

Gymnázium, Praha 6, Arabská 14

Programování

ROČNÍKOVÁ PRÁCE



2022

Jana Šrámková

Arabská 14, Praha 6, 160 00

ROČNÍKOVÁ PRÁCE

Předmět: Programování

Téma: Textová hra

Autorka: Jana Šrámková

Třída: 2. E

Školní rok: 2021/2022

Vedoucí práce: Mgr. Jan Lána

Třídní učitel: PhDr. Markéta Šlegerová

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 dne

Anotace

Program je ve stylu prvních *Choose your own adventure* textových her. Příběh je prezentovaný jako text a hráč v určitých úsecích udělá rozhodnutí, které ovlivní další běh hry. Textové hry mohou mít mnoho podob, v případě tohoto programu se jedná o příběhovou hru s několika možnými konci.

Zadání

Hra bude mít krátký lineární příběh a hráč si bude v určitých chvílích vybírat, jak chce na určitou situaci zareagovat. Tato rozhodnutí budou pak mít vliv na další běh příběhu. Ve hře bude systém životů a náhodné střety s monstry, které mohou hráče zabít.

Obsah

Anotace.....	4
Zadání.....	5
Obsah	6
Úvod.....	7
Grafické uživatelské rozhraní	7
Příkazy a tvorba	8
První verze.....	8
Druhá verze	8
Závěr.....	11
Zdroje	12

Úvod

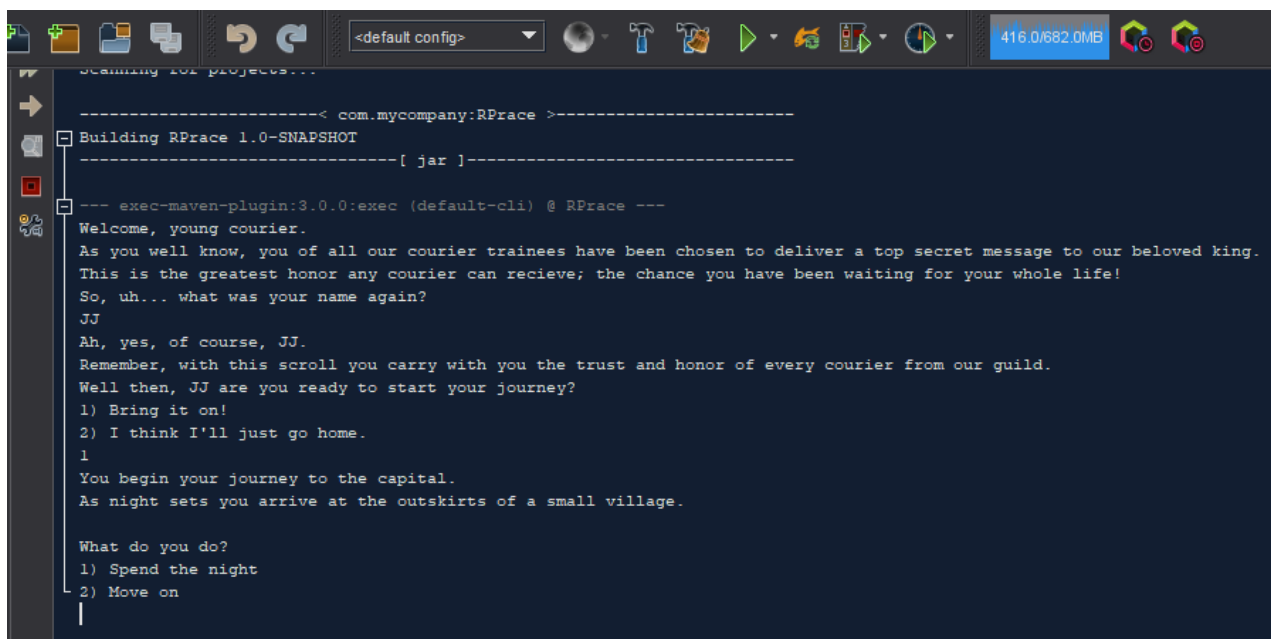
Tato práce je dokumentací ke tvorbě mojí textové hry. Jedná se o jednoduchou hru po vzoru nejstarších *Choose your own adventure* her, kde si hráč může vybírat reakce své postavy v dané situaci. Takové hry mají různé podoby, od nekonečných her, kde se stále dokola náhodně vybírají například místnosti, nebo monstra se kterými hráč bojuje, nebo jsou také hry s jednoduchým příběhem.

V případě mojí hry se jedná o jednoduchý lineární příběh s několika konci, ke kterým se hráč dostane podle svých rozhodnutí v průběhu hry.

Grafické uživatelské rozhraní

Celý program je naprogramovaný v Apache NetBeans IDE 12.6 v jazyku java a s použitím knihovny Java Swing.

V první fázi se stejně jako úplně první textové hry spouštěla přímo v uživatelském prostředí v základní konzoli. V konzoli se nejprve objeví text se součástí příběhu a volbou, kterou má hráč zrovna před sebou, a hráč pak jednoduše stisknutím kláves 1–3 a enter zvolí, kterou



```
Scanning for projects...
-----< com.mycompany:RPrace >-----
Building RPrace 1.0-SNAPSHOT
-----[ jar ]-----

--- exec-maven-plugin:3.0.0:exec (default-cli) @ RPrace ---
Welcome, young courier.
As you well know, you of all our courier trainees have been chosen to deliver a top secret message to our beloved king.
This is the greatest honor any courier can recieve; the chance you have been waiting for your whole life!
So, uh... what was your name again?
JJ
Ah, yes, of course, JJ.
Remember, with this scroll you carry with you the trust and honor of every courier from our guild.
Well then, JJ are you ready to start your journey?
1) Bring it on!
2) I think I'll just go home.
1
You begin your journey to the capital.
As night sets you arrive at the outskirts of a small village.

What do you do?
1) Spend the night
2) Move on
|
```

Obrázek 1: První verze hry

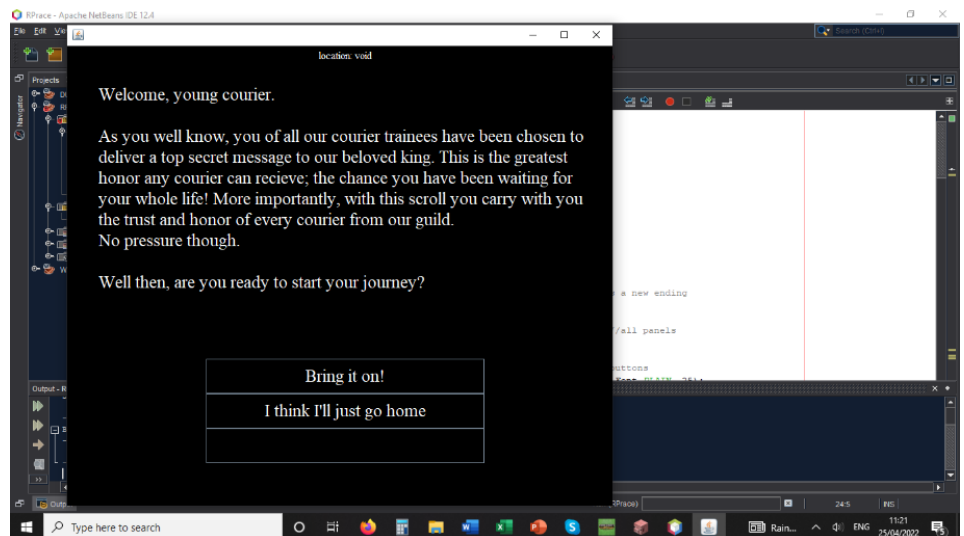
z nabízených možností by chtěl vybrat.

V této fázi šlo hlavně o to, vytvořit příběh a základní verzi hry, kterou by bylo možné použít v případě, že bych nestihla dodělat druhou verzi.

Ve druhé fázi pak šlo o to vytvořit okenní aplikaci, ve které by bylo příběh možné hrát s pomocí Jbuttonů.

Příkazy a tvorba

První verze



Obrázek 2 Druhá verze hry

První verze stála v podstatě na dvou druhích příkazů; **System.out.println** a **switch**.

System.out.println jednoduše vytiskne do konzole zadanou část příběhu.

Switch zpracovává rozhodnutí hráče s pomocí integeru *choice* a spustí následující část příběhu, která navazuje na hráčovo rozhodnutí.

```
public void auntMeet() throws InterruptedException { //second part of auntLetter
    System.out.println("Around noon you arrive at a small cottage at the outskirts of a small wood.");
    System.out.println("Girl: You smell funny. Like paper and ink. And 'sniff' is that ointment in ");
    System.out.println("I thought... Feyfey said his aunt lived alone?");
    System.out.println("2) Goodbye.");
    choice = sc.nextInt();

    switch (choice) {
        case 1 -> {
            System.out.println("The girl looks at you as if questioning your intelligence.");
            System.out.println("Girl: I do live alone, can't you see? Now give me that ointment, my ");
            System.out.println("The girl snatches the bottle from you and climbs back in her blanket ");
            System.out.println("Woman: Ah, much better! You know, Feyfey may be tad too overprotecti ");
            System.out.println("The woman forces a single sheet of ink black paper into your hands ");
            paper = 1;
            campfire();
        }
        case 2 -> {
            System.out.println("You turn on your heel and hightail it out of the cottage.\nAs you gl ");
            campfire();
        }
    }
}
```

Obrázek 3: Příkazy první verze

Druhá verze

Jako první jsem samozřejmě vytvořila samotné okno (JPanel) s rozměry 800 na 700, což bylo na daný účel tak akorát. Následně jsem vytvořila několik metod jako předlohy pro různé herní

prvky jako JButton, textové pole pro text příběhu (JTextArea), pole pro název místa (JLabel), samotný font pro celou hru atd.

Takový postup je v tomto případě pro všechno krom JButtonů v podstatě zbytečný, jelikož se všechny prvky použijí jen jednou, ale

```
28 //template for buttons
29 public JButton choiceButton(String text1) {
30     JButton button1 = new JButton(text1);
31     button1.setBackground(Color.black);
32     button1.setForeground(Color.white);
33     button1.setFont(normal);
34     return button1;
35 }
36
37 public JPanel mainPanel() {
38     JPanel panel = new JPanel();
39     panel.setBounds(45, 50, 700, 400);
40     panel.setBackground(Color.black);
41     return panel;
42 }
43
44 public JTextArea story(String st) {
45     JTextArea s = new JTextArea(st);
46     s.setBounds(45, 50, 700, 400);
47     s.setBackground(Color.black);
48     s.setForeground(Color.white);
49     s.setFont(normal);
50     s.setLineWrap(true); //wraps text if too long
51     s.setEditable(false);
52     s.setWrapStyleWord(true); //splits text by words not letters
53     return s;
54 }
```

Obrázek 4: Předlohy, druhá verze

pokud by bylo například potřeba vytvořit několik polí pro místo najednou, jako je to u JButtonů, výrazně by to usnadnilo práci. Tady to pouze dělá program o něco přehlednějším. Následné se z předlohových metod vytvoří jednotlivé prvky, které se dosadí na plochu v herním okně. Do těch je pak napsaný text příběhu, nebo hráčových rozhodnutí.

JPanel je v programu celkem třikrát; *locationP* (předloha pro *locationPanel*) a *mainPanel* (předloha pro *main*), slouží v podstatě jako úložné prostory pro *JLabel* *locationN* (předloha pro *locationName*), a *JTextArea* *story* (předloha pro *storyText*), zatímco *choiceP* (předloha pro *choicePanel*) je úložištěm pro *JButton* *choiceButton* (vyskytuje se celkem třikrát jako *choice1*, *choice2* a *choice3*).

JTextArea a **JLabel** dělají v podstatě to samé; zobrazují text, který je do nich vložen. Rozdíl je v tom, že *JTextArea* jde i po spuštění programu editovat, pokud to, jako v mém programu, není vypnuto příkazem *JTextAreaName.setEditable(false)*; *JLabel* pak dokáže zobrazit jen tolik textu, který se vejde do jeho daného pole, a zbytek pak nahrazuje třemi tečkami (...), zatímco *JTextArea* zobrazuje i text přesahující svoje pole. Není pak od věci přidat příkazy *JTextAreaName.setLineWrap(true)*; což zalomí text na další řádku, pokud přesáhne pole, a *JTextAreaName.setWrapStyleWord(true)*; což zajistí, že je text rozdělen podle slov a ne písmen.

Celý příběh je pak složen z řetězce **public void voidName()**, které jsou pospojovány metodou *switch*. Každá metoda přepíše text příběhu a na buttonech a uloží nový text do *Stringů*, které pak odkazují na danou podmínku v metodě *switch*.

JButton je ovládacím prvkem hry. Na panelu jsou celkem tři a na každém je uložena jedna z možností, kterou si hráč může vybrat. Po té, co hráč danou možnost vybere, práce přejde do metody *actionPerformed* ve třídě *choiceHandler*. V té je

```
389 //class for button choices
390 public class ChoiceHandler implements ActionListener {
391
392     @Override
393     public void actionPerformed(ActionEvent event) {
394         String choice = event.getActionCommand(); //recognises chosen button
395
396         switch (position) {
397             case "void" -> {
398                 switch (choice) {
399                     case "c1" -> {
400                         townOutside();
401                     }
402                     case "c2" -> {
403                         death();
404                     }
405                     case "c3" -> {
406                         break;
407                     }
408                 }
409             }
410             case "townOutside" -> {
411                 switch (choice) {
412                     case "c1" -> {
413                         townTavern();
414                     }
415                 }
416             }
417         }
418     }
419 }
```

Obrázek 5: Metoda *switch*

uložený dlouhý řetězec metody **switch**, ve které je podle **Stringu position**, který se v každé scéně mění, vybrána správná podmínka, která pak podle hráčova výběru přesměruje hru do další scény.

To může svádět k tomu nastavovat text příběhu a rozhodovacích tlačítek přímo v metodě switch, místo vytváření úplně nové metody v hlavní třídě, v takovém případě je ale potřeba změnit i **String position**, protože jinak se sice změní text příběhu a tlačítek, ale program si myslí, že se nachází pořád ve stejné metodě a bude tedy hráče podle jeho dalšího rozhodnutí přesunovat do metody/scény, která je svázaná s předchozím rozhodnutím, což může hráče buď přesunout do úplně jiné části příběhu, nebo ho chytit v nekonečné smyčce.

V programu se také vyskytuje **integer letterOpened**, který je v jedné ze scén změněn na hodnotu 1 a díky tomu se hráči v pozdější scéně otevře nová možnost a s tím nový konec příběhu.

Ke konci příběhu se pak vyskytuje další **String endingName**, do kterého se ukládá v posledních scénách a má pouze tu funkci, že se v konečné zprávě objeví jako název konce, ke kterému se hráč dopravoval.

Závěr

Program nakonec splňuje zadání, ale kvůli omezenému času a přebytku další práce jsem musela hodně věcí vyjmout a příběh výrazně zkrátit. Původně jsem plánovala zakombinovat například náhodné střety s monstry, další menší scény a životy, ale z toho bohužel kvůli nedostatku času sešlo. I tak jsem ale dokázal udělat celkem dobrý příběh a s finálním programem jsem vcelku spokojená.

Zdroje

Vzorová hra [online]. Dostupné z: <https://www.ryisnow.online/2021/04/text-adventure-game-code-for-beginner.html>

Wrapping text in buttons [online]. Dostupné z: <https://stackoverflow.com/questions/5766175/word-wrap-in-jbuttons>

Handling plazer choices [online]. Dostupné z: <https://www.youtube.com/watch?v=LnDQTT3u0MQ>

Seznam obrázků

Obrázek 1: První verze hry

Obrázek 2: Druhá verze hry

Obrázek 3: Příkazy první verze

Obrázek 4: Předlohy, druhá verze

Obrázek 5: Metoda switch