

09.06.2020  
Warszawa

# Specyfikacja funkcjonalna programu JLawn

<https://github.com/Liath44/JLawn>  
Maciej Dragun  
PW WE 298748

## 1) Pojęcia

W specyfikacji mogę posługiwać się pewnymi pojęciami/skrótami myślowymi. Poniżej podaję ich definicje wraz z angielskimi odpowiednikami, którymi posługiwałem się w komentarzach do kodu

**Warunki (conditions)** – parametry używane podczas operacji podlewania trawnika np. promień podlewaczki lub kształt trawnika

**Podlewalny pixel (waterable pixel)** – pixel, który nie jest ścianą

**Obszar (area)** – maksymalny zbiór pixeli w którym z każdego pixela istnieje składająca się z podlewalnych pixeli ścieżka do każdego innego pixela (idea obszaru jest analogiczna do grafów spójnych)

**Rektangulizacja (rectangulization)** – podział obszaru na maksymalnie szerokie prostokąty

**Kompresja (compression)** – skalowanie współczynników podlania pixeli tak aby możliwa była reprezentacja graficzna trawnika za pomocą 106 odcieni zieleni i 106 odcieni czerwieni

**Współczynnik kompresji (compression factor)** – wartość przez którą skalowane są współczynniki podlania pixeli w ramach kompresji

**Reset (reset)** – w kontekście trawnika jest to przywrócenie go do stanu pierwotnego tj. przed podlaniem

## 2) Opis teoretyczny zagadnienia

Celem projektu było stworzenie programu równomiernie rozstawiającego podlewaczki po zadanym trawniku. Program powinien być napisany w Javie, posiadać graficzny interfejs użytkownika oraz umożliwiać eksport zarówno listy ustawionych podlewaczek do pliku jak i 8-bitowej bitmapy reprezentującej podlany trawnik.

Trawniki powinny dawać się wczytać poprzez podanie ścieżki do pliku go reprezentującego oraz naciśnięcie przycisku potwierdzającego chęć jego importu. Program powinien reagować na błędny format pliku.

Po naciśnięciu odpowiedniego przycisku trawnik powinien zostać równomiernie podlany za pomocą podlewaczek dostępnych w 4 wariantach:

- **Podlewaczka 360** – podlewa N-krotnie obszar w postaci pełnego koła o promieniu R

- **Podlewaczka 270** – podlewa  $2 \cdot N$ -krotnie obszar w postaci koła z wyciętą ćwiartką o promieniu  $2 \cdot R$
- **Podlewaczka 180** – podlewa  $3 \cdot N$ -krotnie obszar w postaci półkoła o promieniu  $3 \cdot R$
- **Podlewaczka 90** – podlewa  $4 \cdot N$ -krotnie obszar w postaci ćwiartki koła o promieniu  $4 \cdot R$

W przypadku natrafienia na ścianę woda odbija się od niej lustrzanie

Program powinien umożliwiać modyfikację niektórych ze swoich parametrów:

- **Liczba cykli** podlewaczki360  $N$  (domyślnie 1)
- **Promień** podlewaczki360  $R$  (domyślnie 50 pixeli)
- **Obecność** lustrzanego odbicia w przypadku natrafienia wody na ścianę (domyślnie true)

Program nie umożliwia pokazania animacji podlewania trawnika

### 3) Kompilacja i uruchomienie programu

Program można skompilować wchodząc do folderu src i wpisując komendę:

*javac Main.java*

Kompilacja pozwala na jego uruchomienie za pomocą polecenia:

*java Main*

Uruchomienie programu wyświetla GUI udostępniające następujące funkcjonalności:

- Zmiana liczby cykli poprzez wprowadzenie liczby i zatwierdzenie jej przyciskiem
- Włączenie/wyłączenie odbicia lustrzanego wody poprzez naciśnięcie odpowiedniego przycisku
- Zmiana promienia podlewaczki360 poprzez wprowadzenie liczby i zatwierdzenie jej przyciskiem

- Import pliku reprezentującego niepodlany trawnik poprzez podanie ścieżki do pliku i zatwierdzenie jej przyciskiem
- Podlanie aktualnie wczytanego trawnika poprzez naciśnięcie odpowiedniego przycisku
- Export listy ustawionych podlewaczek do pliku SPRINKLERS poprzez naciśnięcie odpowiedniego przycisku
- Export graficznej reprezentacji trawnika do pliku BITMAP poprzez naciśnięcie odpowiedniego przycisku
- Powiadomienie użytkownika o statusie ostatniego polecenia

Program można zamknąć klikając czerwony przycisk w górnym prawym rogu

#### 4) Działanie poszczególnych części programu

Importowany plik powinien mieć odpowiedni format – n wierszy złożonych ze znaków ‘\*’ oraz ‘#’. Wszystkie wiersze powinny mieć równe sobie ilości znaków i kończyć się znakami końca linii (enterem) z wyjątkiem ostatniego wiersza. n nie powinno przekraczać 80, z kolei liczba znaków w każdym wierszu nie powinna przekraczać 41. Zadany plik nie powinien być pusty. Program informuje o niepoprawnym formacie pliku.

Plik odczytywany jest znak po znaku. Symbol ‘\*’ jest równoważny kwadratowi 100 na 100 podlewalnych pixeli. Symbol ‘#’ jest równoważny kwadratowi 100 na 100 pixeli ściany

O ile jakkolwiek trawnik został zaimportowany, naciśnięcie przycisku „water” spowoduje podział trawnika na obszary, rektangulizację każdego z nich oraz równomierne podlanie każdego z otrzymanych prostokątów w zależności od jego rozmiarów oraz promienia podlewaczki<sup>360</sup>

Raz podlany trawnik można ponownie podlać wyłącznie przy zmianie warunków. W takim wypadku trawnik zostaje zresetowany.

Naciśnięcie przycisku „to file” generuje lub aktualizuje SPRINKLERS, którego zawartością jest lista podlewaczek użytych do podlania trawnika wraz z ich parametrami.

Naciśnięcie przycisku „bitmap” generuje lub aktualizuje 8-bitową bitmapę zapisaną w pliku BITMAP, która jest graficzną reprezentacją trawnika. Dokonywana jest kompresja (patrz sekcja 1. Pojęcia)

## 5) Potencjalne błędy

Polecenia kończące się niepowodzeniem powodują wyświetlenie się osobnego okienka z komunikatem o błędzie. Poniżej lista potencjalnych błędów i nazw wyjątków je generujących:

- **EmptyFileException** – wyrzucany gdy odczytywany plik, mający reprezentować trawnik, jest pusty
- **ImproperCharException** – wyrzucany gdy odczytywany plik, mający reprezentować trawnik, zawiera znaki nie będące ‘\*’, ‘#’ ani znakiem końca linii
- **InconsistentCharAmountException** – wyrzucany gdy wiersze odczytywanego pliku, mającego reprezentować trawnik, posiadają nierówne ilości znaków
- **TooManyColumnsException** – wyrzucany gdy liczba znaków w dowolnym wierszu pliku, mającym reprezentować trawnik, przekracza 40
- **TooManyRowsException** – wyrzucany gdy liczba wierszy w pliku, mającym reprezentować trawnik, przekracza 80
- **AlreadyWateredException** – wyrzucany gdy naciśnięty zostanie przycisk „water” w przypadku podlanego trawnika dla którego nie nastąpiła zmiana warunków
- **ImproperCycleNumberException** – wyrzucany gdy wprowadzona liczba cykli nie jest liczbą naturalną dodatnią
- **ImproperRadiusException** – wyrzucany gdy wprowadzona wartość promienia podlewaczki360 nie jest liczbą naturalną dodatnią
- **InitializeFirstException** – wyrzucany w momencie podlania trawnika/generacji pliku SPRINKLERS lub BITMAP jeżeli nie został zaimportowany żaden trawnik

Wymienione wyjątki zostały podzielone na dwa pakiety – FileExceptions (pierwsze 5) i GUIExceptions (ostatnie 4)

## 6) Opisy wykorzystanych klas

Klasy nie związane z błędami umieszczono w 3 pakietach. Poniżej ich opisy, wartości i opisy ich zawartości:

1) **Property** – pakiet w którym znajdują się wszystkie klasy konieczne do zarządzania trawnikiem

- **Exporter** – klasa odpowiedzialna za export pliku SPRINKLERS oraz BITMAP
- **Picasso8** – klasa odpowiedzialna za wspomaganie klasy Exporter przy tworzeniu bitmapy. Metody tej klasy są odpowiedzialne za pisanie właściwego pliku (Exporter jedynie wywołuje jej metody)
- **Planner** – klasa odpowiedzialna za podział trawnika na obszary
- **Gardener** – klasa odpowiedzialna za komunikację z Plannerem i przeprowadzenie operacji podlania trawnika
- **Lawn** – klasa odpowiedzialna za przechowywanie trawnika oraz jego parametrów
- **LawnReader** – klasa odpowiedzialna za import trawnika z pliku
- **Sprinkler** – klasa abstrakcyjna opisująca pola i metody wspólne dla wszystkich podlewaczek
- **Sprinkler90/180/270/360** – klasy odpowiedzialne za udostępnianie informacji o odpowiednich podlewaczkach i posiadające możliwość podlania trawnika

2) **GUI** – pakiet w którym znajdują się wszystkie klasy odpowiedzialne za graficzny interfejs użytkownika

- **WaterButton** – przycisk do podlania trawnika
- **RadiusPanel** – panel umożliwiający modyfikację promienia podlewaczki360
- **PrevStatusPanel** – panel wyświetlający status poprzedniego polecenia (**OK** w przypadku sukcesu, **ERROR** w przeciwnym razie)
- **MainFrame** – klasa odpowiedzialna za wyświetlanie całego GUI
- **ImportPanel** – panel umożliwiający wprowadzenie i zaimportowanie pliku reprezentującego trawnik
- **FileButton** – przycisk umożliwiający generację/aktualizację pliku SPRINKLERS
- **ErrorFrame** – okienko, które wyświetli się podczas błędu programu
- **CycleNumberPanel** – panel umożliwiający modyfikację liczbę cykli podlewaczki360

- **BounePanel** – panel umożliwiający włączenie i wyłączenie odbicia lustrzanego wody
- **BitmapButton** - przycisk umożliwiający generację/aktualizację bitmapy
- **AnimationTimePanel** – panel, który miałby zastosowanie gdyby program posiadał funkcjonalność animacji procesu podlania trawnika
- **AnimationButton** - przycisk, który miałby zastosowanie gdyby program posiadał funkcjonalność animacji procesu podlania trawnika
- **ButtonPanel** – panel którego elementami są BitiamButton, AnimationButton, WaterButton, FileButton

3) **AuxiliaryClasses** – pakiet klas wspomagających klasy z pakietu Property. Ich istnienie nie jest konieczne z punktu widzenia funkcjonalności programu, ale upraszcza strukturę kodu

- **Colour** – klasa reprezentująca kolor według modelu RGB
- **GardenersNote** – przechowuje informacje o najlepszej możliwej podlewaczce do podlania danego prostokąta
- **Point** – klasa reprezentująca punkt. Każdy obszar jest charakteryzowany przez punkt
- **Rectangle** – klasa reprezentująca prostokąt