

JavaGridGraph - Dokumentacja

Skoczek Mateusz, Jędrzejewski Sebastian

24 maja 2022

Streszczenie

Dokument zawiera specyfikację funkcjonalną i implementacyjną dotyczącą projektu *JavaGrid-Graph* oraz opis testów programu.

Spis treści

1	Specyfikacja funkcjonalna	2
1.1	Cel projektu	3
1.2	Opis funkcji	4
1.3	Opis interfejsu programu	5
1.3.1	Okno główne - widok startowy	5
1.3.2	Okno generowania grafu	5
1.3.3	Okno główne - po wczytaniu grafu	6
1.4	Format danych wejściowych i wyjściowych	8
2	Specyfikacja implementacyjna	9
2.1	Diagram klas	10
2.2	Interfejs (Pakiet App)	11
2.2.1	App	12

Rozdział 1

Specyfikacja funkcjonalna

1.1 Cel projektu

Program **JavaGridGraph** ma na celu umożliwić wykonanie dwóch podstawowych zadań

- wygenerowanie grafu siatkowego o podanych paramentrach
- sprawdzenie wybranych parametrów dowolnego grafu siatkowego

Program posiada interfejs graficzny. Grafy są przedstawiane w plikach w postaci listy sąsiedztwa.

1.2 Opis funkcji

Program oferuje dwie główne funkcje: generowanie grafu oraz sprawdzanie grafu.

Program pozwala wygenerować graf o:

- określonej wysokości (ilości wierszy)
- określonej szerokości (ilości kolumn)
- określonej minimalnej i maksymalnej wadze krawędzi
- określonej minimalnej i maksymalnej ilości krawędzi wychodzących z pojedynczego wierzchołka (dla grafu skierowanego)
- określonej minimalnej i maksymalnej ilości krawędzi wchodzących do pojedynczego wierzchołka (dla grafu skierowanego)
- określonej minimalnej i maksymalnej ilości "sąsiadów" pojedynczego wierzchołka (dla grafu nieskierowanego)
- niestandardowym ziarnie generatora liczb losowych

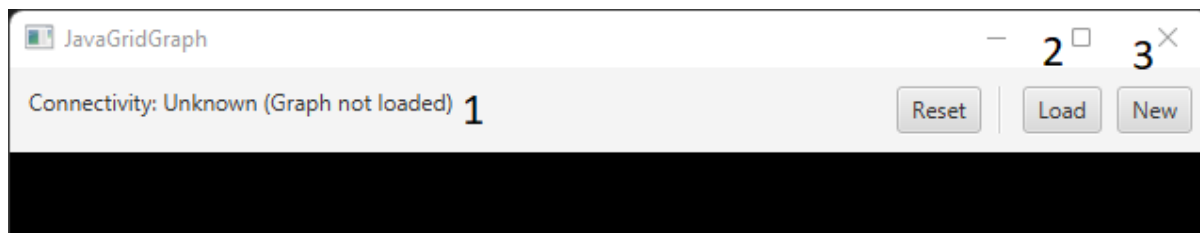
Program umożliwia zapis wygenerowanego grafu do pliku oraz/lub wczytanie grafu do sprawdzenia.

W ramach funkcji sprawdzania grafu, program pozwala na sprawdzenie następujących parametrów wczytanego grafu:

- spójność grafu
- najkrótsze ścieżki od wybranego wierzchołka A do wybranych wierzchołków B_n

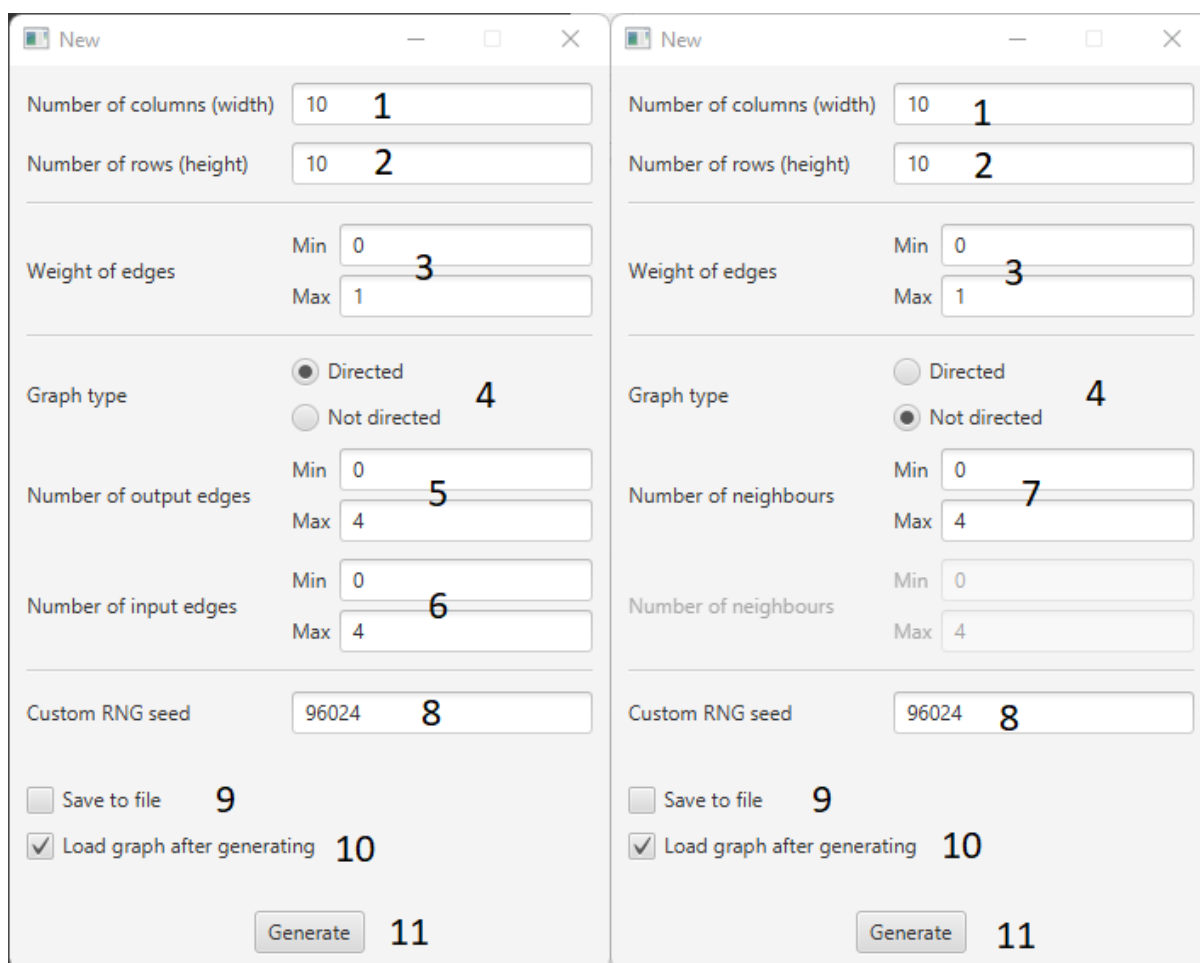
1.3 Opis interfejsu programu

1.3.1 Okno główne - widok startowy



1. Wynik sprawdzenia spójności grafu. W przypadku gdy graf nie został wczytany, zostanie pokazana informacja o tym że graf nie został wczytany.
2. Przycisk "Open" otwiera systemowe okno wyboru pliku w celu wybrania pliku zawierającego graf.
3. Przycisk "New" otwiera okno generowania grafu. Więcej informacji w punkcie "Okno generowania grafu".

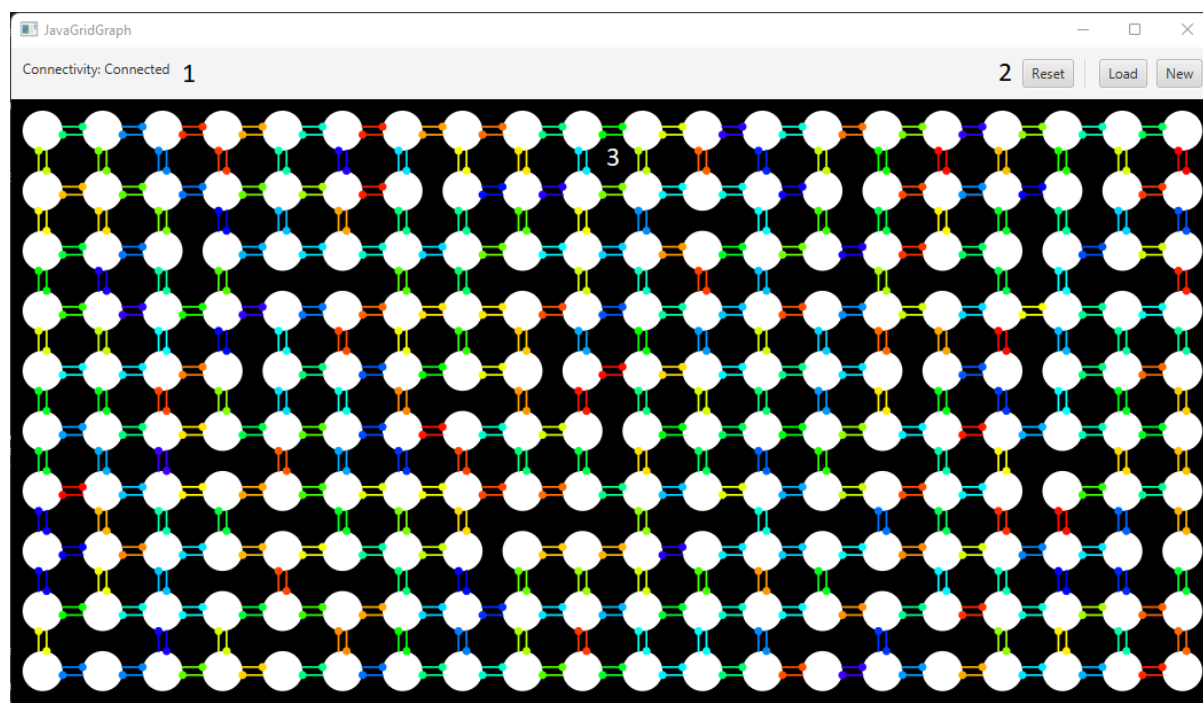
1.3.2 Okno generowania grafu



1. Liczba kolumn (szerokość) grafu. Pole nie może być puste.
2. Liczba wierszy (wysokość) grafu. Pole nie może być puste.

3. Waga krawędzi. Wartość w pierwszym polu (min, domyślnie 0) musi być mniejsza bądź równa wartości w drugim polu (max, domyślnie 1).
4. Wybór między generowaniem grafu skierowanego (directed), generowaniem grafu nieskierowanego (not directed).
5. Liczba krawędzi wychodzących z pojedynczego wierzchołka. Wartość w pierwszym polu (min, domyślnie 0) musi być mniejsza bądź równa wartości w drugim polu (max, domyślnie 4). [Tylko gdy zaznaczone generowanie grafu skierowanego]¹
6. Liczba krawędzi wchodzących do pojedynczego wierzchołka. Wartość w pierwszym polu (min, domyślnie 0) musi być mniejsza bądź równa wartości w drugim polu (max, domyślnie 4). [Tylko gdy zaznaczone generowanie grafu skierowanego]¹
7. Liczba sąsiadów pojedynczego wierzchołka. Wartość w pierwszym polu (min, domyślnie 0) musi być mniejsza bądź równa wartości w drugim polu (max, domyślnie 4). [Tylko gdy zaznaczone generowanie grafu nieskierowanego]¹
8. Niestandardowe ziarno generatora liczb losowych.
9. Zaznaczenie tego pola spowoduje zapisanie grafu w pliku wybranym w systemowym oknie zapisu pliku, które wyświetli się po naciśnięciu przycisku "Generate".
10. Zaznaczenie tego pola spowoduje wczytanie grafu do programu zaraz po jego wygenerowaniu.
11. Kliknięcie przycisku spowoduje wygenerowanie grafu o podanych parametrach.

1.3.3 Okno główne - po wczytaniu grafu



¹Program będzie dążył do utworzenia co najmniej minimalnej liczby krawędzi (połączeń do sąsiadów), ale nie może tego zagwarantować. Nie jest możliwe wygenerowanie więcej niż 2 krawędzi dla wierzchołków w narożnikach oraz więcej niż 3 dla wierzchołków bocznych. Nie jest możliwe także utworzenie krawędzi, jeżeli wszystkie wierzchołki wokół osiągnęły już swoją nominalną (wylosowaną z podanego przedziału) liczbę krawędzi.

1. Wynik sprawdzenia spójności grafu. "Connected" - spójny, "Unconnected" - niespójny.
2. Kliknięcie przycisku spowoduje odznaczenie wszystkich zaznaczonych wierzchołków na grafie.
3. Pole w którym wyświetlany jest graf. Aby znaleźć najkrótsze ścieżki musimy wybrać wierzchołek A lewym przyciskiem myszy (zostanie zaznaczony na czerwono) oraz wierzchołki B_n prawym przyciskiem myszy (zostaną zaznaczone na zielono). Po wybraniu wierzchołka A oraz przynajmniej jednego wierzchołka B zostanie narysowana najkrótsza ścieżka od wierzchołka A do wierzchołka B_n .

1.4 Format danych wejściowych i wyjściowych

Dane wejściowe i wyjściowe przechowują graf w postaci listy sąsiedztwa. W pierwszej linii znajdują się dwie liczby, które oznaczają odpowiednio liczbę kolumn i wierszy danego grafu. Każda następna linijka reprezentuje jeden wierzchołek, przy czym wierzchołki numerujemy od 0 od lewej do prawej. Zatem druga linijka w pliku zawiera numery wierzchołków, z którymi połączony jest wierzchołek numer 0, kolejna dotyczy wierzchołka numer 1 itd. Przy każdym numerze wierzchołka po dwukropku podana jest waga krawędzi pomiędzy tymi dwoma wierzchołkami.

Przykład:

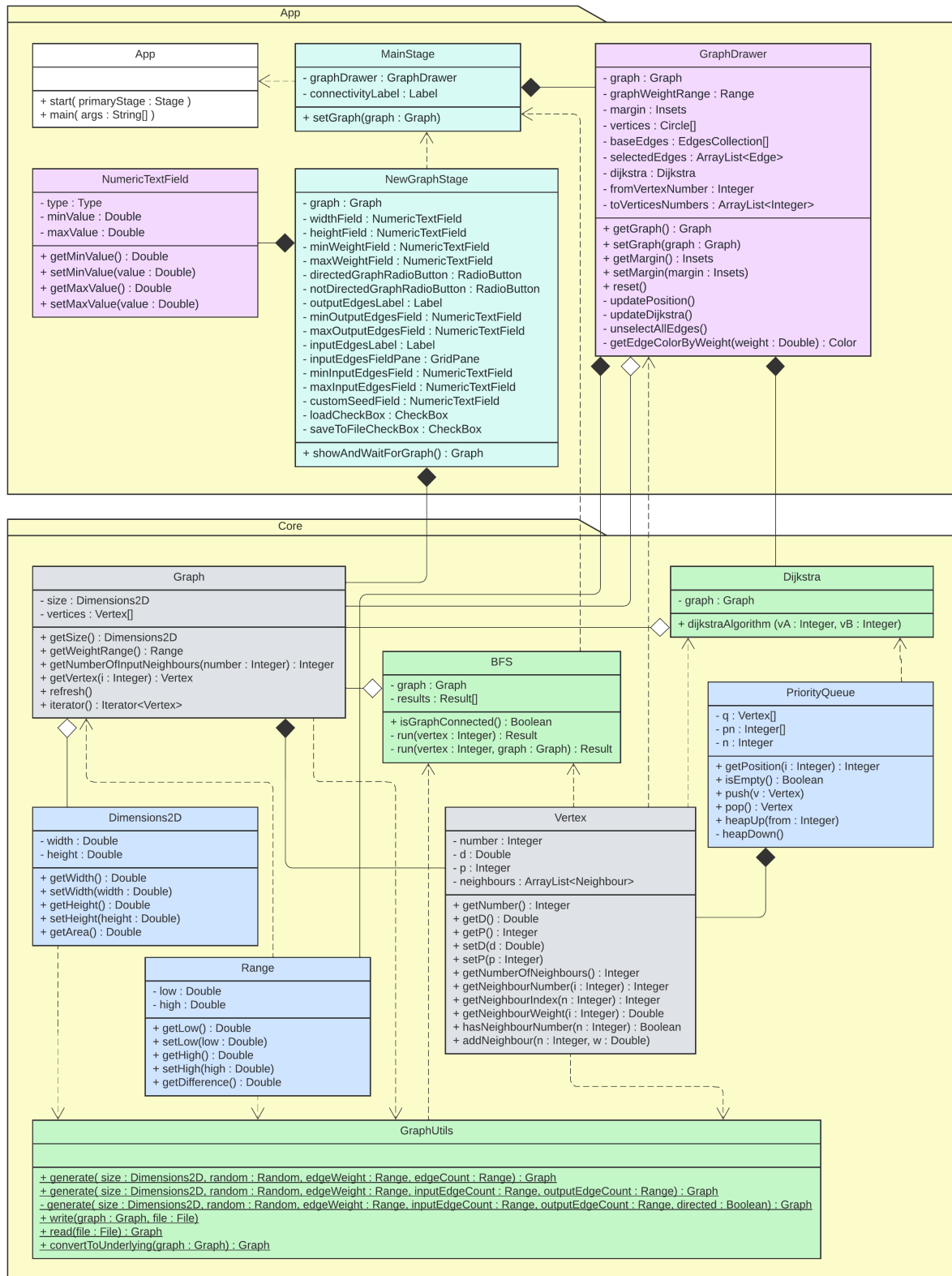
```
2 2
1 :0.54 2 :0.78
0 :0.54 3 :0.12
0 :0.78 3 :0.89
1 :0.12 2 :0.89
```

Powyżej przedstawiona jest przykładowa zawartość pliku przechowującego graf. W pierwszej linii można odczytać, że jest to graf o dwóch kolumnach i dwóch wierszach. W drugiej linii przedstawiona jest informacja o tym, że wierzchołek numer 0 połączony jest z wierzchołkiem numer 1, a krawędź ta ma wagę 0.54. Istnieje również krawędź pomiędzy wierzchołkiem 0 a 2 o wadze 0.78. W trzeciej linii znajdują się numery wierzchołków połączonych z wierzchołkiem numer 1 wraz z wagami itd.

Rozdział 2

Specyfikacja implementacyjna

2.1 Diagram klas



2.2 Interfejs (Pakiet App)

Interfejs aplikacji został stworzony przy wykorzystaniu biblioteki JavaFX. Pakiet **App** składa się z dwóch podpakietów:

- **Controls** zawierającym niestandardowe kontrolki
- **Stages** zawierającym widoki aplikacji

2.2.1 App