POLITECHNIKA WARSZAWSKA
WYDZIAŁ ELEKTRONIKI I TECHNIK INFORMACYJNYCH
TELEKOMUNIKACJA

PRM2T

PROJEKT NUMBERLINK
ETAP 3

Autorzy:
Marcin Szymosz 311467
Adam Wiśniewski 311484
Agata Zatorska 325338
Anh Quan Do 325268
Prowadzący:
Dr. mgr inż. Grzegorz Galiński

Spis treści

1 Wstęp

Zadanie projektowe obejmuje napisanie kompletnego programu w Javie umożliwiającego rozwiązywania gry Numberlink. W jego skład wchodzi generator plansz, solwer oraz interfejs graficzny. Ponadto program umożliwia cofanie ruchów, zapis aktualnego stanu gry do pliku i jego późniejszy odczyt oraz zapis planszy do pliku umożliwiającego wydruk łamigłówki.

2 Kod

Nasz kod składa się z trzech klas: ArrayMapGenerator, NumberlikGame oraz Victory.

2.1 ArrayMapGenerator

Klasa ArrayMapGenerator służy do generowania tablic, które następnie używane są przez program do wyświetlania ich w formie plansz do gry. Na początku tablice generowane są razem z odpowiedziami, a dopiero po stworzeniu tablicy usuwane są połączenia między liczbami.
Opis metod użytych w klasie:

- ArrayMapGenerator() - Konstruktor, tworzący NxN wymiarową tablicę wypełnioną zerami.
- inArea() - Sprawdza czy dany punkt znajduje się w obszarze tablicy naszej tablicy "map".
- isEmpty() - sprawdza czy dany punkt na mapie jest pusty (wartość równa 0). Przyjmuje obiekt typu Point oraz dwuwymiarową listę reprezentującą mapę.
- areNeighboursEmpty() - Sprawdza czy sąsiedzi danego puntu na mapie są pustymi polami. Przyjmuje obiekt typu Point, dwuwymiarową listę reprezentującą mapę oraz zmienną currentNumber, określająca, która z linii jest aktualnie tworzona.
- whichNeighboursEmpty() - Zwraca listę "wynik" sąsiednich punktów, które są puste i mają inną wartość niż currentNumber.
- isSameNumber() - Sprawdza czy sąsiedni punkt ma tę samą liczbę, co ta oznaczająca aktualnie tworzoną linię.
- areNeighboursOfNeighbourEmpty() - Sprawdza czy sąsiednie punkty danego sąsiada są puste w celu
- threeInARow() - Sprawdza czy istnieją trzy puste pola obok siebie. Jest to używane dalej w programie, żeby zapobiec tworzeniu się punktów początkowych, które nie mają miejsca na swoją kontynuację.
- isAvailable() - Sprawdza czy istnieją wolne miejsca na mapie, na których można by postawić punkt początkowy dla linii.
- generateMap() - Generuje mapę na podstawie poprzednich metod i dostępnych miejsc.
- listForGUI() - Przygotowuje listę bareMap, która jest wykorzystywana do stworzenia planszy do wyświetlania dla użytkownika bez linii.
- writeBareMap() - Zapisuje bareMap (tablica bez rozwiązania) do pliku w folderze resource.
- writeMap() - Zapisuje Map (tablica z rozwiązaniem) do pliku w folderze resource.
- readBareMap() - Odczytuje bareMap z folderu resource.
- readMap() - Odczytuje Map z folderu resource.

2.2 NumberlinkGame

Klasa odpowiadająca za tworzenie planszy oraz interfejsu graficznego wraz z funkcjami, które są aktywowane w momencie użycia przycisków. Opis metod użytytych w klasie:

- createColorButtons() - tworzy przyciski aktywujące kolory zaznaczania w panelu na górze okna oraz tworzy przyciski funkcyjne na dole okna(wraz z ich actionListener
- checkWin() - funkcja sprawdzająca czy plansza została poprawnie rozwiązana
- setNumbers() - funkcja ustawia kolory startowe na planszy
- captureScreenshot() - funkcja robi zrzut ekranu aktualnego stanu gry i zapisuje do pliku .png

2.3 Victory

Klasa w momencie wywołania tworzy nowe okno informujące o poprawnym rozwiązaniu planszy.

3 Funkcjonalności

- Tworzenie losowej mapy do rozwiązania
- Wybieranie aktualnie używanego koloru
- Możliwość zrobienia zrzutu ekranu aktualnej planszy
- Możliwość zapisu aktualnego stanu gry
- Odczytanie zapisu gry
- Sprawdzenie czy rozwiązaliśmy planszę poprawnie

4 Wnioski

Program działa poprawnie i udało nam się umieścić w nim wszystkie założenia.