Dokumentace k zápočtovému programu z programování II.

Autor: Tadeáš Tomiška

                Cílem mého zápočtového programu bylo vytvořit ve Windows Form hru v jazyce c# založenou na hře [Bad Ice Cream](https://www.1001hry.cz/zrucnost/bad-ice-cream). Hra používá soubor plan.txt pro načtení herní mapy a soubor ikonky.png pro vykreslení prvků hry. Úkolem hráče je v každém levelu sesbírat všechno ovoce (banány). Pokud uspěje, zpřístupní se mu další úroveň. Je-li hráč během snahy sesbírat banány chycen příšerou, prohrál. Hráč může ke své obraně využívat schopnosti vytvářet ledové kostky (dále **ice**). Hra má dohromady 16 levelů.

Stisknutím tlačítka spustit se zahájí hra. Program program.cs spustí form.cs. Logika hry je uložena v souboru hra.cs (sdílení namespacu). Samotná hra se ovládá pomocí tlačítek (**button**). Hráč ovládá hrdinu pomocí šipek (pohyb) a mezerníku (**ice**). Samotné načítání levelů řídí objekt mapa, která postupně hýbe pohyblivými prvky (hrdina - při stisku tlačítka, příšery) a vykresluje objekty. Počet odemčených levelů si hra eviduje v souboru počet\_odemcenych\_levelu.txt. Některá tlačítka mají na pozadí obrázky = zbývající soubory png. Herní melodie je v souboru song.wav.

Logika hry (Hra.cs):

Objekty:

Hrdina, Prisera – odvozeny od třídy PohyblivyPrvek – metoda UdelejKrok

Mapa

* Řídí všechny levely
* Metody:
  + NactiMapu – načte mapu levelu ze souboru plan.txt (řádek, na kterém začít číst se předává parametrem offset, který má každé tlačítko uložené v Button.TabIndex)
  + NactiIkonky – načte ikonky ze souboru ikonky.png
  + VykresliSe – vykresluje obsah mapy levelu v závislosti na pozici hrdiny
  + PriseryTahnete, PohniHrdinou:
    - Metody volané při hraní levelu
    - Objekty zavolají metodu Udelej\_krok
    - Metoda Udelej\_krok pohne objektem a zavolá příslušné metody mapy
  + Presun\_hrdinu, Presun\_priseru – hne objektem na mapě levelu
  + Otoc\_hrdinu, Otoc\_priseru – otočí objekt na mapě levelu
  + JeVolno, JeBanan, JeKostka, JeStenaPriseraneboKostka, JeHrdina, JePrisera:
    - Stav mapy levelu
    - Boolovské funkce
  + Seber\_Banan
    - Přesune hrdinu na pozici, kde byl banán
    - Zmenší proměnnou počet\_bananu o jedna
  + PostavKostku, SmazKostku – meni obsah mapy
  + Zmena\_Stavu\_Kostek:
    - Boolovská metoda
  + Premen\_Kostky:
    - Došlo-li ke změně stavu kostek, budou na mapě obrázky vznikajících či rozpadajících se kostek, které tato metoda překreslí

Form1.cs:

* Start\_button:
  + Přesun z úvodní obrazovky
  + Zobrazení levelů
* hide\_level\_buttons
* odemkni\_levely:
  + zobrazí tlačítka s levely
* timer1\_Tick:
  + stará se o hru při hraní levelu
* mute\_button, unmute\_button - přehrávání hudny
* button – levely 1-16
* continue\_victory – ukáže se při výhře
* game\_over – ukáže se při prohře

ikonky:

„a“ = hrdina jdoucí směrem dolů

„b“ = hrdina jdoucí vlevo

„c“ = hrdina jdoucí vpravo

„d“ = hrdina jdoucí nahoru

„v“ = příšera jdoucí směrem dolů

„<“ = příšera jdoucí vlevo

„>“ = příšera jdoucí vpravo

„^“ = příšera jdoucí nahoru

„i“ = kostka ledu

„X“ = překážka

„B“ = ovoce (banán)

„r“ = rozpadající se kostka ledu

„n“ = nově vznikající kostka ledu

„?“ = banán přikrytý kostkou ledu

Závěr:

Jsem rád, že jsem si mohl vyzkoušet vývoj velmi jednoduché počítačové hry. Když jsem před rokem nastupoval na Matfyz s cílem studovat vývoj počítačových her, vůbec jsem netušil, kolik to dá práce (asi změním zamýšlený obor). Zejména v grafice mám velké nedostatky, proto jsem si vypůjčil ikonky přímo ze hry BadIceCream. To je taky důvod, proč v celé hře existuje pouze jeden typ příšery. Dalším nedostatkem je neúplně plynulý přechod mezi pohyby objektů.

Má hra je založená na hře, kterou jsme programovali na jednom ze cvičení. Přesto mám pocit, že se někde vytratila přehlednost kódu… Abych nebyl ale jenom negativní, mám velkou radost, protože moje první počítačová hra po několikadenním odlaďování funguje! Jakékoliv nedostatky se případně rád pokusím odstranit.

Tadeáš Tomiška