Gymnázium, Praha 6, Arabská 14
Arabská, Praha 6, 160 00

ROČNÍKOVÝ PROJEKT

Předmět: Programování
Téma: Společenská hra Solitaire

Autor: Nikola Jankovič
Třída: 1.E
Školní rok: 2021/2022
Vyučující: Mgr. Jan Lána
Třídní učitel: Mgr. Blanka Hniličková

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 Praze dne: .....

Nikola Jankovič

Anotace:

Pro tuto ročníkovou práci jsem si připravil program v programovacím jazyce Java. Program je desková hra s názvem Solitaire (nezaměnit se stejnojmennou karetní hrou). Hra funguje na podobném principu jako populární desková hra dáma. Solitaire se na rozdíl od dámy hraje pouze v jednom hráči. Hráč má za úkol postupně přeskakovat žetony dokud nezbyde na desce pouze jeden žeton.

Annotation (English):

For this coursework I have prepared a programme in the Java programming language. The programme is a board game named Solitaire (different from the card game Solitaire). The game follows a similar principle to checkers. The main difference between the two games is that Solitaire is only played by one player. The player's goal is to remove pegs one by one until there is only one peg left on the board.

Zadání ročníkového projektu:

Vytvořte hru Solitaire, kde se při zapnutí objeví žetony v předurčeném rozhraní. Hráč pak bude moci přetahovat kurzorem jednotlivé žetony a postupným přeskakováním žetony vyhazovat dokud na desce nezbude jenom jeden žeton.

Možné vylepšení do budoucna:

- Přidat novou scénu pokud hráč vyhraje nebo prohraje
- Změnit vzhled rohů hracího pole, protože se v nich nemohou nacházet žetony
- Zlepšit vzhled žetonů

Platforma:

- Java
- JavaFX

Obsah

Úvod

Tato dokumentace pojednává o hře Solitaire napsané v programovacím jazyce Java. Cílem hry je přeskakovat postupně přeskakovat žetony, dokud nezbyde jenom jeden.

Toto téma mi zvolil pan profesor Mgr. Jan Lána, jelikož jsem v době vybírání témat nebyl ve škole, z důvodu vážného úrazu pravé spodní končetiny. Jsem názoru, že téma mi vybral dobré a přiměřeně obtížné.

Program jsem napsal v programovacím jazyce Java a vývojovém prostředí IntelliJ IDEA, který používáme i na hodinách Programování.

1. Programovací jazyk a vývojové prostředí

Pro napsání předkládaného programu jsem použil Javu 11 a vývojové prostředí IntelliJ IDEA.

1.1. Java

Java je programovací jazyk vyvinutý firmou Sun Microsystems v roce 1995. Do roku 2020 byla Java nejpoužívanější programovací jazyk v roce 2020 jí předběhl Python a C. Velkou výhodou Javy je dostupnost. Aplikace vytvořené v programovacím jazyce Java se dají bez problému spustit na jakémkoliv zařízení bez ohledu na operační systém zařízení.

Obrázek 1 - Duke maskot Javy

1.2. JavaFX

JavaFX je rozšíření Javy které umožňuje tvoření grafických aplikací. FX je zkratka značící slovo effects.

1.3. IntelliJ IDEA

IntelliJ IDEA je komerční vývojové prostředí vynalezeno firmou JetBrains. Zároveň existuje bezplatná Community verze. Tato verze je open-source a využil jsem jí k napsání mého kódu.

Obrázek 2 - Logo IntelliJ IDEA

2. Solitér

2.1. Solitér a Pasiáns

Pod hrou Solitér (v překladu samotář) si většina lidí představí karetní hru pro jednoho hráče. Tato hra se oficiálně nazývá Pasians (v překladu trpělivost). Název solitér se v USA a používá pro spoustu karetních her pro jednoho hráče.

2.2. Solitér

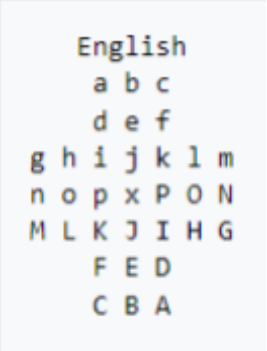
Cílem hry solitér je postupně legálními tahy vyřazovat jednotlivé žetony, dokud na desce nezbyde pouze jeden žeton. Tradičně existují dvě počáteční desky, které se liší v rozmístění žetonů. Anglická a evropská deska (viz. obrázek 3).

Obrázek 3 - Anglická deska (nalevo) Evropská (napravo)

Hra se dá hrát jak s diagonálními tahy tak bez. V mé verzi hry používám anglickou desku a diagonální tahy jsou legální. V případě evropské desky s počátečním volným polem uprostřed se hra nedá bez diagonálních tahů vyhrát.

2.3. Strategie

Existuje mnoho různých řešení standardního problému a jeden zápis používaný k jejich popisu přiřazuje dírám písmena(viz. obrázek 4):



Obrázek 4 - popsání jednotlivých polí

*Na následujících obrázcích · označuje žeton v díře, zvýrazněná * označuje žeton, který má být přemístěn, a o označuje prázdné pole. Modrá ¤ je prázdné pole, ze kterého se aktuální žeton přesunul; červená * je konečná pozice tohoto žetonu, červené o je volné pole po žetonu, který byl přeskočen a odstraněn.*[1]

Obrázek 5 - Nejkratší možné řešení anglické desky

Toto řešení bylo nalezeno v roce 1912 Ernestem Bergholtem a v roce 1964 John Beasley dokázal, že je to nejkratší možné řešení.[2]

3. Program

3.1. Grafické rozhraní

Na grafické zobrazení jsem nevyužíval scene builder, ale psal jsem to jako normální kód do JavaFX.

```
Ellipse bg = new Ellipse( v: tileSize * 0.3125,  v1: tileSize * 0.26);
bg.setFill(Color.BLACK);

bg.setStroke(Color.BLACK);
bg.setStrokeWidth(tileSize* 0.03);

bg.setTranslateX((tileSize - tileSize * 0.3125 * 2 ) / 2);
bg.setTranslateY((tileSize - tileSize * 0.26 * 2 ) / 2 + tileSize * 0.07);

Ellipse ellipse = new Ellipse( v: tileSize * 0.3125,  v1: tileSize * 0.26);
ellipse.setFill(type == PieceType.RED ? Color.valueOf("#c40003") : Color.valueOf("#fff9f4"));

ellipse.setStroke(Color.BLACK);
ellipse.setStrokeWidth(tileSize* 0.03);

ellipse.setTranslateX((tileSize - tileSize * 0.3125 * 2 ) / 2);
ellipse.setTranslateY((tileSize - tileSize * 0.26 * 2 ) / 2);

getChildren().addAll(ellipse);
```

Obrázek 6 - Grafické zobrazení žetonu

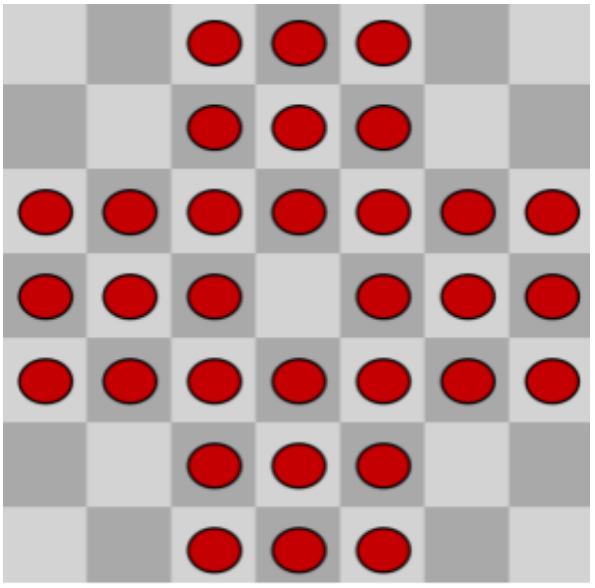
Pro vytvoření žetonu jsem využil dvou elips jedno pro vytvoření samotného žetonu (ellipse) a druhou pro obrys (bg). Zároveň jsem si vytvořil žetony červené a bíle, které se dají v kódu během pár vteřin zaměnit. Dle mého názoru červené vypadají lépe.

3.2. Vložení žetonu na desku

Pro vložení žetonů na desku používám velmi jednoduchou a krátkou metodu makePiece, kterou budu využívat i při přesunu jednotlivých žetonů. Poté už stačilo napsat pár if podmínek aby se žetony vložili na správné místa (viz. obrázek 7).

```
if (x > 1 && y == 0 && x < 5) {
    peg = makePiece(PieceType.RED, x, y);
}
if (x > 1 && y == 1 && x < 5) {
    peg = makePiece(PieceType.RED, x, y);
}
if (y == 2) {
    peg = makePiece(PieceType.RED, x, y);
}
if (y == 3 && x != 3) {
    peg = makePiece(PieceType.RED, x, y);
}
if (y == 4) {
    peg = makePiece(PieceType.RED, x, y);
}
if (x > 1 && y == 5 && x < 5) {
    peg = makePiece(PieceType.RED, x, y);
}
if (x > 1 && y == 6 && x < 5) {
    peg = makePiece(PieceType.RED, x, y);
}
```

Obrázek 7 - podmínky pro vložení žetonů



Obrázek 8 - výchozí deska

3.3. Pohyb žetonů

Asi nejtěžší část samotného programu byli pravidla samotného pohybu. Prvně jsem si vytvořil dva výsledky pohybu. NONE pokud pohyb není legální a KILL pro legální pohyb. Pohyb je legální pouze pokud se žeton posune přesně o 2 pole a přeskočí při pohybu nějaký jiný žeton (viz Obrázek 9).

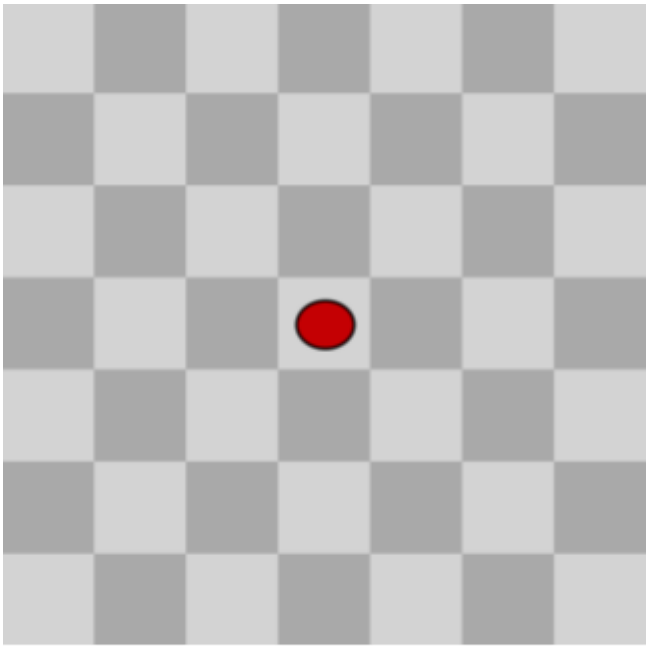
```
if (Math.abs(newX - x0) == 2 || Math.abs(newY - y0) == 2) {

    int x1 = x0 + (newX - x0) / 2;
    int y1 = y0 + (newY - y0) / 2;

    if (board[x1][y1].hasPeg()) {
        return new MoveResult(MoveType.KILL, board[x1][y1].getPeg());
    }
}
```

Obrázek 9 - podmínky pro pohyb s výsledkem KILL

Pohyb je nelegální pokud by výsledná pozice žetonu byla mimo hrací plochu nebo by na výsledném poli už byl žeton(viz. Obrázek 10).

```
if (board[newX][newY].hasPeg()) {
    return new MoveResult(MoveType.NONE);
}
if (Math.abs(newX) < 2 && Math.abs(newY) < 2) {
    return new MoveResult(MoveType.NONE);
}
if (Math.abs(newX) > 4 && Math.abs(newY) < 2) {
    return new MoveResult(MoveType.NONE);
}
if (Math.abs(newX) > 4 && Math.abs(newY) > 4) {
    return new MoveResult(MoveType.NONE);
}
if (Math.abs(newX) < 2 && Math.abs(newY) > 4) {
    return new MoveResult(MoveType.NONE);
}
```

Obrázek 10 - podmínky pro pohyb s výsledkem NONE



Obrázek 11 - Finální výherní pozice

3.4. Instalace programu

Program se pouští přímo z kódu a není potřeba žádná speciální instalace.

Závěr

V tomto ročníkovém projektu jsem vytvořil deskovou hru Solitér. Myslím si, že přestože jsem si téma nevybral sám, tak mě bavilo a bylo přiměřeně obtížné pro první ročník.

Citace

1. Peg solitaire - Wikipedia. [online]. Dostupné z: https://en.wikipedia.org/wiki/Peg_solitaire
2. Berlekamp, E.R., Conway, J.H., Guy, R.K. 2003. Winning ways for your mathematical plays. Taylor and Francis. Michigan. 1004 s. ISBN: 1568811446

Seznam obrázků