wynikowego, ale w danej lokalizacji już istnieje plik o tej nazwie. Program pyta o chęć nadpisania pliku.