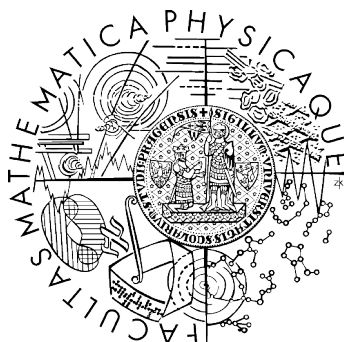


UNIVERZITA KARLOVA V PRAZE
MATEMATICKO-FYZIKÁLNÍ FAKULTA



DOKUMENTACE K PROGRAMU
TETRIS 4DIRECTIONS

ZÁPOČTOVÝ PROGRAM

JAZYK C# A PLATFORMA .NET (NPRG035), ZS 2016/2017
POKROČILÉ PROGRAMOVÁNÍ PRO .NET I (NPRG038), LS 2016/2017

KATEŘINA BŘICHÁČKOVÁ

OBSAH

Úvod.....	4
Uživatelská část.....	5
1 Spuštění hry.....	5
2 Popis hry.....	5
2.1 Skóre.....	6
2.2 Ovládání.....	6
2.3 Uložení hry.....	6
Programátorská část.....	7
1 Celková struktura programu.....	7
2 Logika hry.....	7
2.1 Tetris4DController.....	7
2.2 Tetris4DBox.....	8
2.3 Tetris4DGrid.....	8
2.4 IPiece.....	9
2.5 Ničení zaplněných čtverců.....	9
3 Grafika.....	9
3.1 Windows forms.....	9
3.1.1 MenuForm.....	9
3.1.2 GameForm.....	9
3.1.3 Top10ScoreViewForm.....	10
3.1.4 AskNameForm.....	10
3.1.5 ControlsForm.....	10
3.1.6 TetrisMessageBox.....	10
3.2 Views.....	10

3.3	Messages.....	10
4	Serializace.....	11
5	Možné modifikace programu.....	12

ÚVOD

TETRIS je hra mého dětství a zná ji snad každý (nejenom) v mém věku. Existuje nespočet variací a designů této hry, můžeme si ji zahrát online, na telefonu s Androidem, a dokonce existují už i 3D a 4D varianty této hry.

Má verze, nazvaná Tetris 4Directions, se od klasického Tetrisu liší v tom, že kostičky putují do středu ze všech čtyř směrů, jak už ostatně napovídá sám název.

Tato dokumentace obsahuje uživatelskou část, tedy popis toho, jak s programem pracovat a jak se hra hraje, a programátorskou část, ve které se zaměřuje na hlavní použité třídy a jejich metody, důležité algoritmy, nebo způsob, jak lze hru modifikovat.

Programátorská dokumentace je doplněna o komentáře přímo v kódu, ze kterých je vygenerována přiložená XML dokumentace.

UŽIVATELSKÁ ČÁST

1 SPUŠTĚNÍ HRY

Hra se spouští pomocí souboru Tetris4D.exe. Ve složce s ní jsou všechny soubory potřebné pro správné spuštění programu.

Po spuštění se objeví první okno, ve kterém se nachází menu. Z něj je možné spustit novou hru, pokračovat v rozehrané hře, uložit hru nebo ji ze souboru načíst, nastavit ovládání hry, zobrazit tabulku s nejlepšími hráči nebo hru ukončit.

Po spuštění hry se zobrazí druhé okno, ve kterém probíhá samotná hra. Okno tvoří herní plán a postranní sloupec, ve kterém se zobrazuje další kostička a směr, ze kterého přijde, a také dosažené skóre, skóre nejlepšího hráče a počet zaplněných čtverců.

2 POPIS HRY

Cílem hry je dosáhnout co nejvyššího skóre. Hráč se snaží zůstat ve hře co nejdéle a skládat kostičky, které přicházejí, na hracím plánu.

Na rozdíl od klasického Tetrisu, kde kostičky mizí, když je jimi zaplněna celá řada, zde se hráč snaží poskládat celý čtverec kolem středu. Pokud je čtverec zaplněn, zbytek kostiček se posune směrem ke středu. Pokud se má více kostiček přesunout do stejného místa (tento problém je obvykle s rohy čtverce), posunou se všechny tyto kostičky a jen se „smíchá“ jejich barva.

Padající kostičkou je dovoleno pohybovat pouze v omezeném prostoru – pokud kostička například padá shora, nelze jí pohybovat do pravé a levé části plánu. Může ale projít do dolní části a pokud se hráči nepodaří ji o něco zaseknout, může projít až na druhou stranu plánu a úplně z něj vyjít. To je trestáno záporným skóre.

Hra se postupně zrychluje, takže je stále těžší udržovat plán nezaplňený.

Hra končí, když je plán tak zaplněn, že nově přicházející kostička nemá kam se posouvat, a zůstane „uvězněná“ na startu.

2.1 SKÓRE

Skóre lze získat různými způsoby:

- Umístění kostičky.
- Zaplnění čtverce. Čím více čtverců najednou se hráči podaří zničit, tím větší skóre získává.
- Projití kostičky skrz hrací plán – minusové skóre.

2.2 OVLÁDÁNÍ

Hra se ovládá klávesami a jejich nastavení je možno provést z menu.

Pohyb kostičkou je možný ve třech směrech – směrem, kterým kostička postupuje, a do stran. Kostička se dá otočit klávesou opačnou ke směru pohybu. Pokud tedy kostička půjde shora, jako v klasickém Tetrisu, kostičku lze otočit klávesou Up. Pokud ale kostička půjde zleva, otočení je možné pomocí klávesy Right.

Dále je možné kostičku „shodit“ dolů klávesou Drop.

Hru je kdykoliv možné pozastavit klávesou Pause.

2.3 ULOŽENÍ HRY

Lze mít uloženo libovolné množství her. Rozehranou hru je možno uložit z menu. Hráč vybere umístění, kam požaduje hru uložit, a název souboru. Libovolná uložená hra může být opět načtena, taktéž z menu.

Hra je ukládána s příponou `.Tetris4D`.

Uloženou hru je možno jakkoliv přesouvat či kopírovat, nesmí být ale změněn obsah souboru ani jeho přípona.

Pokud se ukládání nebo načítání hry nepodaří, uživatel je upozorněn chybovou hláškou.

PROGRAMÁTORSKÁ ČÁST

3 CELKOVÁ STRUKTURA PROGRAMU

Program se skládá ze dvou hlavních projektů:

- `TetrisControlProject` – popisuje logiku hry. Hlavním rozhraním je `ITetrisController`, přes který je možno ovládat průběh celé hry. Zvenčí je dále využíváno rozhraní `IPiece` reprezentující Tetris kostičku, a také třídy ze jmenných prostorů `TetrisControlProject.Enums`, `TetrisControlProject.Helper` a `TetrisControlProject.EventArgs`.
- `Tetris4D` – hlavními součástmi jsou `MenuForm` (okno s menu, běží v něm aplikace a poskytuje hlavní ovládání hry) a `GameForm` (herní okno, v němž se vykresluje celá hra). O vykreslování se stará třída `ViewController`, která využívá další views představující různé součásti hry. Jsou zde i další pomocná okna a třídy.

4 LOGIKA HRY

4.1 TETRIS4DCONTROLLER

Třída propojující rozhraní `IScoreProvider` (oceňování různých úkonů příslušným skóre), `IPieceGenerator` (generování dalších kostiček a směrů), `ITetrisBox` (manipulování s kostičkami, které jsou vygenerovány a přidány do něj) a `ILevelProvider` (nastavování levelů s odlišným nastavením).

Když se zavolá metoda `StartGame`, controller nastaví první level, vygeneruje kostičku a přidá ji do boxu. Box s ní dělá, co potřebuje, a když je kostička umístěna, vyvolá se `PiecePlaced` event, kterou controller odchyť. Na základě předaných argumentů zjistí, jestli nedošlo ke konci hry, získá od `IScoreProvideru` příslušné bodové ohodnocení, zkontroluje, jestli se nemá spustit další level, přidá do boxu další kostičku a vygeneruje novou. Takto se pokračuje až do konce hry.

Controller vyvolává události, které říkají, jaký view je potřeba změnit.

Controller dále zpracovává signály od `GameForm` – pokud uživatel stiskne klávesu ovládání, controller zpracuje její význam. Pokud se má kostička pohnout, controller tento signál přijme od `GameForm` a předá `TetrixBoxu`.

4.2 TETRIS4DBOX

Manipuluje s kostičkami, které dostane, v mřížce (třída `ITetrisGrid`), pomocí metod `Move`, `MoveInDirection`, a `DropPiece`.

Uchovává si informaci o aktuální kostičce, její pozici v mřížce a směru. Aktuální kostička se změní metodou `AddNewPiece`.

Umí vrátit mřížku, která se má vykreslit.

4.3 TETRIS4DGRID

Představuje mřížku, ve které se pohybují kostičky.

Pamatuje si pole barev, které představují buď pozadí (tj. volné místo hracího plánu, kde není umístěna žádná kostička), okraje (místo, které by se nemělo vykreslit a jsou tam umístěny nové kostičky), „hrany“ (místa mimo hrací plán), nebo samotné kostičky.

Pokud je kostička přidána do mřížky, nechová se dále jako kostička, ale pouze se zkopírují její barvy na příslušné umístění v mřížce a jsou dále brány jako jednotlivé barevné čtverečky. Kostička je takto natvrdo přidána metodou `AddPieceToGrid` do mřížky až poté, co skončil její pohyb a byla umístěna.

Mřížka umí nacházet a ničit zaplněné čtverce, posouvat zbytkem kostiček, rozhodnout počáteční pozici pro kostičku, rozhodnout, zda kostička prošla skrz plán nebo je zaseknutá na začátku (to znamená konec hry), zjistit, zda lze kostičkou pohnout v daném směru a pokud ano, udělat to, apod.

4.4 IPIECE

Rozhraní, které reprezentuje kostičku Tetrisu. Hlavní vlastností je **Shape**, což je pole barev, které určuje tvar dané kostičky. Protože kostičky obvykle nejsou obdélníkové, může být v poli i nul – znamená to, že tvar zde nepokračuje.

4.5 NIČENÍ ZAPLNĚNÝCH ČTVERCŮ

Čtverec pro jednoduchost představuje třída **TetrisSquare**, kde je čtverec reprezentován levým horním rohem a délkou stran.

Nejprve se najdou čtverce, které jsou zaplněny pouze barvami kostiček. Potom se tyto čtverce vymažou, aby byly prázdné. Nyní se projdou všechny čtverce v hracím plánu od nejmenšího po největší. Pokud je čtverec prázdný, najde se první větší, který prázdný není, a tyto čtverce se vymění. Prázdný tedy bude ten větší čtverec a menší se zaplní barvami. Takto se postupuje, dokud všechny prázdné čtverce nejsou „navrchu“. Prázdné čtverce tedy postupně „probublají“ ven.

Jelikož tímto způsobem můžou vzniknout nové plné čtverce, algoritmus se opakuje, dokud jsou nějaké plné čtverce nacházeny.

5 GRAFIKA

5.1 WINDOWS FORMS

Celá grafika je vytvořena ve Windows Forms. Použitá okna jsou následující:

5.1.1 MENUFORM

Okno s menu. Samo si vytvoří bitmapu, která se vykreslí na pozadí.

5.1.2 GAMEFORM

Herní okno předá své komponenty `ViewControlleru`, který je mění podle událostí, které se dějí. Pokud se změní velikost okna, předá se tato informace `ViewControlleru`.

5.1.3 TOP10SCOREVIEWFORM

Otevírá se z menu. Vykresluje 10 nejlepších skóre získaných ze seznamu, který je mu předáván v konstruktoru.

5.1.4 ASKNAMEFORM

Je využíváno, když se ukládá nové skóre. Uživatel je vyzván, aby zadal své jméno, případně může ukládání skóre zrušit.

5.1.5 CONTROLSFORM

Slouží k zobrazení kláves ovládání a k jeho změně.

5.1.6 TETRISMESSAGEBOX

Je obdobou klasického `MessageBoxu`. Slouží ke zobrazování hlášek, varování a chyb uživateli.

5.2 VIEWS

Při předání potřebných argumentů vrátí bitmapu zobrazující aktuální stav. `GridView` zobrazuje stav hrací mřížky s pohybující se kostičkou. `NextPieceView` ukazuje následující kostičku a směr, ze kterého přijde. `ScoreView` zobrazuje aktuální skóre, nejlepší skóre a počet zaplněných čtverců.

Nad nimi stojí třída `ViewController`, který získané bitmapy zobrazuje do komponent `GameFormu`.

5.3 MESSAGES

Třída `Messages` uchovává na jednom místě všechny texty a zprávy zobrazované uživateli. U oken je metoda `initMessages`, která nastaví komponentám příslušné texty. Díky tomuto bz bylo snadné změnit jazyk celé aplikace.

6 SERIALIZACE

Pro veškerou serializaci i deserializaci je používán `MyFormatter`.

Po každé změně ovládání se serializuje `IKeysSettings` do souboru `keysSettings.Tetris4D`. Stejně tak při každém uložení nového skóre se `IScoreStorage` serializuje do souboru `score.Tetris4D`.

Při spuštění aplikace se hledají tyto dva soubory a pokud nejsou nalezeny nebo se nenačtou správně, použije se defaultní nastavení. Při serializaci se staré soubory smažou.

Uložení rozehrané hry je možné kdykoliv v jejím průběhu. Jedna hra může být uložena i ve více stavech a uživatel může mít uložené libovolné množství her. Sám uživatel si zvolí umístění, kam chce hru uložit. Musí si ji pojmenovat. Soubor je uložen s příponou `.Tetris4D`.

6.1

7 MOŽNÉ MODIFIKACE PROGRAMU

V průběhu psaní programu mě napadalo spoustu možností, jak by šlo hru vylepšit, a někdy v budoucnu bych se k ní určitě ráda vrátila a dotáhla jí do konce.

Díky tomu, jak je hra navržena, není velký problém oddělit logiku hry od grafiky. To by mohlo pomoci, pokud bych se rozhodla vytvořit hezčí grafiku například v Unity nebo naprogramovat hru pro Android.

Dal by se například vytvořit editor kostiček, kde by si hráč vymodeloval svoje kostičky. Poté by stačilo implementovat rozhraní `IPieceGenerator` a předat mu tyto kostičky.

Lehce se dá měnit velikost plánu – stačí vytvořit `Tetris4DGrid` příslušných rozměrů.

Jednoduché by bylo i vyrobení nových levelů a bodování skóre.

Různými implementacemi třídy `ITetrisGrid` by bylo možné vytvořit hrací plány různých tvarů. Hra by byla jistě převoditelná na klasický Tetris.

Vzhled kostiček by se změnil změnou metody `TetrisSquareImage.GetTetrisSquareImage`.

Zajímavá by byla možnost současné hry dvou hráčů.