

Gymnázium, Praha 6, Arabská 14,
předmět Programování, vyučující Mgr. Jan Lána



Ročníkový projekt

1.E Radim Tříletý

Hledání min

Květen 2021

Čestné prohlášení:

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Hostivících dne 2. 5. 2021

Radim Tříletý

Anotace:

V tomto ročníkovém projektu je předkládán program v programovacím jazyce Java. Cílem mé práce bylo naprogramovat hru Hledání min, hra spočívá v tom že odkrýváte pole tlačítek, pod tlačítkem je mina, nebo číslo udávající počet min v okolí. Program je napsán pomocí grafického rozhraní JavaFX, které je součástí programovacího jazyka Java.

Annotation:

This project presents a program in the Java programming language. The aim of my work was to program the game Minesweeper, the game is that you uncover a field of buttons, below the button is a mine, or a number indicating the number of mines in the area. The program is written using the JavaFX graphical interface, which is part of the Java programming language.

Zadání ročníkového projektu

Popis:

Hledání min je hra pro jednoho hráče. Hráč postupně odhaluje pole, které jsou skrytá. Po odhalení se buď objeví číslo, které znázorňuje počet min v okolí tohoto čísla nebo se objeví mina a v tu chvíli hráč prohrává. Hráč má za úkol odkrýt všechna čísla.

Platforma:

Java
JavaFX

Obsah

1. Úvod.....	6
2. Uživatelská dokumentace.....	7
2.1. Bližší popis hry minesweeper.....	7
2.2. Moje verze hry.....	7
2.3. Uživatelské problémy.....	7
3. Programátorská dokumentace.....	8
3.1. Grafické rozhraní.....	8
3.2. Struktura kódu.....	9
3.3. Problém.....	10
4. Závěr.....	11
5. Citace.....	12

1. Úvod

Tento dokument se zabývá popisem mé ročníkové práce s úkolem naprogramovat Hledání min dle předem schváleného stručného zadání. Práce je rozdělena do dvou hlavních kapitol: Uživatelská a Programátorská Dokumentace.

Uživatelská Dokumentace se zabývá hrou z pohledu uživatele, který hru spustí (pravidla, ukázky ze hry atd.)

V programátorské dokumentaci popisují kód programu a problémy, na které jsem narazil při psaní programu.

2. Uživatelská dokumentace

2.1. Bližší popis hry Hledání min

Cílem hry je odhalit všechna políčka obsahující miny, nebo ještě přesněji, odkrýt všechna políčka, která minu neobsahují. To, jestli nějaké neodkryté políčko minu obsahuje, či nikoliv, se hráč dozví tak, že na ně klepne levým tlačítkem myši. Pokud políčko minu neobsahuje, objeví se číslo, které ukazuje, kolik min se celkem nachází na osmi políčkách s ním sousedících.

Je-li zřejmé, že některé neodkryté políčko obsahuje minu, lze jej pro větší přehlednost označit vlajkou, kterou hráč přidá podržením tlačítka myši na dané políčko.

Hra končí vítězstvím, pokud se podaří odkrýt všechna políčka neobsahující minu, nebo prohrou v případě odkrytí tlačítka s minou.

2.2 Moje verze hry

Moje verze hry je v základu stejná jako všechny ostatní hry typu Hledání min, menší rozdíly jsou ve velikosti pole a množství min. Větší rozdíl je v tom, že jsem nedokázal vymyslet náhodné generování min na políčkách, a tak by se v zásadě dalo říci, že můžete hrát jenom jednu úroveň.

2.3. Uživatelské problémy

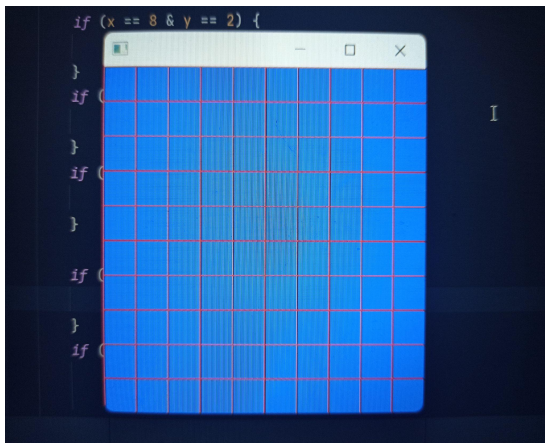
Jak jsem, již psal výše, tak moje verze hry má jenom jednu úroveň. Nemůže se tedy hrát opakovaně, když ji dohrajete. Není stvořena pro uživatele, který nezná pravidla hry Hledání min. Hra taktéž nemá žádnou výherní obrazovku.

3. Programátorská dokumentace

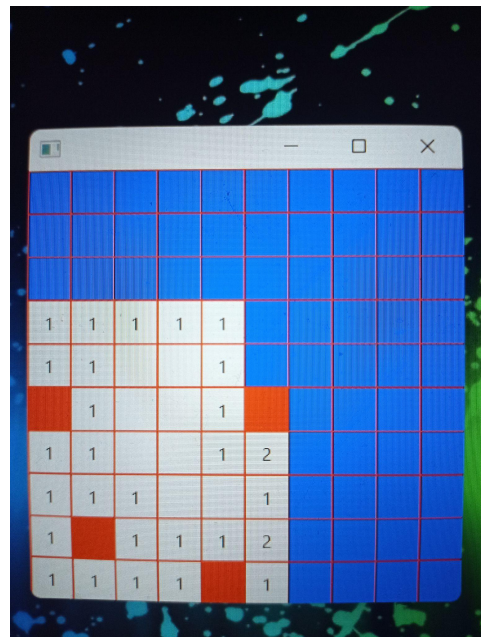
3.1. Grafické rozhraní

Grafické rozhraní bylo vepsáno rovnou v kódu celé aplikace a jenom kousek programu v JavaFX Scene Builder.

Hlavní okno je čtvercového tvaru. S hlavním oknem lze hýbat po ploše a měnit jeho rozměr, ale velikost políček se velikosti okna nepřizpůsobuje, tudíž zůstávají stále stejně velké. Na obrázku 1 můžete vidět, jak vypadá aplikace, když ji spustíte. A na obrázku 2, když ji hrajete.



Obrázek 1 - aplikace , když ji spustíme



Obrázek 2 - aplikace, když ji hrajeme

3.2 Struktura kódu

Kód má pouze jedno celistvé tělo.

První funkce `static Parent createContent()` společně s kouskem třídy `static class Tile extends StackPane` vytváří pole čtverečků, za kterými se schovávají čísla nebo miny.

Zde je:

```
static Parent createContent() {
    Pane root = new Pane();
    int y;
    int x;
    for (y = 0; y < sloupec; y++) {
        for (x = 0; x < radek; x++) {
            Tile tile;
            tile = new Tile(x, y);
            root.getChildren().add(tile);
        }
        return root;
    }
    static class Tile extends StackPane {
        private final Text text = new Text();
        boolean i = true;
        final Rectangle border = new Rectangle(30,30);
        .....
        border.setFill(Color.BLUE);
        text.setVisible(false);
        border.setStroke(Color.RED);
        getChildren().addAll(border, text);
        setTranslateX(x * 30);
        setTranslateY(y * 30);
    }
```

Dále následuje funkce, která dává polím určité hodnoty, které mám napsané `if-ama`, který určují jestli `boolean i` je `true` nebo `ne`. Jestliže je `true` tak se na dané souřadnice (`x` je souřadnice pro políčko v řádku a `y` je pro políčko ve sloupci) vytvoří mina nebo číslo.

Příklad jednoho `ifu` kde se nachází mina:

```
if (x == 0 & y == 0) {
    i = false;
}
```

A jednoho kde se nachází číslo:

```
if (x == 0 & y == 2) {
    i = true;
    text.setText("2");
}
```

Poté, když uživatel klikne na políčko tak funkce `setOnMouseClicked` spustí `public void otevrit` kde se buď odhalí zakliknuté políčko, když se jedná o číslo, nebo se objeví obrazovka “Prohrál jsi!” a hra se resetuje. Obrazovku “Prohrál jsi!” jsem udělal pomocí scene builderu.

Uživatel má možnost také podržet políčko, a tím se mu ono políčko označí.

Zde je `public void otevrit`:

```
public void otevrit(Stage stage) throws IOException {

    if (i) {
        text.setVisible(true);
        border.setFill(null);
    }
    if (!i) {

        Parent root =
FXXMLLoader.load(Objects.requireNonNull(getClass().getResource(
"hello-view.fxml")));
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
        getScene().setRoot(createContent());
    }
}
```

3.3. Problém

Narazil jsem na spoustu problémů při programování této hry. Nejhlavnější z nich, který jsem bohužel nedokázal vyřešit, bylo generování čísel podle toho, kolik je okolo min. Proto jsem musel miny i čísla umístit na jednotlivá políčka a nemohu je již měnit.

4. Závěr

Závěrem bych napsal, že zadání jsem nesplnil na 100%. Aplikace funguje, ale nestihl jsem implementovat vše, co jsem chtěl a tudíž má práce nesplňuje můj původní plán, který jsem si v úvodu vymyslel a to je, že hra negeneruje miny náhodně. Jednou z příčin, že hra nefunguje, jak jsem na začátku zamýšlel, bylo i špatné rozvržení mého časového plánu, tzn. začal jsem pracovat později než jsem měl. Rozhodně se chci do budoucna poučit a příští zadaný úkol začnu řešit mnohem dřív, tedy s předstihem.

I tak si myslím, že byl pro mne projekt přínosný.

5. Citace

Malé části kódu, nebo jen nápad jsem čerpal z těchto stránek:

<https://jenkov.com/tutorials/javafx/index.html>

<https://minesweeper.online/> <https://docs.oracle.com/en/java/>

<https://www.tutorialspoint.com/how-to-add-an-image-as-label-using-javafx>

<https://stackoverflow.com/questions/35512648/adding-a-timer-to-my-program-javafx>

<https://zetcode.com/javagames/minesweeper/>

https://www.youtube.com/watch?v=JwcyxuKko_M&t=130s

<https://github.com/gyarab/2020-1e-bittner-matej-minesweeper>

https://www.youtube.com/watch?v=DvqURR_XN5U