

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ  
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

# TVORBA POČÍTAČOVÝCH HER S VYUŽITÍM TECHNOLOGIE XNA

COMPUTER GAMES DESIGN USING XNA TECHNOLOGY

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**VOJTĚCH ROJÍČEK**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**ING. JAN ROUPEC, PH.D.**

BRNO 2013

## **ZADÁNÍ ZÁVĚREČNÉ PRÁCE**

(na místo tohoto listu vložte originál a nebo kopii zadání Vaší práce)



## **ABSTRAKT**

Cílem této bakalářské práce je seznámit se s historií a teorií obklopující herní průmysl a následně vypracovat ukázkové herní aplikace pro platformu počítačů s využitím XNA. Tyto ukázky budou posléze zdarma ke stažení na webových stránkách Ústavu automatizace a informatiky.

## **ABSTRACT**

The aim of this bachelor thesis is to inform about history and theory of game industry and afterwards develop sample games for computer platform using XNA technology. It will be possible to download these games for free from the webpage of Institute of Automation and Computer Science.

## **KLÍČOVÁ SLOVA**

XNA, .NET, hra

## **KEYWORDS**

XNA, .NET, game



## **PROHLÁŠENÍ O ORIGINALITĚ**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval sám, za použití uvedených zdrojů.

Rojíček Vojtěch, Brno 2013

## **BIBLIOGRAFICKÁ CITACE**

ROJÍČEK, V. Tvorba počítačových her s využitím technologie XNA. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2013. 37 s. Vedoucí bakalářské práce Ing. Jan Roupec, Ph.D.



**Obsah:**

	<b>Zadání závěrečné práce.....</b>	<b>3</b>
	<b>Abstrakt.....</b>	<b>5</b>
	<b>Prohlášení o originalitě.....</b>	<b>7</b>
<b>1</b>	<b>Úvod.....</b>	<b>11</b>
<b>2</b>	<b>Historie videoher.....</b>	<b>13</b>
2.1	Počátky herního fenoménu.....	13
2.2	Zábava se mění v bussines.....	13
2.3	Válka mezi konzolemi.....	14
2.4	Vzestup počítačů.....	15
2.5	Nové tisíciletí.....	15
<b>3</b>	<b>Rozdělení her.....</b>	<b>17</b>
3.1	Hry podle žánrů .....	17
3.1.1	Dobrodružné hry (Adventure games).....	17
3.1.2	Deskové a karetní hry (Board and Card Games) .....	17
3.1.3	Bojové hry (Fighting games).....	17
3.1.4	Závodní hry (Racing games).....	17
3.1.5	Hry na hrdiny (Role-Playing Games – RPG).....	17
3.1.6	Akční – „střílečka“ (Shooter).....	18
3.1.7	Simulátory (Simulations).....	18
3.1.8	Strategie (Strategy).....	18
3.1.9	Textové hry (Text Adventure).....	18
3.2	Hry podle počtu hráčů.....	18
3.2.1	Hra pro jednoho hráče (Singleplayer).....	18
3.2.2	Hra více hráčů (Multiplayer).....	18
3.3	Hry z grafického pohledu.....	19
3.3.1	2D hry.....	19
3.3.2	3D hry.....	19
<b>4</b>	<b>.NET framework, C#, XNA.....</b>	<b>21</b>
4.1	C#.....	21
4.2	.NET framework.....	21
4.2.1	Základní charakteristika.....	21
4.2.2	.NET vrstvy.....	22
4.3	XNA Framework.....	23
4.3.1	Úvod do XNA.....	23
4.3.2	Programování v XNA.....	23
4.3.3	Grafická část XNA.....	24
<b>5</b>	<b>Řešení .....</b>	<b>27</b>
5.1	Had.....	27
5.1.1	Pravidla hry.....	27
5.1.2	Architektura hry.....	27
5.1.3	Rozbor algoritmu.....	28
5.2	Pexeso.....	30
5.2.1	Pravidla hry.....	30
5.2.2	Architektura hry.....	30
5.3	Skákací kulička.....	32
5.3.1	Pravidla hry.....	32
5.3.2	Architektura hry.....	32
<b>6</b>	<b>Závěr.....</b>	<b>35</b>
	<b>Seznam použité literatury.....</b>	<b>37</b>



# 1 ÚVOD

Videohry jsou dnes neodmyslitelnou součástí našeho každodenního života. Setkáváme se s nimi denně na každém kroku buďto jako aktivní uživatelé nebo minimálně jako pozorovatelé či posluchači. Hráčů je ve většině států víc, než věřících a určitě už nějakou tu videohru vyzkoušeli i ti, co se za hráče nepovažují, neboť nehrají pravidelně.

Tato bakalářská práce v úvodních kapitolách seznamuje s historií herního průmyslu a dělením her do skupin. Následuje analýza prostředků použitých k vytvoření ukázkových her a detailnější rozbor jejich vnitřní architektury a funkčnosti.



## 2 HISTORIE VIDEOHER

Herní svět během své, ne tak dlouhé, existence prošel neuvěřitelnou proměnou. Z počátečních pokusů, kdy šlo hlavně o zábavu mezi hrstkou lidí, se za pár let stal obrovský business s miliardovou základnou odběratelů, který bezesporu patří k největším fenoménům moderní doby.

Informace v následující kapitole jsou, pokud není uvedeno jinak, čerpány z následujících zdrojů.[1][2][3][4]

### 2.1 Počátky herního fenoménu

Dříve než můžeme říct, kdy byla první videohra vytvořena, je nutné stanovit, co se za videohru dá považovat a co už nikoliv. Pokud vyjdeme z definice: „Videohra je elektronická hra, hraná manipulováním obrazu, vytvořeného programem, na televizním nebo jiném typu obrazovky.“[5], vznikají první videohry po druhé světové válce.

Konkrétně roku 1947, vytvořili fyzici Thomas Goldsmith a Estle Ray Manna první stroj s videohrou, který nazvali: *Cathode-Ray Tube Amusement*. Display vzhledem připomínal vojenský radar. Uživatel mohl pomocí knoflíku ovládat směr, kterým se po něm pohyboval světelný bod, jenž měl simulovat letící střelu. O rok později získali jeho autoři na tento stroj patent.[6] Toto byl ovšem jeho jediný exemplář. Nicméně se tímto vynálezem podařilo dokázat, že stroje nemusí jen ze zadání vstupních podmínek spočítat výsledek, ale dokážou interaktivně reagovat na příkazy uživatele a tím ho dokonce i pobavit.

Další pokusy s vytvořením zábavných her na sebe nenechaly dlouho čekat. V roce 1951 vytvořil autor Ralph Baer hru *NIM* na zařízení NIMROD. Jedná se o adaptaci známé hry na odebrání sirek. NIMROD bylo prvním zařízením zkonstruovaným speciálně pro hru. Výstupem však pouze žárovky.

Následující rok 1952 vznikla první skutečná počítačová hra. Jednalo se o hru *OXO* česky známou jako piškvorky. Jejím tvůrcem je Alexander Douglas a běžela na počítači Edsac. Použitý algoritmus je první herní umělou inteligencí. Nejpopulárnější videohra tehdejší doby, určená pro širší veřejnost, ovšem vznikla až v roce 1958 a byla jí hra *Tennis for Two*, kterou vyvinul nukleární fyzik William Higginbotham pro návštěvníky Brookhavenské národní laboratoře. Hra v sobě obsahovala dokonce i prvek gravitace.

Nicméně k dnešní podobě her měly tyto první pokusy ještě velmi daleko, hlavně z důvodu velkého množství hardwarových komponent, které byly k jejich provozu zapotřebí, nemohly zasáhnout tak velkou masu uživatelů jako v současnosti. Tento handicap se pokusili vizionáři změnit v následujícím desetiletí.

### 2.2 Zábava se mění v bussines

V roce 1961 naprogramoval na počítači PDP-1 student Massachusettského technického institutu Steve Russel hru *Spacewar!*. Základem hry jsou dvě rakety ovládané hráči. Rakety létají v gravitačním poli Slunce s omezeným množstvím paliva a střel. Úkolem je sestřelit protivníka. Právě tyto základní myšlenky hry, soupeření více hráčů, bodovací systém hry a omezené zdroje, inspirovaly nejen spoustu modernějších her, ale také vedly ke vzniku prvních komerčních zařízení.

S příchodem integrovaných obvodů se tento cíl stal i skutečností. Začaly se vyvíjet herní arkádové automaty sloužící k hraní her za peníze, a tak v roce 1971 spatřila světlo světa první komerční hra *Computer Space* jejímiž tvůrci jsou Nolan Bushnell a Ted Badney, kteří založili firmu Atari. *Computer Space* se inspirovala právě starší *Spacewar!*, která nebyla nikdy patentována.

Roku 1972 vydala firma Atari komerčně veleúspěšnou hru *Pong*. První přístroj umístěný v jednom z barů přestal po čtrnácti dnech fungovat z důvodu velkého množství čtvrtáků uvnitř.[7]

Ještě téhož roku byla veřejnosti představena i první domácí herní konzole, která běžela i na obyčejných televizorech. Brown box, uvedený na trh pod obchodním názvem Magnavox Odyssey, je výtvoem Alpha Bauera. S vývojem začal v roce 1966 a první prototyp dokončil o dva roky později. Úspěch těchto zařízení byl tak obrovský, že ho brzy následovalo mnoho dalších pokusů.

V roce 1974 se objevila hra *Gran Trak 10*, první hra využívající volant a řadící páku.

Roku 1976 byla ve hře *Space Wars* poprvé použita vektorová grafika a o další dva roky později představila firma Midway první hru v barvách *Space Invaders*. Dnes velmi dobře známá firma Nintendo, uvolnila v tomto roce svoji první arkádovou hru *Othello*.

Hra *Ozma Wars*, představená v roce 1979, přinesla poprvé koncept rozdílných úrovní a závěrečného bossa. Ve hře *Tailgunner* oproti tomu mohli hráči poprvé spatřit herní svět z pohledu první osoby tzv. first-person camera. Největší úspěch tohoto roku zaznamenala ovšem novinka od Atari – *Asteroids*. Arkádových automatů s touto hrou se prodalo celosvětově přes padesát tisíc kusů. Roku 1980 ovšem vydala japonská firma Namco nejúspěšnější arkádu všech dob. Hra *Pac-Man* jejíž původní název zněl *Puck-Man*, z japonské fráze „paku-paku“ používané pro zvuk vznikající při jídle, ale vydavatel se bál zaměňování úvodního písmene v názvu hry, a proto ji přejmenoval, byla obrovským hitem. Prodalo se jí přes tři sta tisíc kusů a později se dočkala mnoha nových adaptací.

V oblasti domácích herních konzolí také probíhal bouřlivý vývoj, protože rozhodně nechtěly zůstat pozadu.

Americká společnost Fairchild Camera & Instrument předvedla světu v roce 1976 první herní konzoli, která používala kazety, tzv. cartridge. Tento průlomový výtvar, postavený na prodávání jednotlivých her na kazetách místo dodání celého hardwaru, využila následně Atari pro svůj Atari 2600 nebo také Atari VCS tzn. video computer system. Z počátku nemělo toto zařízení tak velký úspěch, neboť hardware byl příliš drahý a hry nebyly moc kvalitní a zábavné. Zlom nastal v polovině roku 1980, kdy společnost Atari předělala populární hru *Space Invaders* a následujícího roku přidala ještě *Pac-Mana*. Firma měla v této době podíl na herním trhu přes 80%.

Jelikož kazety nebyly kódovány, mohl hry vyvíjet kdokoliv. Do roku 1982 bylo na světě přes stovku her, které si od té doby mohli vlastníci konzolí zahrát v teple domova. A trh se začal přeplňovat.

Bohužel pro americký videoherní průmysl, byly tyto hry mnohdy nekvalitní a drahé, a tak nastal velký úpadek. Společnosti měly velké finanční ztráty a mnohdy i zkrachovaly. Těžiště vývoje se přesunulo z USA do Japonska a zlatý věk konzolí v této oblasti na čas skončil.

Velkou měrou k tomu přispěl i vzestup nových levných osobních počítačů. Ty převzaly štafetu hlavní herní platformy. Hovořil pro ně lepší hardware, hry se daly jednodušeji kopírovat přes diskety, měly lepší grafiku, zvuk a větší paměť. Postup ve hře se dal dokonce uložit, takže uživatel mohl pokračovat tam, kde naposledy přestal. V roce 1981 se na počítačích objevil *Castle Wolfenstein* fungující v roce 1984 i na PC sestavách se systémem MS-DOS.

### 2.3 Válka mezi konzolemi

Ztroskotání na trhu s konzolemi využily nové společnosti, které se chtěly prosadit. V roce 1984 uvedla na japonský trh firma Nintendo konzoli Famicom. O rok později ji představila i na americkém kontinentu pod názvem Nintendo Entertainment System neboli NES s jednou z nejpopulárnějších her, která je k vidění dodnes, *Super Mario Bros*. Následujícího roku ji následovaly také Sega se svou Sega Master System a Atari s Atari 7200. I když měla Sega vyspělejší zařízení, prodávanější bylo Nintendo. Opět se tedy prokázalo, že to, co se prodává, jsou hry, nikoliv stroje.

Roku 1987 vydalo Nintendo další populární ságu *The Legend of Zelda*.

Konec osmdesátých let patřil opět Nintendo, jenž uvedlo na trh Game Boy. Vůbec první herní přístroj do kapsy. Spolu s Game Boyem se uvedla také velice návyková hra *Tetris* od ruského autora Alexeje Pajitnova. Ale ani Sega se svou novinkou Sega Game Gear a Atari s Atari Lynx nechtějí být pozadu a přinášejí do těchto zařízení barevné displeje. Nevýhodou byla ovšem výdrž baterie.

V devadesátých letech dvacátého století začal další velký herní boom. Neplatilo to sice pro starší a velké arkádové automaty, jejichž popularita velmi klesla a to převážně z důvodu kvality. Vývojáři se prostě vydali jinou cestou, z toho vyplynul obrovský nárůst výkonu v konzolích.

Nintendo vytvořilo novější verzi NESu: Super NES spolu se hrou *Super Mario Bros 3*, Sega na oplátku předvedla Mega Drive také včetně hry hrané dodnes: *Sonic the Hedgehog*.

Na scéně byly i další firmy, ale těmito dvěma velikánům patřila naprostá většina trhu.

## 2.4 Vzestup počítačů

Ale jak se říká: „Když se dva perou, třetí se směje.“ Mezi tím, co se vedla válka na trhu s konzolami, probíhala další velká bitva na poli počítačů. Ty měly vysoké ambice i ve světě videoher a vzhledem k jejich přednostem se začaly rychle prosazovat. Firmy jako Hewlett-Packard, IBM, Apple, Intel a Microsoft pracovaly na co nejlepším zdokonalení PC sestav jak po hardwarové, tak po softwarové stránce. Oproti konzolám tak měly několik podstatných výhod. Vyznačovaly se vyšším rozlišením, větší operační pamětí, mohly pohodlně uložit data ze hry na harddisk, poskytovaly o hodně kvalitnější zvukové efekty, bývaly vybaveny výkonnějšími procesory a hraní pomocí klávesnice a myši se ukázalo velice pohodlné. I běžný domácí počítač disponoval větší výkonností než konzole a byl tedy schopen bez problémů utáhnout i novější hry. Nevýhodou bylo, že většina lidí používala operační systém DOS, který nebyl úplně uživatelsky přívětivý, jelikož bylo potřeba instalovat různé ovladače na zvuk a video. Instalace her tedy nebyla pro úplné laiky. Pozitiva ovšem výrazně převážila a i tento nedostatek se potlačil s příchodem Windows 95 (1995).

Jejich vzestupu také neodmyslitelně pomohla řada skvělých her, jejichž popularita v novějším podání, často přetrvává dodnes.

V devadesátých letech vznikly jedny z neznámějších her. Patří mezi ně například *Prince of Persia* (1989) od tvůrce Jordana Mechnera. Velmi známá 2D plošinovka, kde má hráč-prince, přesně šedesát minut na záchranu své milované. *Road Rash* (1991), populární simulátor pouličních závodů na motorce, kde není nouze o nečisté praktiky jako udeření protihráče řetězem nebo baseballovou pálkou, od Electronic Arts. *Mortal Kombat* (1992) z dílny Midway, známá hra kde se v aréně bojuje na život a na smrt. *Wolfenstein 3D* (1992), vyvinutý firmou Apogee, jedná se o pokračování předchozí klasiky *Castle of Wolfenstein*. O rok později si mohli lidé zastřílet z brokovnice a dalších zbraní v kultovní hře *Doom* od vydavatele Id Software. Roku 1995 přichází velice povedená strategie *Command & Conquer*, kterou uvedlo na trh Westwood Studios a později přešla pod křídla EA. Téhož roku představil New World Computing, později odkoupený 3DO a následně Ubisoftem, tahovou strategii *Heroes of Might and Magic*.

Následující rok vypustil Id Software další veleúspěšnou střílečku *Quake*, kterou si v online verzi každý s chutí zahraje i v dnešních dnech. V říjnu 1996 se také na obrazovkách ukázala, pod záštitou společnosti Core Design, nejslavnější archeoložka Lara Croft v prvním dílu série *Tomb Rider*.

Koncem tohoto desetiletí se ještě odhalilo známé *Grand Theft Auto* od Rockstar Games a *Half-Life* od Valve Corporation.

Úspěch těchto her také velice zvýšil zájem o vývoj grafických karet podporujících 3D grafiku. Stále víc her se vyvíjelo v 3D pohledu a tak se tyto karty rychle staly standardním vybavením každého počítače.

Výrobci konzolí si uvědomili, že už nesoupeří jen mezi sebou, ale mají za nepřítel také domácí počítače. V roce 1995 vyvinula firma Sony další herní konzoli Playstation.

Nintendo se na oplátku vytasilo v roce 1998 s Game Boy Color, který svého předchůdce předčil ve všech parametrech a chlubil se rozlišením obrazovky 160x144 pixelů. Hlavní výhodou také byla schopnost komunikovat s dalšími zařízeními a umožnit tak hru více hráčů. Jednou z nejhranějších her, nad kterou hráči strávili pořádný kus času, bylo RPG *Pokémon*. Tato hra využila mánie kolem virtuálního světa Pokémonů, který vynikl v Japonsku, a přivedla ho z televizních pláten a sběratelských hracích karet i do světa videoher. Během deseti let se jí prodalo přes sto čtyřicet miliónů kopií.

## 2.5 Nové tisíciletí

S příchodem nového tisíciletí se toho ve světě her opět spoustu událo. Přichází nová generace konzolí, která přináší zlepšení, jak po výpočetní, tak po grafické stránce. Jako první představuje Sega zařízení Dreamcast s bojovou hrou *Virtua Fighter 3*. Nicméně roku 2001 zanechává Sega vývoje vlastních zařízení a pokračuje ve vývoji her pro ostatní výrobce.

Naproti tomu Sony přichází se svým PlayStationem 2, Nintendo s GameCube a Game Boy Advance. Microsoft se vytasil se svým Xboxem také v roce 2001. Všechna tato zařízení potkala výrazné zkvalitnění oproti předchůdcům, nicméně vývoj her pro konzole se stal mnohem náročnějším,

neboť komunita požadovala lepší design, kvalitnější zvuk a delší herní čas.

Hraní na PC se mezitím stále rozvíjelo. Hráči preferují počítače, protože mají stále kvalitnější výkonostní parametry a klávesnici s myší se prostě v této oblasti nic nevyrovná.

Pro tvůrce her se zde začaly ovšem vyskytovat značné problémy. Ve světě bylo čím dál větší množství procesorů, grafických karet, pamětí a dalších periférií, a tak bylo stále obtížnější vyvíjet hry, které by spolehlivě fungovaly na všech. Dalším kamenem úrazu se stalo „crackování“. Nehledě na to, co firmy podnikly pro ochranu, byly hry nabourávány a šířeny přes internet zdarma. Společnostem takto utíkaly obrovské zisky. Mnoho vývojářů proto začalo vydávat hry na konzolích a pro počítače je uvolňují až s odstupem času. I přes tyto obtíže však vznikají zajímavé výtvořky, které stojí za to zkusit. Na některých z nich se dokonce lidé stali závislí a strávili nad nimi nejen mládí.

Mezi takové hry patří například *World of Warcraft*, vydaný Blizzard Entertainment roku 2004. Tento virtuální svět je jedním z největších na světě i v dnešní době, díky více než deseti miliónům platících zákazníků. Dále zde uvedu *Prince of Persia: The Sands of Time* (2003 - Ubisoft), *Half-Life 2* (2004 - Valve), *Battlefield 2* (2005 – Electronic Arts), *Sid Meier's Civilization IV* (2005 – 2K Games), *Football Manager 2007* (2006 - Sega), *Call of duty4: Modern Warfare* (2007 - Infinity Ward, Aspyr, Activision), a mnoho dalších.

Z nejžhavějších novinek jsou to *Assassin's Creed III* (2012 - Ubisoft), *BioShock Infinite* (2013 – Irrational Games, 2K Games), *Tomb Raider* (2013 - Eidos Interactive). V příštím roce se má ukázat například hra *Thief* od Eidos Interactive, která slibuje obrovský herní zážitek.

Konzole ovšem také nezasply. Stále pracují na odstranění hardwarových nedostatků a s každým novým výtvořem zvyšují výkon. Kromě toho ovšem hledají i cesty, jak přinést něco nové a snaží prodat svoje přednosti.

V roce 2005 uvedl Microsoft na trh Xbox 360, který patří mezi nejnovější generaci herních konzolí. Sony roku 2006 poslal do prodeje svůj PlayStation 3. Nintendo se oproti tomu vydala úplně jiným směrem a vyvinula Nintendo Wii (2006). Nintendo Wii nebyl tak výkonný jako jeho konkurenti, ale přinesl úplně nový způsob ovládání. Hráč má ovládač, který snímá jeho pohyby a řídí tak děj ve hře. Mnoho sportovních her tak pobavilo běžné uživatele, neboť jim Wii doneslo sportovní zážitek rovnou do obývacích pokojů.

Úspěch byl obrovský. Nintendo ještě vydalo v roce 2008 Wii Fit, které se ovládalo pomocí balanční podložky, což dovoľovalo hráčům hrát si a zároveň formovat postavu. Reakce konkurence na tento úspěch přišla roku 2010. Sony představilo ovládač ke konzoli PlayStation Move a Microsoft zamířil ještě o kousek dál a vyvinul pro svůj Xbox 360 zařízení Kinect. Tento aparát snímá pohyby lidského těla bez jakýchkoliv jiných pomůcek. Přináší tak pobavení široké veřejnosti a rozhodně nemíří na profesionální hráče.

V těchto letech se objevil ještě jeden nový herní konkurent a to v podobě mobilních telefonů a tabletů. Rok 2007 byl rokem, kdy Apple ukázal první generaci iPhone. Mobilní telefon ovládaný dotykovou obrazovkou si vydobyl velkou popularitu a následovalo jej mnoho dalších konkurentů. Chytré telefony a tablety vybavené gyroskopy a dotykovými displeji se ukázaly jako velmi zajímavé přístroje pro hraní her a trh toho řádně využil. Největší novinkou v této oblasti jsou phablety, které svou velikostí spadají mezi oba kolegy, ale jinak se v podstatě ničím neliší.

Minulost videoher je, i přes jejich krátkou existenci, velice bohatá. Toto je jen stručný přehled herní historie se snahou poukázat na nejdůležitější milníky, kterých bylo během tohoto vývoje opravdu mnoho. Uvedené hry jsou jedny z nejoblíbenějších. Denně jsou vyvíjeny desítky dalších, nových her, jenž se snaží pobavit majitele a udržet je co nejdéle u obrazovek. Není tedy možné zde uvést úplně všechny. Věřím, že i přes tento prudký vývoj je v herním průmyslu velký potenciál a v nejbližší době nás překvapí dalšími úžasnými novinkami.

## 3 ROZDĚLENÍ HER

V posledních padesáti letech bylo vytvořeno tolik videoher, že se stalo nezbytným jejich členění do skupin podle stejných nebo podobných herních prvků. Existuje mnoho možností jak je dělit. Nejčastěji se využívá rozdělení podle žánrů. V této oblasti se také velice často používají originální anglické názvy, zkratky nebo slangové výrazy, které nezní tak často jako české ekvivalenty. Hlavním zdrojem informací pro tuto kapitolu je: [8]

### 3.1 Hry podle žánrů

Toto rozdělení her je nejběžnější a nejvíc využívané. Mnoho titulů ovšem nespadá jednoznačně jen do jedné oblasti, ale spojují v sobě prvky různých žánrů. Některé videohry jsou dokonce natolik specifické, že časem došlo k vytvoření samostatné skupiny jen pro ně. Já zde ovšem uvedu jen ty zásadní.

#### 3.1.1 Dobrodružné hry (Adventure games)

„Adventure“ se obvykle skládají z herního světa, který je ohraničen a reaguje jen na přesné pokyny. Je zde spojeno několik různých obrazovek nebo místností, hráč tedy nemá plnou volnost pohybu. Nicméně určitá míra svobody je nezbytná, neboť uživatel musí vyřešit stanovené problémy za pomoci skládání a využívání nalezených předmětů a dosáhnout tak stanoveného cíle hry.

#### 3.1.2 Deskové a karetní hry (Board and Card Games)

Tato oblast nebývá zrovna v hledáčku profesionálních hráčů videoher, ovšem některé deskové a karetní hry se staly tak populárními, že buďto nadšenci nebo společnosti vytvořily jejich adaptace pro svět počítačů. Většinou je u hraní nutné být připojen k internetu, což dovoluje vyzvat na duel soka z kteréhokoliv koutu Země.

Z deskových her uvedu pro příklad *Šachy*, *Dámu*, *Scrabble* a *Monopoly*. Karetní hru *Solitaire* určitě každý zná, ale patří sem i *Bang* nebo pro dnešní dobu velice populární *Poker* (ve variantě Texas hold'em).

#### 3.1.3 Bojové hry (Fighting games)

U bojových her platí jen jedno jednoduché pravidlo. Musíte zneškodnit soupeře dříve než on vás. Konkurenti na virtuálním bojišti jsou v tomto případě srovnatelně silní, i když mohou mít rozdílné speciální vlastnosti a schopnosti. Vítězství tedy závisí převážně na hráčských schopnostech, postřehu a zkušenostech.

Mezi nejslavnější hry této oblasti patří série *Mortal Combat*.

#### 3.1.4 Závodní hry (Racing games)

Hlavním cílem hry je samozřejmě vyhrát závod popřípadě zajet ten nejlepší čas. Módní záležitostí posledních let se stalo nahrazování tohoto typu her simulátory, které se snaží být maximálně realistické. Občas si ale každý rád zajezdí s autem, které ani po nárazu do domu neutrpí žádné škrábnutí a může tedy směle pokračovat dále směrem za cílovou pásku. Fyzika ani reálné zákony tu neplatí. Hlavní je závod samotný, detaily jsou vedlejší.

Na internetu má popularitu například *TrackMania*.

#### 3.1.5 Hry na hrdiny (Role-Playing Games – RPG)

Základním hnacím motorem je rozsáhlý virtuální svět kde hráč, ovládající zpravidla svou jednu postavu, může dělat v podstatě cokoliv. Hra je řízená úkoly, při jejichž plnění se postava za odměnu stává silnější a bojeschopnější. Děj zde bývá velice promyšlený a je na něj kladen velký důraz. Typickým prvkem zde bývá cestování po rozsáhlých oblastech, díky němuž narůstá rapidně herní čas.

Známou RPG ságou je *Diablo*, přičemž *Diablo III* se stalo, podle ankety, druhou nejlepší RPG hrou roku 2012.[9]

### 3.1.6 Akční – „střilečka“ (Shooter)

Jedna postava, stovky nepřátel a spousta arsenálu. Úkol je tedy jasný, všechny je zabít a zničit. Tak bych představil standardní „střilečku“. Na rozdíl od her jeden na jednoho je tady hlavní postava výrazně silnější než protivníci. Ti tento nedostatek ovšem kompenzují počtem a hráč musí mít dobré reflexy a přesnou mušku aby prošel až do konce. Zpravidla zde bývají zastoupeny i vizuální prvky, které nejsou určeny dětem.

Dělíme je na dvě základní skupiny podle pohledu na herní svět. Buďto se jedná o pohled první osoby (first-person shooter, FPS), kam patří hry *Doom* a *Wolfenstein*, nebo pohled třetí osoby, mezi takovéto patří například legendární *Asteroids* (third-person shooter, TPS).

### 3.1.7 Simulátory (Simulations)

Všechny simulace se v podstatě snaží o to stejné, přiblížit hráčům nějakou skutečnou situaci, prostředí nebo realitu. Možností co simulovat je opravdu neskutečné množství. Existují simulátory letecké, sportovní, závodní, atd. U jemnějšího pohlaví se těší velké oblibě také populární simulátor života, *The Sims*. Úkolem v této hře je starat se o růst, zdraví a štěstí vašeho digitálního člena rodiny popřípadě rodiny celé.

### 3.1.8 Strategie (Strategy)

Mezi nejúspěšnější herní žánr patří strategie. Vzhledem k jejich množství se ještě dále rozdělily na strategie tahové (TBS), reálné (RTS) a budovatelské (manažerské).

V tahových může každý hráč během svého tahu dělat vše, co mu pravidla dovolí a posléze předá tah dále. Důležité je zde plánování. Známou hrou, která se stále vyvíjí a dočkala se zatím svého šestého pokračování, je *Heroes of Might and Magic*.

Oproti tomu reálné strategie nejsou nijak rozdělené, kladou tak na hráče velké nároky na postřeh a umění dělat více věcí najednou - stavět, bojovat a těžit zdroje. Zde spadá například *Warcraft III* a *Age of empires*.

Budovatelské strategie oproti tomu se nesnaží zničit protivníka, naopak herní myšlenkou je stavět vlastní prospěšnou civilizaci, nemocnici apod.

### 3.1.9 Textové hry (Text Adventure)

„Textovky“ jsou jen vzpomínkou na dávné časy. Dnes se vývojáři snaží o co nejuvěrnější podání prostředí a každého detailu, což je umožněno výrazným zvýšením grafického i výpočetního výkonu v posledních letech. V minulosti se ovšem hrávalo i hry bez jakýchkoliv obrázků. Celá herní scéna byla jen popsána slovy a uživatel tak mohl nechat obraz vzniknout jen ve své vlastní hlavě. Pomocí textových příkazů pak hledal cestu z bludiště nebo osvobozoval zakletou princeznu.

## 3.2 Hry podle počtu hráčů

Tento způsob dělení je celkem jasný. Hry se podle tohoto kritéria dělí na dvě oblasti. Hry pro jednoho hráče a pro více hráčů.

### 3.2.1 Hra pro jednoho hráče (Singleplayer)

Tento herní mód má téměř každá hra. Člověk zde potřebuje jen svůj počítač nebo herní konzoli, na které může měřit sílu, dovednosti a zručnost s programem.

### 3.2.2 Hra více hráčů (Multiplayer)

Většina her mívá mód jak pro jednoho hráče tak pro více hráčů. Nebývají zde ovšem v podstatě rozdíly, jen soupeřem se místo algoritmu stává lidská bytost. Toto vylepšení dělá hru



samozejmě mnohem zábavnější, neboť naprogramované algoritmy nejsou schopny tak pružně reagovat a vymýšlet velké množství strategií a triků.

Jsou zde ale i hry určeny především pro více hráčů. Mezi takové patří například masové online světy (MMORPG) jako například svět *World of Warcraft*.

Více živých lidí na jednom bojišti ovšem nemusí nutně znamenat jen konkurenci a snahu porazit ty ostatní. Existuje typ her zvaný kooperativy (také kooperativní mód), kdy se skupina uživatelů snaží společně porazit počítač.

### **3.3 Hry z grafického pohledu**

#### **3.3.1 2D hry**

Přestože 2D hry nepatří v dnešní době k těm nejčastěji vyvíjeným a propagovaným velkými společnostmi přes reklamní kampaně, stále mají své místo ve virtuálním světě. Jedná se především o minihry s jednoduchými pravidly, které hrajeme stále dokola ve snaze vylepšit si skóre. Na internetu je takovýchto her zdarma víc než milion kusů a mnoho z nich se těší mezi lidmi oblibě. Pro takto jednoduché aplikace je 2D pohled plně dostačující a třetí rozměr rozhodně nechybí.

#### **3.3.2 3D hry**

Občas si zahrát 2D hru na odreagování je určitě zajímavé, ale pokud máte čas, tak zážitku z pořádné 3D záležitosti se to prostě nevyrovná.

Tento pohled se spoustou detailů, dodává hrám nádech realističnosti a může to hráči připravit parádní zážitek.

Nejčastějším zobrazením scény bývají již zmiňované pohledy první a třetí osoby.



## 4 .NET FRAMEWORK, C#, XNA

Před začátkem programování počítačové hry je potřeba se rozhodnout, na jaké architektuře bude postavena, v jakém programovacím jazyce bude tato hra napsána a jaký bude zvolený standard pro grafické součásti. Možností je zde opravdu mnoho. Při programování grafických prvků lze zvolit OpenGL standard, který je možno použít na jakékoliv platformě včetně OS Linux a MacOSX. Tento standard může pomoci knihoven rozšiřovat kdokoliv. Výtvor musí jen projít potřebnými testy.[10]

Jestli ovšem není cílem vyvíjet pro jiný operační systém než Windows, může být lepší volbou DirectX. Toto API (Application Programming Interface – rozhraní pro programování aplikací) se používá ve většině počítačových her a to především díky knihovnám nejen pro grafickou část, ale i pro zvuk a komunikaci po síti.

Mnou naprogramované hry budou vytvořeny pro počítače s Windows. Rozhodl jsem se tedy vyzkoušet nástroje, které sám tvůrce tohoto systému pro tuto příležitost nabízí.

Aplikace jsou naprogramovány v programovacím jazyce C# (čteno - „sí šarp“). Použitým API je DirectX. Zvoleným vývojovým prostředím (IDE – Integrated Development Environment) je Visual Studio 2010. Hry tedy poběží na platformě .NET (čteno - „dot net“).

Microsoft ovšem v podpoře vývoje her pokročil ještě o kousek dál a vydal Microsoft XNA platformu pro usnadnění tvorby her výhradně pro zařízení s operačním systémem Windows, Xbox 360 a přístroje s operačním systémem Windows Phone 7. Bude tedy také využito.

### 4.1 C#

Programovací jazyky zprostředkovávají komunikaci mezi programátorem a počítačem. Mimoto se pokouší zautomatizovat a usnadnit řešení některých základních úloh, které se dříve musely složitěji psát neustále znovu a znovu.

C# je objektově orientovaný jazyk vytvořený pro vývoj pod systémy Windows. Jeho syntaxe má značnou podobnost s populárními jazyky JAVA, C a C++. Základní programovací prvky jako třídy, deklarace funkcí, výrazy, řízení toku programu, atd. jsou natolik totožné, že tento jazyk spadá, jak je již z názvu patrné, do rodiny jazyků C. Nicméně oproti svým rodinným předchůdcům přináší několik nových věcí. Jedna z nich je tzv. Garbage Collector. Ten dovoluje vývojářům ulehčit si práci se správou paměti, která bývá často kamenem úrazu. Mimo jiné také C# ověřuje bezpečnost kódu při kompilaci a upozorňuje na širokou škálu potenciálních chyb a rizik.

Takováto vylepšení ovšem přinášejí i svá úskalí. Jednou z nevýhod je, že C# byl navržen hlavně pro práci s .NET frameworkem, není tedy primárně určen pro jiné operační systémy než Windows. I přes toto omezení bývá nejvíce vytýkanou zlou vlastností rychlost. Velká část programátorů v C++ to uvádí jako hlavní argument proč C# nepoužívat.

Nicméně v dnešní době kdy se výpočetní rychlost přístrojů stále zvyšuje, to možná není tak relevantní.

Každý vývojář má své priority a podle toho si zvolí jazyk, který je mu bližší.[11]

### 4.2 .NET framework

#### 4.2.1 Základní charakteristika

Platforma .NET nahrazuje starší COM technologii novějším způsobem řešení problémů s komunikací mezi aplikacemi. První verze byla oficiálně vydaná v roce 2002. V roce 2010 pak byla vydána velice populární verze .NET 4 a nakonec v roce 2012 s příchodem Windows 8 momentálně nejaktuálnější .NET 4,5.

Nejdůležitější částí celé .NET platformy je jádro, tzv. .NET Framework. To je run-time. Nachází se v něm objektově orientované knihovny a pomáhá při vývoji webových stránek a aplikací s uživatelským rozhraním.[12]

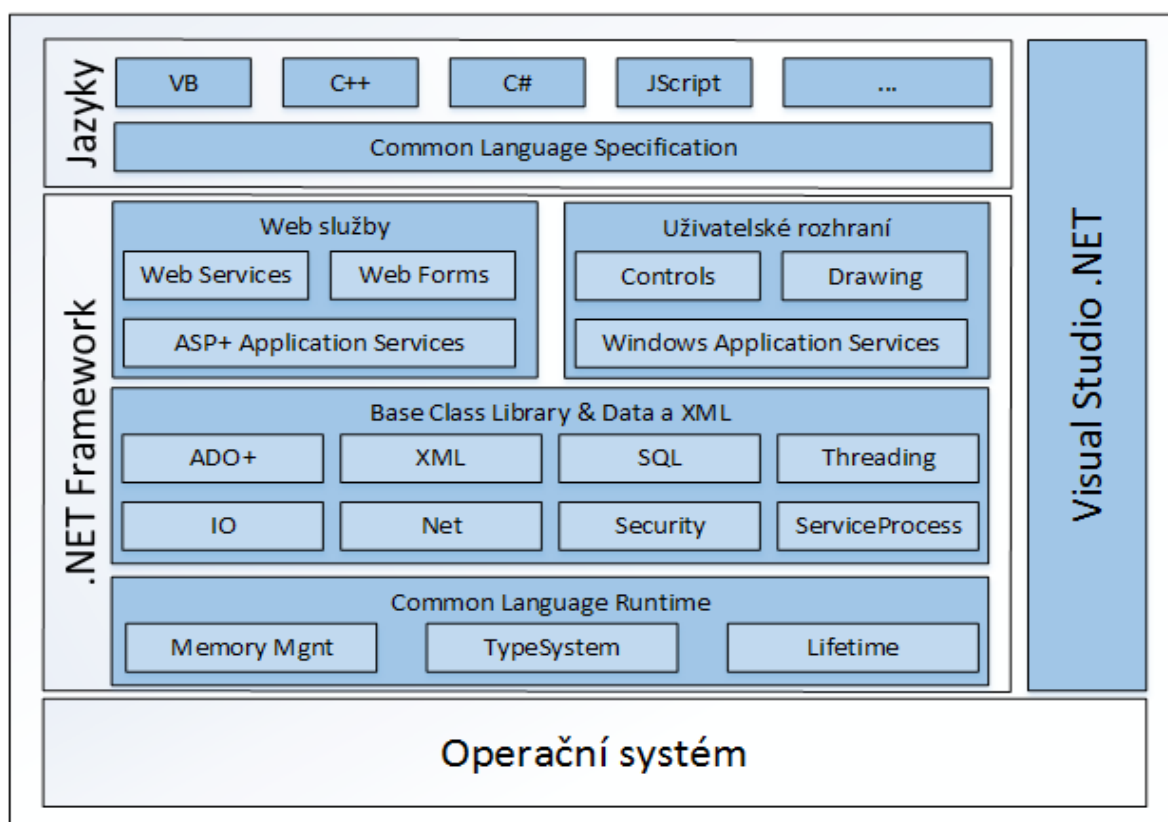
K výhodám .NET platformy, na kterých si zakládá patří:

1. *objektově orientované programování* – celý .NET je postaven na myšlence objektově orientovaného programování už od svých základů a poskytuje tedy tomuto přístupu k řešení

- problému maximální podporu
2. *dobrá architektura* – základní knihovny tříd jsou navrhovány velice intuitivně a programátoři i s menšími zkušenostmi se v nich zvládnou orientovat
  3. *jazyková nezávislost* – přestože C# je s platformou úzce provázaný, je možné použít jakýkoliv programovací jazyk, který splňuje stanovené normy
  4. *podpora webových služeb* - .NET předkládá plnou podporu i v oblasti vývoje pro web a snaží se usnadnit práci a ušetřit čas
  5. *Visual Studio* – toto vývojové prostředí je s platformou velmi úzce spjato a zaručuje příjemnou oporu, velice dobře zvládá způsob ladění programu a popis vzniklých chyb

#### 4.2.2 .NET vrstvy

.NET je velké prostředí, které se stará o vše, od komunikace s operačním systémem až po usnadnění vývoje aplikací a zaručení jejich bezpečného chodu. Je sestaven z různých nezávislých vrstev, komunikujících mezi sebou, které mu tyto vlastnosti umožňují.



Obr. 1 .NET architektura

Nejnižší vrstvu, komunikující přímo se systémem, tvoří Common Language Runtime (CLR). Jednou z jeho součástí je již zmiňovaný Garbage Collector, odstraňující nečinné objekty. Hlavním úkolem CLR je poskytnout velké, bezpečné prostředí pro běh všech aplikací.

Vyšší část tvoří knihovny pro pohodlnou tvorbu aplikací a přístup k datům. Nad nimi se nachází další vrstva usnadňující vývoj webových a uživatelských programů. Tyto knihovny nejsou nijak závislé na jazyce. Nejvyšší vrstvu pak tvoří právě programovací jazyky. Podporovaný je jakýkoliv jazyk splňující standart Common Language Specification.

Skrz celou tuto architekturu prochází vývojové prostředí Visual Studio, poskytující oporu ve všech vrstvách a oblastech.[13]

## 4.3 XNA Framework

### 4.3.1 Úvod do XNA

V dobách, kdy hlavním operačním systémem na počítačích byl DOS, museli programátoři při programování her komunikovat pomocí nízko úrovněvého kódu přímo se zvukovou kartou, grafickou kartou a dalším hardwarem. Tento způsob zvyšoval riziko vzniku chyb a hry na jednom zařízení fungovaly, ale na jiném už nemusely.

S vydáním Windows 95 se objevil v této oblasti nový problém. Nebyl v nich umožněn žádný přístup k hardwarovým komponentám. Microsoft tedy přišel s řešením a vydal DirectX vzápětí po Windows 95. Hlavním cílem bylo sjednotit způsob komunikace s hardwarem a usnadnit tak práci programátorům. Nejen tvůrci her se díky tomu mohli začít soustředit na to, co je opravdu důležité, jádro programů, v tomto případě tvorbu herní logiky.

XNA pokračuje v této myšlence a jde v tomto směru ještě o kousek dále. Jedná se o framework, který si bere základní myšlenky z DirectX a některé části s ním i sdílí. Je ovšem postaven od základu nově a přináší vývojářům spoustu užitečných věcí navíc. Programátor se nemusí zajímat, na jakém přístroji a hardwaru jeho hra poběží, má podpůrnou sadu knihoven pro práci se vstupními komponentami jako je klávesnice, myš, gamepad a joystick. Jsou zde vyřešeny problémy s minimalizací herního okna a módem celé obrazovky (fullscreen), je zde také možnost uložit aktuální údaje o hře a jednoduše je znovu obnovit.

XNA je určen jednotlivcům nebo malým týmům, není v něm totiž zabudována podpora pro vývoj ve velkých týmech, proto se o tento produkt momentálně nezajímají profesionální firmy, zabývající se herním vývojem. I v této oblasti se má však výhledově pokročit.

Abychom mohli v XNA tvořit vlastní výtvořky je nutné, kromě .NET frameworku a XNA frameworku mít nainstalováno také podporované IDE. Jedinou možnou jsou Microsoft Visual Studio nebo jeho bezplatná verze Visual Studio Express. Programování je momentálně přípustné jen v jazycích Visual Basic a C#.

Informace o XNA uvedeny v celé této kapitole byly čerpány z: [14][15]

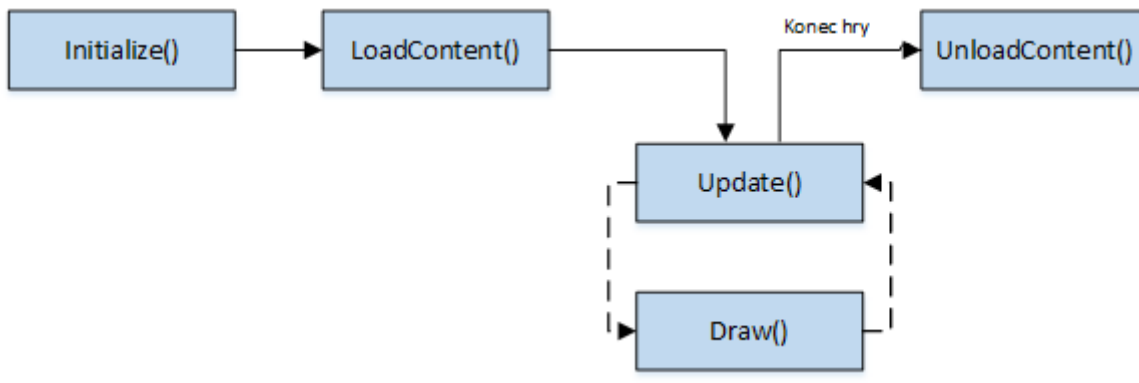
### 4.3.2 Programování v XNA

Hlavním rozdílem mezi typickým programováním aplikací a psaním her je interakce s uživatelem. V běžné aplikaci zobrazíme uživateli okno a čekáme na jeho reakci. Tou může být stisk tlačítka, kliknutí myši, atd.

Ve videohrách tyto akce také zachytáváme a podle nich měníme průběh hry, nemáme ovšem možnost na ně čekat. Musíme se postarat o to, aby se pohybovali nepřátelé, naše auto, jedoucí kupředu, vidělo stále novou krajinu, případně havarovalo nebo zastavilo. Bez ohledu na to, jestli hráč něco dělá nebo ne, hra musí být v pohybu a plnit svůj účel. Pokud člověk odejde na 10 minut od programu, nic se pravděpodobně nezmění. Odejde-li ovšem od počítačové hry, zpravidla ho přivítá oznámení o ukončení hry. Samozřejmě toho často pod oponou hry probíhá více, než je aktuálně nutné. To je také hlavním důvodem, proč jsou hry tak náročné na výpočetní hardware. Pro správnou funkci je to bohužel nezbytné.

Problém této „živosti“ aplikace je řešen pomocí nekonečné smyčky, která probíhá stále dokola, dokud není hra ukončena. V XNA je tato herní smyčka přímo součástí architektury. Konkrétně se jedná o koloběh mezi metodou *Update*, aktualizující veškeré změny, a metodou *Draw*, vykreslující herní scénu.

Celý životní cyklus hry v XNA je zobrazen na obrázku číslo 2.



Obr. 2 Životní cyklus hry v XNA

Při startu programu je nejprve volána metoda *Initialize*. V té jsou načteny různé počáteční hodnoty, jako skóre a údaje správného počátečního nastavení. Následuje metoda *LoadContent*, která načte do hry veškeré grafické prvky a postará se, aby hra správně komunikovala s grafickou kartou. Kromě grafické části hry se zde také načítají zvukové stopy.

Následuje již zmíněná herní smyčka, kdy se opakuje volání metod *Update* a *Draw*. XNA se pokouší, aby tato smyčka proběhla šedesátkrát za sekundu na Xboxu360 a počítači. V případě aplikací pro Windows Phone pak třicetkrát za sekundu, z důvodu méně výkonných procesorů v mobilních telefonech. Zvýšit frekvenci volání by bylo naprosto zbytečné, lidské oko nedokáže takový rozdíl rozeznat. Mnohem častěji se stává, že je frekvence volání nižší. Pokud klesne příliš, hra se nevykresluje dostatečně často a my to vnímáme jako sekání.

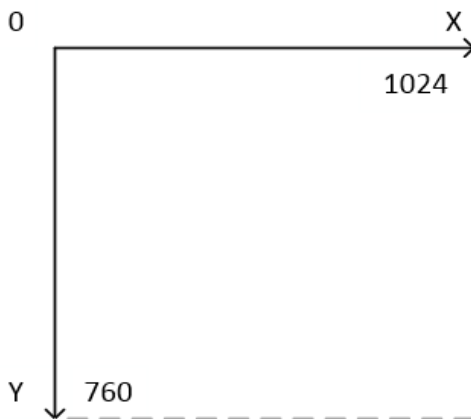
Pokud hru ukončíme, proběhne program už jen přes poslední metodu *UnloadContent*, kde uvolní všechny dříve vytvořené objekty. Hra skončila a všechny nabyté zdroje jsou vráceny zpět systému.

### 4.3.3 Grafická část XNA

Nezanedbatelnou část každé hry, tvoří práce s grafickými prvky. I pokud plánujeme vyvíjet jen 3D hry, bez 2D grafiky se pravděpodobně neobejdeme. Jsou přece jen místa, jako menu, obrazovka s pauzou, zobrazení skóre a jiné, kde se použití jen klasického pohledu nabízí a je plně postačující.

Nejdůležitější věcí, kterou při programování ve 2D pohledu musíme znát, je orientace souřadnicového systému.

Počáteční bod  $[0,0]$  je v XNA vlevo nahoře, načež osa X je vodorovná se směrem zleva doprava a osa Y svislá, orientovaná shora dolů. Jednotkou je pixel (px). Pokud tedy máme okno pro hru o velikosti např. 1024x768 vypadá scéna jako na obrázku číslo 3.



Obr. 3 2D souřadnicový systém v XNA

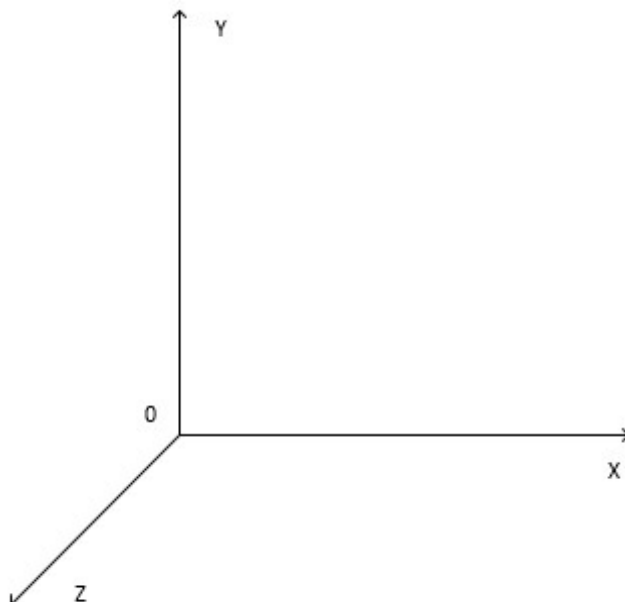
Vykreslení textur (obrázků) je potom provedeno v několika jednoduchých krocích. V inicializační fázi načteme do objektu `2DTexture` naši texturu a v metodě `Draw` ji následně vykreslíme na námi zvolených souřadnicích. K určení souřadnice se v XNA využívají vektory reprezentované objektem `Vector2(X,Y)`.

Podpůrné knihovny nám poté nabízí i velkou škálu metod pro následné úpravy, mezi které patří vyhlazení, zprůhlednění, proložení obrázku libovolnou barvou apod. Zajímavostí je, že barvu zadanou podle RGB barevného modelu jako `(255,0,255)`, vnímá XNA jako „kouzelnou“ a vše co je jí namalováno, je stoprocentně průhledné. Toho lze využít v případě složitějších textur.

Statické obrázky hru ovšem nevytvoří. Proto je zapotřebí aplikaci rozhábat. Animace se provádí buďto změnou pozice textury v metodě `Update` nebo nahrazováním jednoho obrázku následujícím.

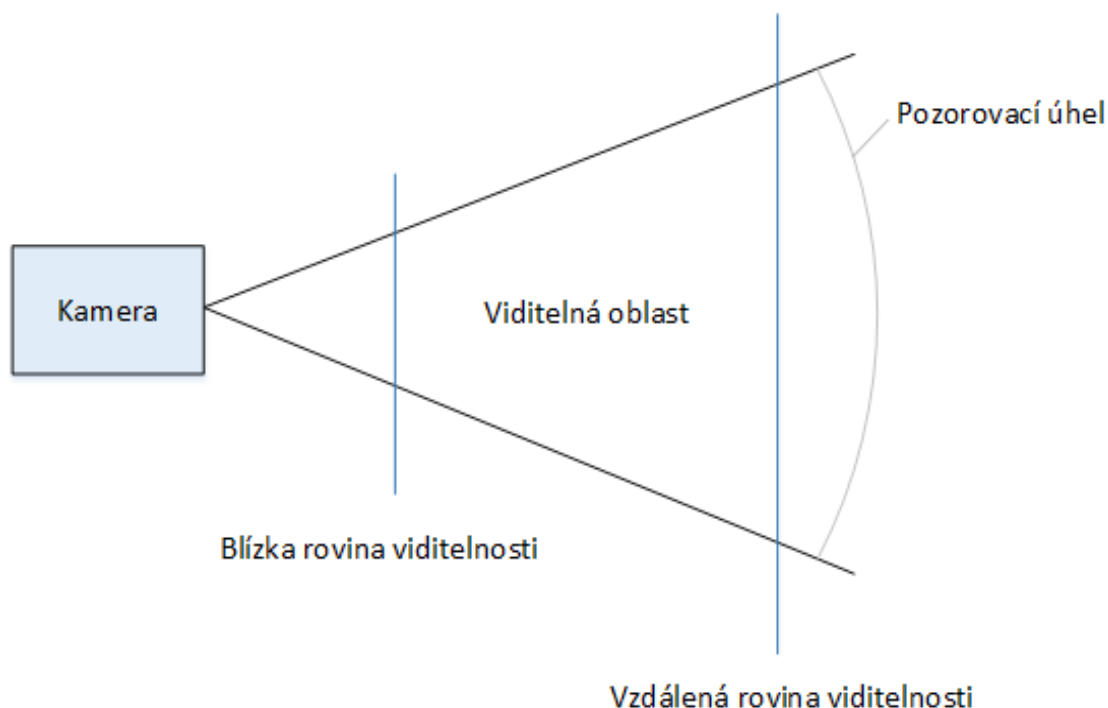
I když je zde, při vývoji 2D prostoru opravdu hodně zajímavých věcí, které jdou dělat, plný výkon grafických karet se tím ani zdaleka nevyužije.

Všechny moderní hry bývají tedy tvořeny ve třech dimenzích. Přidáním jednoho rozměru se v XNA mění jedna podstatná věc, souřadnicový systém. Počátek `[0;0;0]` je umístěn v levém dolním rohu. Osa X je stejně orientovaná jako ve 2D zobrazení. Osa Y roste kladně zdola na horu a osa Z vede směrem k divákovi. Jedná se tedy o pravotočivou kartézskou soustavu souřadnic.



Obr. 4 3D souřadnicový systém v XNA

Nicméně i když umístíme předmět do nulového bodu, není jasně dáno, kde na obrazovce se přesně vykreslí. V dvojrozměrném prostoru stačilo umístit obrázek na určité souřadnice a vše bylo hotovo. V trojrozměrném zobrazování je to mírně složitější. Zobrazování 3D objektů je totiž sestavováno ze dvou základních komponent. Objektu, který chceme pozorovat a kamery, pomocí níž objekt pozorujeme. Kameře musíme nastavit polohu ve světě, směr natočení, minimální a maximální viditelnou vzdálenost a velikost pozorovacího úhlu. Tyto údaje se ukládají do projekční matice. Její vytvoření se provádí voláním statické metody `Matrix.CreatePerspectiveFieldOfView`. Tímto způsobem se tedy vytvoří vymezená oblast komolého kuželu. Všechny objekty umístěné v prostoru uvnitř jsou vykreslovány podle logických principů, vzdálenější prvky jsou menší než ty, co se nacházejí blíže ke kameře. Mimo tuto oblast se nevykresluje nic. Tato vykreslovací oblast je názorně předvedena na obrázku 5.



Obr. 5 Zobrazovací kužel pro 3D objekty

Rozpohybování je prováděno stejným způsobem jako ve 2D prostoru, čili se hýbe s objekty, popřípadě je jeden objekt nahrazován jiným. Navíc je zde také možnost místo jednotlivými předměty pohybovat s celou kamerou a měnit tak úhel pohledu a zobrazovaný prostor. Tohoto pohybu využívají například letecké simulátory.

Jak již bylo uvedeno, základním kamenem 3D her jsou objekty. Pro tvorbu jednoduchých 3D objektů lze využít podporu XNA. Jsme tak schopni vytvořit jednoduché tvary, jako krychli, kouli, válec, atd. Ty poté pokryjeme barvou nebo texturou. Skládat složité obrazce tímto způsobem není ovšem příliš praktické. Takovéto modely se tvoří v profesionálních nástrojích jako je Maya nebo 3D Max a do XNA se potom nahrávají v metodě *LoadContent*. Naučit se pracovat v těchto nástrojích není naštěstí pro tvorbu her nutné. Na internetu lze stáhnout velké množství modelů, které dali jejich tvůrci volně k použití a pro vyzkoušení principů tvorby her to plně postačuje. Profesionální týmy tvořící hry mají na tuto práci ve svých řadách grafiky specialisty.

XNA je tedy framework, vytvořený firmou Microsoft, pro tvorbu videoher běžících na systémech Windows. Velice zpříjemňuje a zrychluje vývojovou činnost pomocí sady knihoven a tříd. Je velice vhodným nástrojem jak pro začínající herní programátory, tak pro pokročilé vývojáře.



## 5 ŘEŠENÍ

Ted, když jsem blíže uvedl použité nástroje, představím vytvořené ukázkové hry, jenž v nich byly vytvořeny.

Celkem se jedná o tři herní tituly, každý s rozdílným prvkem oproti ostatním. Všechny mají v designu zakomponována loga fakulty strojního inženýrství a jeho ústavu automatizace a informatiky, aby splňovaly požadavek na propagaci těchto institucí.

### 5.1 Had

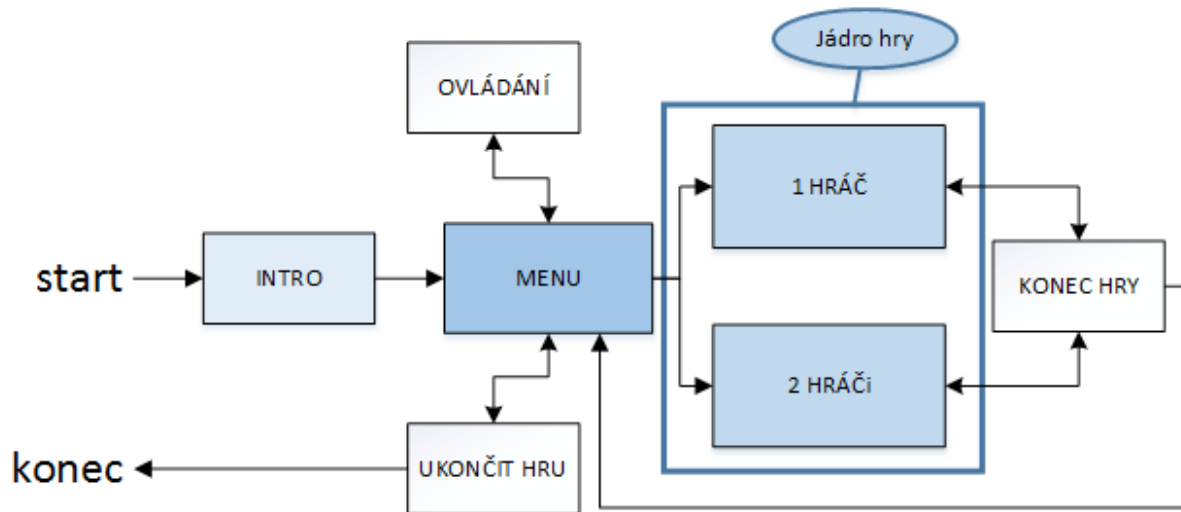
#### 5.1.1 Pravidla hry

Tato hra je velice známá. Hlavním herním prvkem je had, který se snaží pozřít potravu a díky tomu růst a zvedat skóre. Nesmí ovšem narazit při svém pohybu do rozmístěných zdí ani do vlastního těla.

Jelikož má mnou vytvořená hra sloužit jako ukázka, omezil jsem herní čas na tři minuty a úkolem hráče je nechat hada co nejvíc vyrůst pouze za tento časový interval. Pro zpestření byla do hry naimplementována také hra dvou hráčů. V té je úkolem každého hráče, donutit protivníka aby se svým hadem narazil jako první.

#### 5.1.2 Architektura hry

Had je naprogramován pomocí 2D grafiky, ovládání je možné výhradně pomocí klávesnice. Celá tato hra se dělí na sedm nezávislých částí, podle toho, co se právě vykresluje na obrazovce. Přejít mezi jednotlivými fázemi hry se řídí striktními pravidly a je umožněn jen danými směry. Nastává z pravidla po příchozím příkazu od uživatele. Celý životní cyklus je zobrazen na obrázku 6.



Obr. 6 Had – životní cyklus

Zařadíme-li tento děj hry do životnosti celého XNA, pak za start se považuje konec metody *LoadContent*, koncem je myšleno zavolání metody *UnloadContent* a následné ukončení aplikace. Zbylá část programu se odehrává pouze v herní smyčce mezi metodami *Update* a *Draw*.

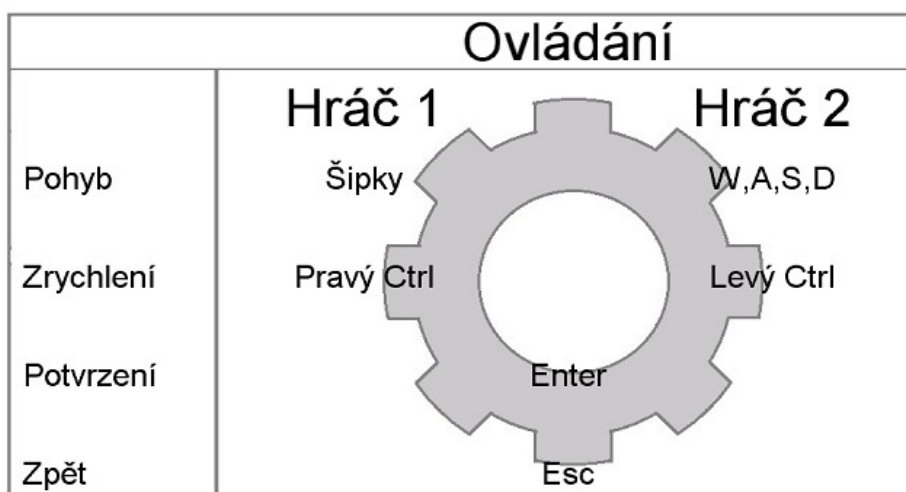
Jádrum hry je funkčnost během aktivní herní fáze. Ostatní části jsou ve hře zakomponovány jen z informativních a estetických důvodů. Nicméně i tyto části jsou podstatné, aby hra budila celkově mnohem lepší dojem. Informace o tom, která obrazovka se aktuálně vykresluje, uchovává třída *GameManager*.

### 5.1.3 Rozbor algoritmu

Ihned po startu se načte úvodní část Intro. Zde se spustí animace, při které se po časový interval posouvá a rotuje 3D objekt. Čas určující tento posun ovšem není reálným časem, neboť pokud by byl počítačový procesor příliš vytížený, mohl by se čas ve hře mírně lišit od skutečného. Pohyb by tedy nebyl plynulý, případně by působil značně nepřesně. Použitý čas pro výpočet translace a rotace, je dobou od posledního volání metody *Update* ve hře. Toto zaručuje korektnost i v případě nečekaného zatížení herní sestavy.

Po uplynutí časového intervalu určeného pro úvod, popřípadě po stisku klávesy escape, je hra přesměrována do Menu. Zde má uživatel možnost pomocí šipek a následného potvrzení klávesou enter možnost postoupit ve hře zvoleným směrem. Tuto funkčnost zajišťuje v algoritmu třída *Menu*. Tato třída obsahuje pole s prvky, které menu tvoří a soukromé metody zachycující stav klávesnice a následně umožňující změnu označeného prvku z pole a přesměrování hry daným směrem.

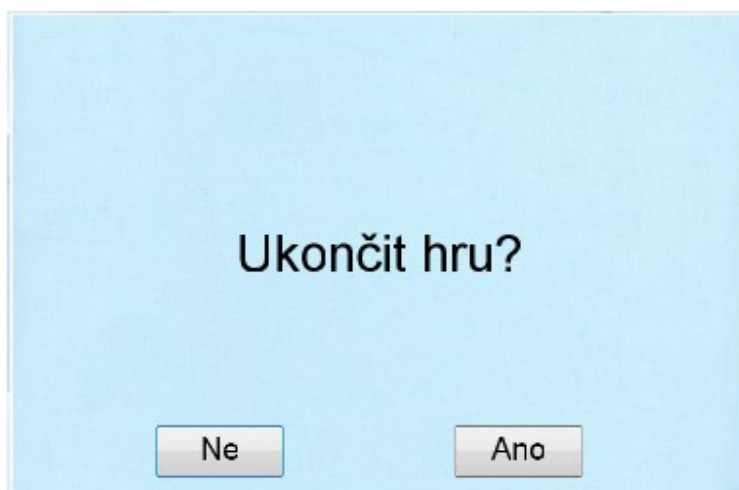
Poslední podpůrnou část programu tvoří tři stavy, Ovládání, Ukončení hry a Konec hry. Všechny tyto tři části vykresluje třídy dědicí své vlastnosti od stejné mateřské třídy *Screen*. Ta obsahuje vlastnosti uchovávající veškeré potřebné informace o pozadí, velikosti a barvě písma, umístění textu a seznam potřebných tlačítek. Pomocí veřejných metod je potom možné buďto přidávat nebo editovat text a přidat potřebná tlačítka. Nastavit pozadí je možné jen při volání konstruktoru. Poslední dvě metody, určené ke komunikaci s okolím jsou veřejné metody *Update* a *Draw*. Metoda *Update* kontroluje stav klávesnice a při stisku správných kláves reaguje, metoda *Draw* vykresluje všechny potřebné informace na obrazovku. Výsledek viz. obrázky 7, 8 a 9.



Obr. 7 Ovládání



Obr. 8 Konec hry



Obr. 9 Ukončení hry

I přesto, že bez již zmíněných částí by se hra neobešla, nejdůležitější část kódu probíhá při aktivních herních oknech. Jakmile se hra posune do herního módu pro jednoho hráče, vytvoří se objekt třídy *Level*. Tento objekt se postará o správné vykreslení zdi. Momentálně není třída *Level* nutná, neboť hra obsahuje pouze jednu úroveň. Počítá se ovšem s možností budoucího rozšíření počtu úrovní, z toho důvodu je do architektury zabudována už teď.

Jakmile máme naimplementovány zdi, začne hra pracovat s asi nejdůležitější třídou, *Snake*. V té jsou obsaženy veřejné metody *Draw*, *AddSegment* a *Move*. Dále jsou zde definovány soukromé virtuální metody *bodyMove*, *keyboard* a *initialize*.

Virtuálnost je zde nutná, protože od třídy *Snake* se dědí třídy *MultipleSnake1* a *MultipleSnake2* použité pro hady ve hře dvou hráčů. V tomto módu je ovšem jiná herní logika. Z důvodu její aplikace se tedy musí zmíněné virtuální metody přepsat.

Metoda *keyboard* získává údaje z klávesnice a umožňuje změnu směru a rychlosti. Pro jednoho hráče je použito jen ovládání pomocí šipek, v případě dvou hráčů se had druhého hráče pohybuje pomocí kláves W,A,S,D.

V metodě *initialize* se nastaví počáteční pozice a délka hada. Pokud by byl aktivní druhý mód, mění se ypsilonová hodnota náhodně. Zde je statická.

Důležitá metoda *bodyMove* se stará o to, aby tělo poslušně následovalo svou hlavu. Zjistí tedy minulou polohu předchozího článku v hadově těle a následně se do ní posune. U dvou hráčů se tělo nijak neposouvá, ale stále se vytvářejí nové články hadova těla.

Jedinou funkcí metody *Move* je pohyb hlavou hada. Hlava se vždy posune o svou velikost, určeným směrem. Jelikož je tato metoda volána z XNA *Update*, proběhne šedesátkrát za vteřinu. Pokud by tedy translace hadovy hlavy nebyla nijak omezena, ihned po spuštění by hra skončila. Had by totiž narazil do zdi dříve než by to naše oko postřehlo. *Move* tedy také snižuje frekvenci vykonání pohybu na jeden posun za osmdesát milisekund v případě běžného pohybu a jeden posun za padesát milisekund při zrychlení.

Pomocí *AddSegment* se přidává článek hadova těla a nakonec *Draw* vše předchozí vykresluje.

Po nastavení celého hada a všech zdí, vygeneruje objekt třídy *Food* potravu na náhodné pozici. Jídlo se ovšem nesmí objevit ve zdi nebo těle.

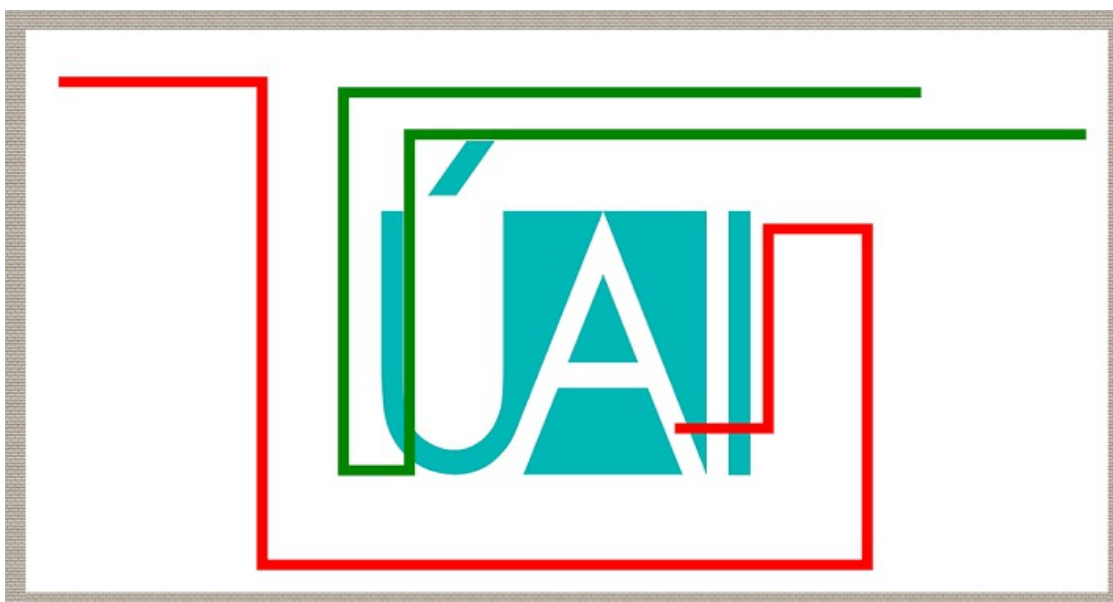
Celou herní obrazovku doplňují ještě informace o času a dosaženém skóre, které jsou obsaženy v komponentě *ScoreAndTime*.

Jádro aplikace poté jen kontroluje, jestli byla snědena potravina, v tom případě vytvoří novou, nebo jestli není nutné hru ukončit a přejít do fáze Konec hry.

Při hře dvou hráčů, kromě výše zmíněných rozdílů, odpadá prvek jídla, času, skóre a středových zdí.



*Obr. 10 Had – hra pro jednoho hráče*



*Obr. 11 Had – hra pro dva hráče*

## 5.2 Pexeso

### 5.2.1 Pravidla hry

Pexeso je další velmi populární hrou pro dvojici s jednoduchými pravidly. Základem jsou šedesát čtyři kartičky se shodným rubem. Svým obrázkem na lici se ovšem shoduje jen třicet dva párů. Úkolem každého hráče je nasbírat co nejvíce stejných dvojic.

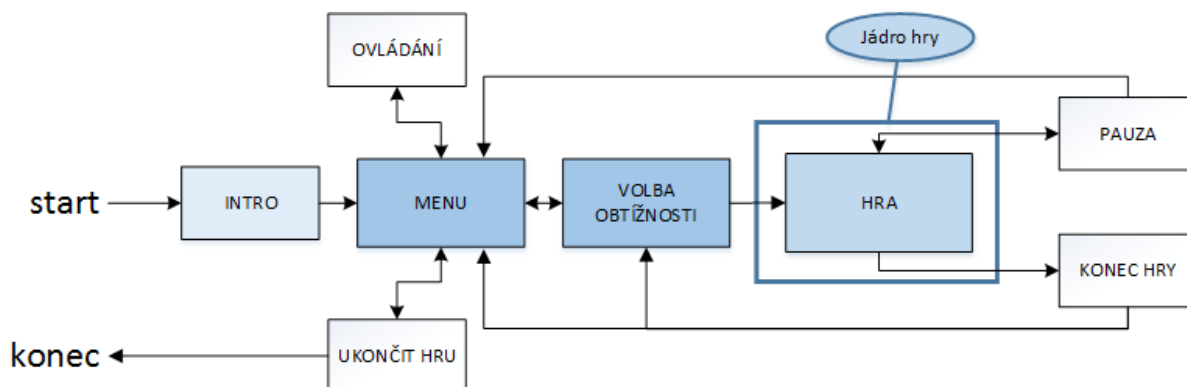
Hra je určená pro jednu osobu, protihráče zde tvoří umělá inteligence s nastavitelnou herní silou.

### 5.2.2 Architektura hry

I když to není na první pohled patrné, je toto pexeso naprogramováno pomocí 3D grafických prvků. Ty jsou zde zastoupeny v podání herních kartiček, které při otáčení rotují kolem své středové osy.

Navíc je v pexesu, na rozdíl od hada, v průběhu jakékoliv herní části umožněno ovládání nejen za pomoci klávesnice ale také myši.

Dalším rozdílem je úbytek možnosti více hráčů a přibyly dva nové podpůrné stavy hry. Jedná se o Volbu obtížnosti a Pauzu.

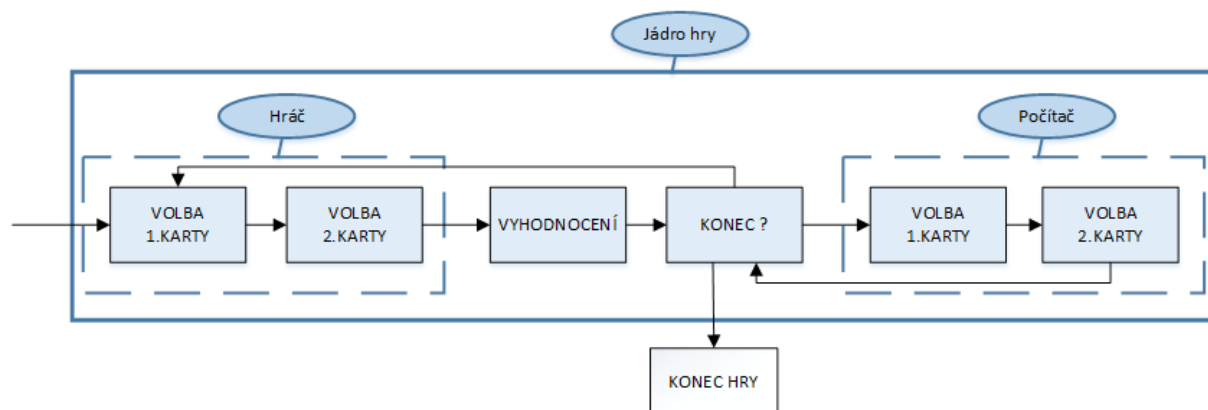


Obr. 12 Pexeso – životní cyklus

Ve stavu Volba obtížnosti, se stará o funkčnost sesterská třída, třídy *Menu*, *PCDifficultMenu*. Funkčnost je naprosto stejná jako v případě třídy *Menu*, jen přesměrovává hru do jiných částí.

Třída *Pause*, jež je potomkem třídy *Screen* a má stejné vlastnosti jako její výše uvedené sestry, vykresluje obraz během Pauza stavu.

Jakmile se hra přehoupne do svého jádra, první věcí, která se vykoná, je zamíchání a rozdání karet. Následuje herní cyklus zobrazený na obrázku číslo 13.



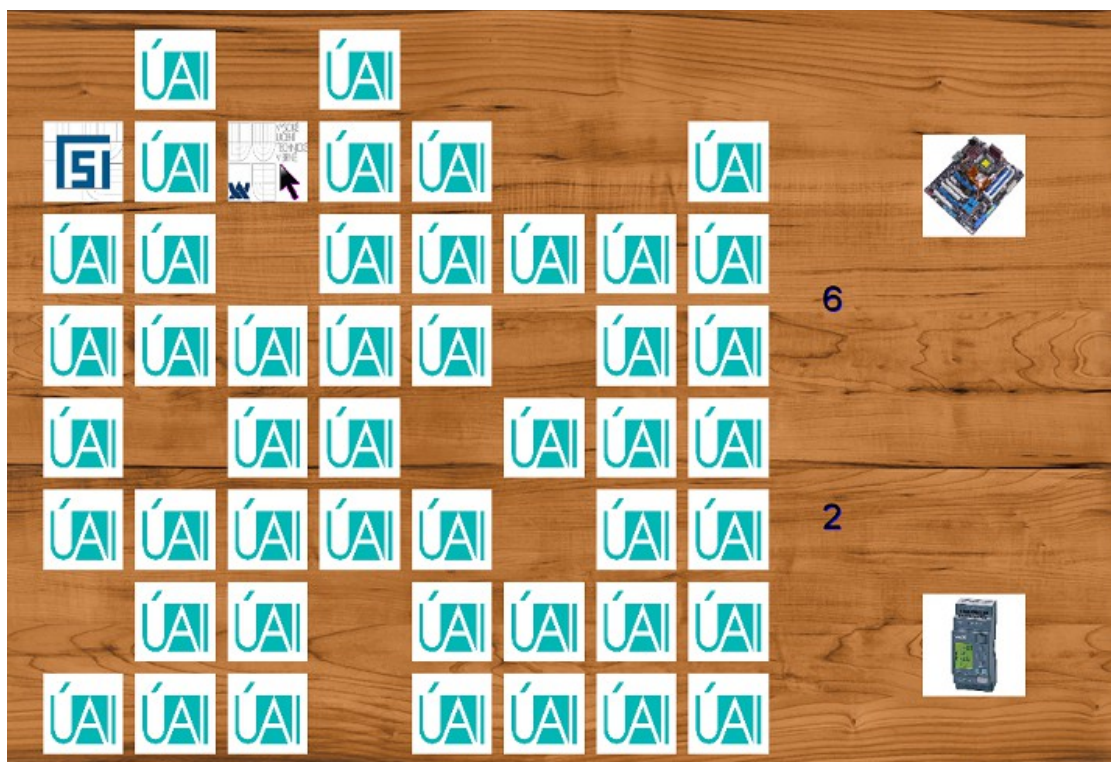
Obr. 13 Pexeso – architektura jádra hry

Během počáteční fáze si hráč vybere první kartu a otočí ji. Následuje další hráčova fáze, během které se zvolí a otočí druhá karta. Uživatele je nutno během obou fází kontrolovat, zda se nepokouší například otočit již zobrazenou nebo vyřazenou kartičku. Po úspěšném průchodu, přes kontrolní algoritmus se hra dostane do fáze vyhodnocovací.

Zde se nejprve vyhodnotí, jestli se hráči podařilo najít dva stejné symboly, pokud ano zvedne se jeho skóre, otočené karty se vyřadí ze hry a majitelem následujícího tahu je opět uživatel. V případě, že hráč úspěch nezaznamenal, jsou kartičky otočeny opět rubem vzhůru a jako vlastník budoucího tahu se nastaví počítač. Následně se zkontroluje případné ukončení hry. Nedošlo-li k němu, pokračuje se v předem určeném směru zpět do fáze jedna nebo zahájí svůj tah počítač v páté fázi. Během ní otočí svou první kartu lícem vzhůru, následně hra pokročí do své šesté a poslední fáze. Po otočení druhé karty nastane opět třetí, vyhodnocovací fáze a celý algoritmus se v těchto směrech pohybuje tak dlouho, dokud není ukončen.

Grafický výsledek je na obrázku 14.





Obr. 14 Pexeso – hlavní herní okno

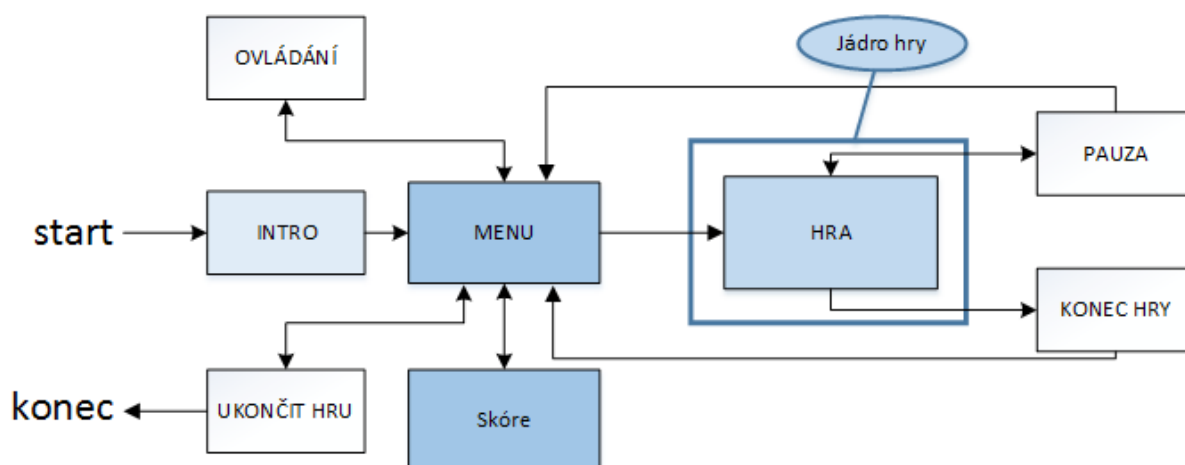
### 5.3 Skákací kulička

#### 5.3.1 Pravidla hry

Ve hře nazvané Skákací kulička má hráč za úkol posbírat všechny předměty a nespadnout přitom z plošinek. Cílem je splnit tento požadavek v co možná nejkratším časovém intervalu.

#### 5.3.2 Architektura hry

Hra se ovládá, stejně jako had, výhradně pomocí klávesnice. V životním cyklu přibyla možnost Skóre, kde se uchovává deset nejrychlejších časů a kdy jich bylo dosaženo. Životní cyklus je vyobrazen na obrázku 15.



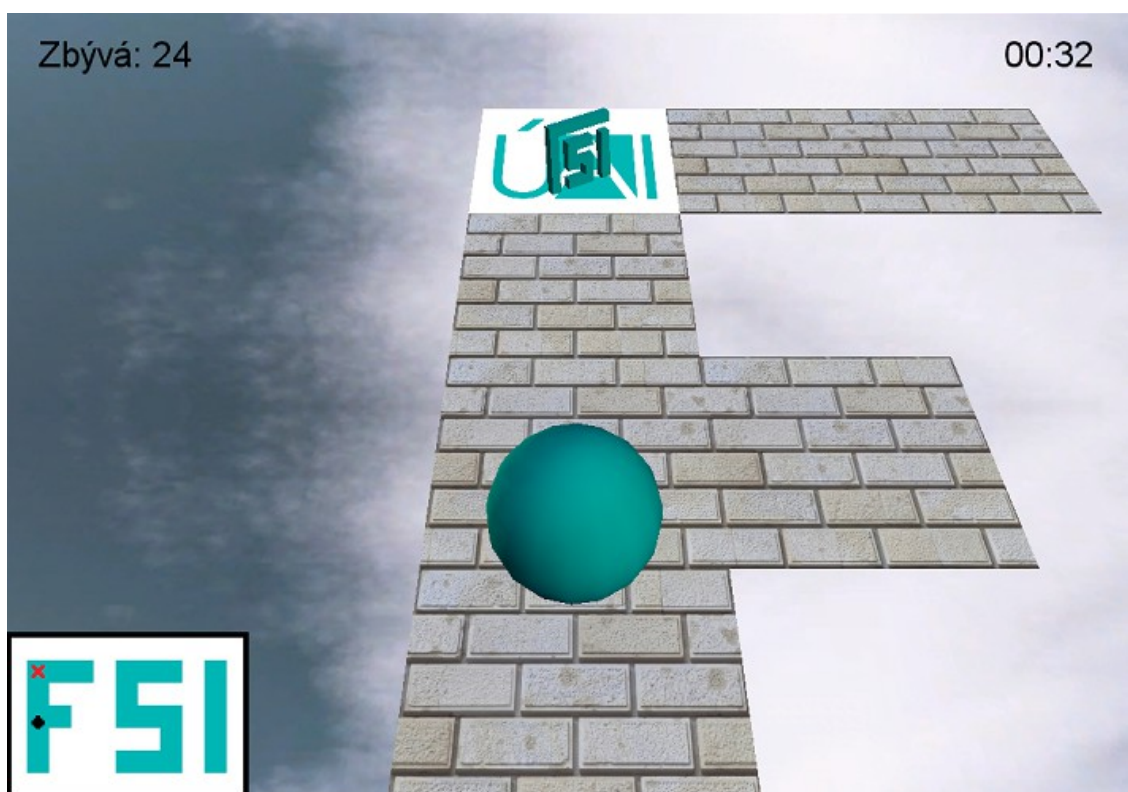
Obr. 15 Skákací kulička – životní cyklus

Jádro hry opět zaznamenalo velkou vzhledovou změnu. Na první pohled je patrné, že je zde použita 3D grafika a také pohyblivá kamera. Ta se pohybuje společně s hráčem ovládanou kuličkou a vykresluje aktuální scénu.

O práci s 3D modely se stará třída *MyModels*, obsahující nejen způsob vykreslování modelů na určených souřadnicích, ale také například zjišťuje nejmenší možný prostor, do kterého se daný objekt vejde. To je nutné z důvodu interakce mezi modely, při jejich sbírání, a také ke zjištění zdali je kulička stále na plošinkách. Dědicové této základní třídy, starající se o modely, si poté přidávají pár dalších metod, zodpovědných za pohyb. V případě třídy *ElementModel* je to rotace sbíraného předmětu a ve třídě *HeroModel* pak pohyb hlavní kuličky pomocí klávesnice, skok a pád v případě vyjetí mimo trať.

Do herního okna je také přidána malá mapa, zobrazující rozmístění hracích desek a aktuální polohu hrdiny a hledaných předmětů. O tuto funkcionalitu se stará komponenta *Map*.

Celá herní scéna je pak završena, mírně upravenou, komponentou *ScoreAndTime*. Oproti pexesu zde čas narůstá a skóre odpočítává zbývající počet předmětů. Celé toto vyobrazení viz obrázek číslo 16.



Obr. 16 Skákací kulička – hlavní herní okno

Zbylé části hry jsou podrobně popsány v předchozích hrách.





## 6 ZÁVĚR

V rámci této bakalářské práce byly vytvořeny tři ukázkové hry, do nichž byla v designu zakomponována loga fakulty strojího inženýrství a jeho ústavu automatizace a řízení kvůli propagaci.

Tyto aplikace byly vytvořeny ve vývojovém prostředí Visual Studio 2010 za pomoci XNA. Tento nástroj příjemně zjednodušuje vývojářskou část práce a pomáhá překonat některé běžné problémy spojené s tvorbou her, jako je komunikace s hardwarem.

Všechny vytvořené hry jsou určeny pro počítače s operačním systémem Windows. Budou volně šiřitelné prostřednictvím internetu, kde se pokusí ukázat, co vše je možné naučit se na vysoké škole a následně také přilákat větší počet zájemců o studium informatiky.

## SEZNAM POUŽITÉ LITERATURY

- [1] BERENS, Kate Geoff HOWARD. *The rough guide to videogames*. New York: Penguin Group, 2008. ISBN 18-435-3995-0.
- [2] KENT, Steve L. *The ultimate history of video games*. New York: Three rivers, 2001. ISBN 07-615-3643-4.
- [3] ŠVÁRA, Ondřej. *Videohry: Historie virtuální zábavy*. Nové Město na Moravě: Vik Lukáš - Databook publishing, 2013. ISBN 978-80-87749-08-1.
- [4] *Giant Bomb - Video game reviews, videos, forums and wiki*. [online]. © 2013 [cit. 2013-04-23]. Dostupné z: <http://www.giantbomb.com/>
- [5] Definition of video game in Oxford Dictionaries (US English). *Oxford Dictionaries Online* [online]. © 2013 [cit. 2013-04-23]. Dostupné z: [http://oxforddictionaries.com/definition/american\\_english/video+game](http://oxforddictionaries.com/definition/american_english/video+game)
- [6] Bibliographic data. *Espacenet* [online]. 13.03.2013 [cit. 2013-04-23]. Dostupné z: [http://worldwide.espacenet.com/publicationDetails/biblio?CC=US&NR=2455992&KC=&FT=E&locale=en\\_EP](http://worldwide.espacenet.com/publicationDetails/biblio?CC=US&NR=2455992&KC=&FT=E&locale=en_EP)
- [7] OVERMARS, Mark. *A Brief History of Computer Games* [online]. 2012 [cit. 2013-05-04]. Dostupné z: [http://www.cs.uu.nl/docs/vakken/b2go/docs/history\\_of\\_games.pdf](http://www.cs.uu.nl/docs/vakken/b2go/docs/history_of_games.pdf)
- [8] WOLF, Mark J. *The Medium of the Video Game*. : University of Texas Press, 2001. ISBN 02-927-9150-X.
- [9] Best of 2012: Hra roku. *GAMES.cz* [online]. 2013 [cit. 2013-05-04]. Dostupné z:
- [10] *OpenGL - The Industry Standard for High Performance Graphics* [online]. 2013 [cit. 2013-05-04]. Dostupné z: <http://www.opengl.org/>
- [11] GRIFFITHS, Ian, Matthew ADAMS a Jesse LIBERTY. *Programming C# 4.0*. Beijing: O'Reilly, 2010. ISBN 978-059-6159-832.
- [12] KAČMÁŘ, Dalibor. *Programujeme .NET aplikace ve Visual Studiu .NET*. Praha: Computer Press, 2001. ISBN 80-722-6569-5.
- [13]. *.NET Tutorial for Beginners* [online]. 2005 [cit. 2013-05-04]. Dostupné z: <http://download.microsoft.com/download/8/e/7/8e725d96-7ec3-498b-9fa7-86779aed101f/dotNET%20Tutorial%20for%20Beginners.pdf>
- [14] CARTER, Chad. *Microsoft XNA unleashed*. Indianapolis: Sams Pub., 2008. ISBN 0-672-32964-6.
- [15] REED, Aaron. *Learning XNA 4.0*. Sebastopol: O'Reilly, 2011. ISBN 978-1-449-39462-2.