

UNIVERZITA PALACKÉHO V OLOMOUCI

PEDAGOGICKÁ FAKULTA

KATEDRA MATEMATIKY



VYUŽITÍ VISUAL BASIC VE STEREOMETRII NA ZTM

DIPLOMOVÁ PRÁCE

Autor: Bc. Blanka EZNÍ KOVÁ

Vedoucí práce: Mgr. David NOCAR, Ph.D.

Olomouc 2014

Prohlášení

Prohláším, že jsem svou diplomovou práci vypracovala samostatně, výhradně s použitím citovaných zdrojů uvedených v příloze.

Souhlasím s tím, že moje práce v souladu s § 60 Zákona č. 121/2000 Sb., ze dne 12.5.2000, o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon.)

V Olomouci dne 17. dubna 2014

í í í í í í í í í í í í í í

Bc. Blanka Štejnčíková

Děkuji Mgr. Davidu Nocarovi, Ph.D. za odborné vedení diplomové práce a poskytování cenných rad při konzultacích.

Obsah

Úvod.....	7
TEORETICKÁ ČÁST	9
1 Pořadí ve výuce.....	9
1.1 Vstup pořadí do škol.....	9
1.2 Využití pořadí ve výuce	10
1.3 Funkce pořadí ve výuce	12
1.4 Negativní důsledky nasazení ICT do výuky.....	13
1.5 E-learning.....	13
1.6 Konkrétní využití pořadí ve výuce	15
2 Výukový software	16
2.1 Historie výukových program	16
2.2 Vymezení pojmu výukový software	17
2.3 Principy účinnosti výukových program	17
2.4 Funkce výukového softwaru	18
2.4.1 Výklad učiva.....	18
2.4.2 Simulace problémů, simulační hry.....	18
2.4.3 Procvičování.....	19
2.4.4 Testování.....	19
2.5 Programové učení.....	19
2.5.1 Lineární výukové programy	20
2.5.2 Vytvářené výukové programy	20
2.6 Výhody a nevýhody výukových program	21
2.7 Kategorizace výukových program	22
2.8 Kritéria hodnocení výukových program	26
2.9 Výběr výukového programu.....	28
3 Tvorba výukového programu.....	29
3.1 Softwarové prostředí.....	29
3.2 Koncepte výuky programovacích jazyk	30
3.3 Volba programovacího jazyku	30
3.4 Postupy k tvorbě výukových program	31
3.4.1 Jednoduchý výukový program.....	31

3.4.2	Univerzální výukový program.....	33
4	Stereometrie na Z TM	34
4.1	Stereometrie v RVP ZV	35
4.2	Stereometrie ve výuce na Z TM	36
5	Visual Basic	39
5.1	Program.....	39
5.1.1	Programovací jazyky a p eklada e	39
5.2	Objektov orientované programování.....	41
5.3	Úvod do jazyka Visual Basic	41
5.4	P ehled vývojových verzí jazyka VB	42
5.4.1	VisualBasic .NET	43
5.5	Vývojové prostředí Microsoft Visual Studio 2010 Express.....	43
5.5.1	Nabídka FILE (Soubor).....	47
5.5.2	Nabídka EDIT (Úpravy).....	49
5.5.3	Nabídka VIEW (Zobrazit).....	50
5.5.4	Nabídka PROJECT (Projekt).....	52
5.5.5	Nabídka DEBUG (Spustit).....	53
5.5.6	Nabídka WINDOW (Okno)	55
5.5.7	Nabídka HELP (Nápov da).....	55
5.6	Prom nné	56
5.6.1	Deklarování prom nných	57
5.7	Operátory	59
5.8	P íkazy	59
5.9	Procedury a funkce	60
5.10	O-et ení chyb p i výpo tu	62
PRAKTICKÁ ÁST		63
6	Výukový program Prostorová t lesa	63
6.1	Systémové pofadavky.....	64
6.2	Spu-t ní a ukon ení výukového programu	64
6.3	Popis výukového programu.....	64
6.3.1	Hlavní okno	65
6.3.2	Výukový text	66

6.3.3	Test.....	68
6.3.4	Pexeso	72
6.4	Didaktický pínos a výhody programu	74
Záv r	75
Zdroje.....	77
Seznam obrázk	80
Seznam p íloh.....	82

Úvod

V dnešní době si svítání vyučovací proces bez počítače a techniky neumíme představit. Informační technologie nás obklopuje na každém kroku. Úroveň dnešní moderní doby pokročila natolik, že lze mnoho učebních pomůcek a výukových prostředků, které učitelé i žáci při výuce používají, nahradit anebo doplnit ICT¹.

Je ovšem nezbytné si uvědomit kdy, kde a pro jaký cíl je vhodné zavést a zapojit ICT do výuky. Učitel by měl správně zhodnotit míru předpokládaného efektu od šetrnosti, která se může při použití objevit (Cihlář, 2006).

Rámcový vzdělávací program pro základní vzdělávání stanovuje, že se žáci mají v hodinách matematiky učet využívat prostředky výpočetní techniky. Zahrnuje to používání kalkulačky, počítačových softwarů a různých výukových programů (msmt.cz, 2014).

Diplomovou práci nesoucí název *Využití Visual Basic ve stereometrii na ZTM* jsem si vybrala z důvodu zaměření mého druhého oboru, čímž je Informační a technická výchova. V hodinách programování jsme se naučili pracovat v jazyce Visual Basic, což mi otevřelo nové možnosti, jak vytvářet zajímavé výukové programy za účelem zpestření hodin matematiky, které jsou podle žáků nevlády zábavné.

Výukový program je dle mého názoru zajímavým a přínosným doplněním učiva. Soudím tak podle reakce žáků na praxi, kdy jsem před jejich prázdninami, hodinu matematiky v nově opakovaném učivu. Jako formu jsem zvolila didaktickou hru, kterou jsem našla na internetu. Aplikace byla vytvořena v programu PowerPoint, a tudíž mnoho možností nenabízela. Přesto byla pro žáky atraktivní a všichni se do hry zapojili. Aniž by si to tehdy uvědomili, ocitli se nenásilnou formou v procesu učení. Tenkrát jsem si uvědomila, jak lze využít VB² při tvorbě komplexního výukového programu.

Cílem této práce je vytvořit komplexní aplikaci k učivu stereometrie, která nabídne jak teoretickou část, tak i praktické ukázky. Dále naprogramovanou didaktickou hru a hodnotící test, který poskytne uživateli zpětnou vazbu. Komplexní pohled na práci doplní potřebná

¹ ICT je zkratka z anglického Information and Communication Technologies - označuje informační a komunikační technologie. (it-slovník.cz, 2014)

² VB je zkratka Visual Basic

teorie o počítačích ve výuce, o výukových programech, o základech programování a o zahrnutí stereometrie do RVP ZV³. Na závěr je potřeba program doplnit o jeho manuál.

Práce je rozdělena na teoretickou a praktickou část. Teoretickou část jsem rozdělila do pěti kapitol. První kapitola nesoucí název *Počítač ve výuce* se zabývá historií vstupu počítače do škol. V této části této kapitoly však pojednává o využití počítače ve výuce. Významná je zde podkapitola s názvem E-learning.

Druhá kapitola je pojmenovaná *Výukový software*. V této části je v novém prostoru pro vymezení pojmu výukového softwaru. Jsou zde především popsány funkce a druhy těchto programů. Závěr kapitoly je v novém výběru výukových programů a jejich zhodnocení.

Třetí kapitola *Tvorba výukového programu* popisuje, na co by se měl programátor zaměřit, než začne aplikaci vytvářet. Zde je kladen důraz na rozmyšlení nárocnosti tvořené aplikace a tedy výběr softwarového prostředí a programovacího jazyka. Dále jsou zde popsány příklady neboli rozvržení kroků při vytváření programu.

Ve čtvrté kapitole nesoucí název *Stereometrie na ZTM* jsem se zaměřila zejména na zařazení tohoto tématu do RVP ZV a jeho rozepsání do jednotlivých ročníků na druhém stupni základní školy. Jsou zde vyjmenovány i dílčí výstupy žáka, které jsou převzaty z TMWP základní školy Heyrovského v Olomouci.

Poslední kapitola teoretické části se nazývá *Visual Basic*. Tato značně obsáhlá část práce je zaměřena na jazyk Visual Basic, na jeho historii, nástupce a na objasnění základních pojmů. Ovšem dle mého názoru je popis uživatelského prostředí aplikace Microsoft Visual Studio 2010 Express, ve kterém je tvořen výukový program. Na závěr je v této kapitole zmíněna teorie potřebná k programování. Zahrnuje poznatky o proměnných, o jejich deklaraci, procedurách, funkcích a příkazech. Vše je doplněno ukázkami.

Praktická část je zaměřena na popis výukového programu, který jsem nazvala *Prostorová tělesa*. Stejně pojmenovaná kapitola detailně popisuje jednotlivá okna aplikace. Tento sepsaný text tvoří návod k jejímu spuštění a je doplněn konkrétními obrázky. Každý uživatel, který tento text nastuduje, by měl být schopen program naplno využívat. Manuál k programu je přiložen jako příloha na CD-ROM⁴.

³ RVP ZV je zkratka rámcového vzdělávacího programu pro základní vzdělávání

⁴ CD-ROM (anglická zkratka pro *Compact Disc Read-Only Memory*) je nepříepisovatelné optické záznamové médium.

TEORETICKÁ ÁST

1 Po íta ve výuce

Po íta v dnešní době plní v oblasti vzdělávání velmi důležitou roli. Je téměř nezastupitelný. Jeho zasažení do výuky již bezpochybně výrazně ovlivňuje a otevírá možnosti použití různých metod. Ovšem neplatí tvrzení:

$$\text{moderní ICT} + \text{výuka} = \text{efektivnost!}$$

Uitelé si musí uvědomit, že k dosažení efektivnosti je třeba uplatit správných metod. Z toho vyplývá rovnice:

$$\text{moderní ICT} + \text{výuka} + \text{správné metodické využití} = \text{efektivnost.}$$

(Dostál, 2007)

Na zavádění počítače do výuky se výrazně podílí především Ministerstvo školství, mládeže a tělovýchovy, které tuto problematiku shrnuje v dokumentu *Koncepce státní informační politiky ve vzdělávání*⁵ (Klement, 2002).

1.1 Vstup počítače do škol

Vše odstartoval vývoj mikropočítače, který se datuje na konec sedmdesátých let. Jejich nástup přinesl do škol zvrát. V té době se kolem počítače pohybovala jen malá skupinka učitelů základních a středních škol, kteří využívali především jako výpočetní technika nebo programování. Na základních školách se pomalu začali organizovat různé zájmové kroužky zahrnující práci na počítači (Černošková, 1998).

Bylo zřejmé, že je nutné zavést pojem informační gramotnost, kterou Kropáček (2004) definuje jako: schopnost člověka využívat moderní informační technologie a prostředky v běžném životě. Moderními informačními technologiemi a prostředky jsou myšleny počítače, počítačové sítě a mobilní telekomunikační prostředky.

Pozvolna se definovala nová tvář školy. Avšak se zdálo, že bude nutností, aby se každé dítě muselo naučit základy programování. Vystaly otázky, v čem učitelé mají programovat.

⁵ Dokument přístupný na adrese: <http://www.fi.muni.cz/~smid/sipvez1.html>

Boulivé diskuze se vedly kolem programu Pascal, Logo, Basic. Vě utichlo nástupem osobních počítačů a jejich příjmem programového vybavení (Kropáček, 2004).

1.2 Využití počítače ve výuce

Po počítači můžeme do výuky zahrnout dvě zapsobí. Oba zapsobí se prolínají, a proto je nejde od sebe jednoznačně oddělit. Jedná se o:

1. Výuka o počítači

Zde jsou zahrnuty všechny poznatky o technickém vybavení (hardware), o programovém vybavení (software), obsluze a údržbě. Rovněž můžeme zařadit výuku o využití aplikacíního software.

Říká se v tomto případě objektem, který se má vzdělávat v oblasti počítače. Tento zapsobí je nezbytným a významným předpokladem pro výuku s počítačem. Toto navázání je nesmírně důležité, protože bez něj by výuka postrádala smysl.

2. Výuka s počítači

Zde jsou obsaženy všechny zapsobí využití počítače pro učební výuky. Po počítač je zde poměrně jednak u učitele a jednak u žáka. Takovéto pojetí výuky může být uplatněno v téměř všech případech tech. Po počítač se stává prostředkem výuky, a jistě ho nemůžeme zařadit pod pojem objekt.

Je nutná jistá míra znalosti práce s počítačem, ovšem u učitele ani u žáka není nutná znalost programovacích jazyků.

(Klement, 2002)

Výuku s počítači Burianová (2003) v publikaci *Využití aplikací programů ve výuce* dále dělí na:

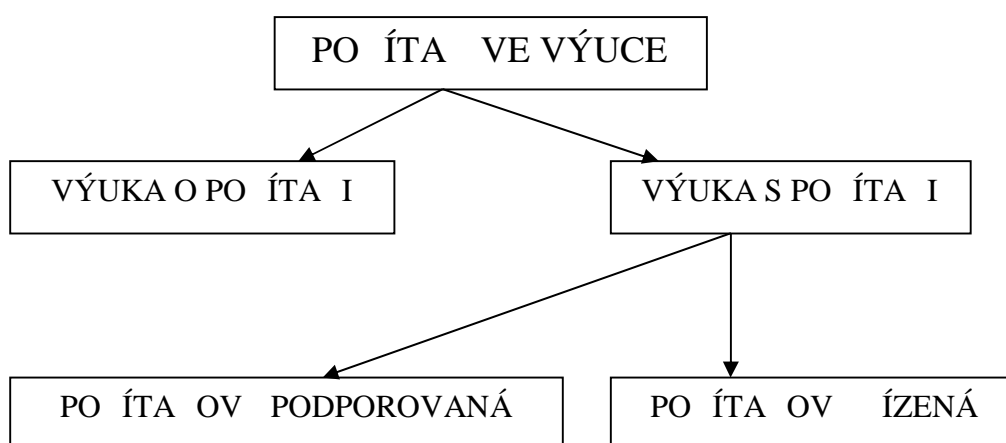
a) počítačově podporovaná výuka (ComputerAidedLearning)

Počítač je zde používán jako doplňující prostředek u učitele, který má plnit pouze dílčí didaktické funkce. Zastává funkci nástroje, který je používán k procvičování látky, výkladu, simulací i k testování.

b) počítačem řízená výuka (ComputerManagedLearning)

Na rozdíl od předchozího odstavce, má zde počítač roli systému zajišťující v rámci didaktických funkcí. Počítač zadává úkoly, nabízí zásobu lekcí a testovacích úloh. Sám úlohy vyhodnotí a zadá je do evidence studijních výsledků. Kdybychom to chtěli shrnout jedním slovem, mohli bychom tento typ výuky nazývat e-Learning, kde jsou učitelé v pozici tzv. konzultantů.

Pro názorné rozdělení slouží níže uvedené schéma.



Obrázek 1: Využití počítače ve vzdělávání

1.3 Funkce počítače ve výuce

Z výše uvedených způsobů aplikací je zejména, má počítač ve výuce mnoho různých významných funkcí. Mezi nejdůležitější patří:

1. počítač jako učební pomůcka

Jedná se o nejužívanější funkci, ve které používáme počítač jako pomůcku při výuce programování, obsluhy počítače, poznávání jednotlivých typů počítačů atd. Počítač jako učební pomůcka zvyšuje názornost pomocí různých simulací, grafiky a modelování. Dále může sloužit k prezentaci učební látky a k zpřístupnění informací.

2. počítač jako didaktický prostředek

Počítač je využíván při výuce s didaktickými programy, při spojení s dataprojektorem nebo s interaktivní tabulí. V dnešním kolektivu je výuka realizovaná v běžných třídách, a kromě třídy pro hodiny informatiky není ve většině případů umožněna samostatná práce. Proto pracuje s počítačem pouze vyučující. V této funkci počítač plně nahradil starou pomůcku jako je například projektor, CD přehrávač, videorekordéry, televizor atd.

3. počítač jako pracovní nástroj

Slouží zejména učiteli při přípravě a plánování pedagogického procesu, při hodnocení a řízení výuky. Pro vedení pedagogické administrativy je na dnešním trhu nejvíce využíván program *Bakaláři*, který eviduje jednak zaměstnance a žáky, ale navíc umožňuje vést tzv. elektronickou třídní knihu, klasifikaci žáků, rozvrhy hodin, tematické plány apod.

4. počítač jako velmi aktivní pomocník učitele

Dochází k posílení práce učitele s informacemi. Učitel může řízení prezentované látky vhodně využít informace o učivu a o míře pochopení žáka. Dochází pak k podrobnější analýze vzdělávací práce učitele a následně možnost zkvalitnit jeho působení.

(Dostál, 2007)

1.4 Negativní důsledky nasazení ICT do výuky

Věchno má svoje klady a zápory, takže není překvapením, že nasazení ICT do výuky má i svoje negativa. Mezi ně patří:

- nároky na údržbu a přípravu ICT před hodinou
- narušení výukového procesu při výpadku ICT
- zvýšení nároků kapacity školy
- snižování jazykové úrovně
- zvyšování sociálního úniku
- snaha uplatnit počítače tam, kde nemá učitel dostatečnou počítačovou gramotnost
- snaha nahradit počítačem učitele
- internet se může stát zdrojem odvádění pozornosti od výuky i místem nevhodných materiálů

(Dostál, 2007)

1.5 E-learning

S pojem E-learning se dnešní společnost začíná setkávat poměrně často. Na tuto problematiku byla již napsána řada publikací, které ovšem vymezují tento pojem různě. Je to především zapříčineno dynamickým vývojem moderních technologií, které vstupují do procesu učení.

Pro jeho správné pochopení je namístě si termín rozložit do dvou částí. Še, které považujeme za zkratku znamenající šelektronické a zahrnující problematiku zaměřenou na využití prostředků a nástrojů ICT. Dalším slovem je e-learning, které překládáme jako učení, což je jeden z klíčových procesů v životě člověka. Po spojení těchto dvou slovy můžeme zformulovat pojem E-learning jako učení, které je moderními technologiemi umožňováno a podporováno. Ovšem ani tato definice není dostatečně přesná.

(Zounek, 2009)

Zounek (2009) dále uvádí, že pojem nelze vymezit pouze na popis možností ICT ve vzdělávání. Za příklad uvádí první generaci e-learningu, kde řada odborníků zastávala názor: odstranit učitele z procesu učení a nechat žáky pracovat svým vlastním tempem.

Autor uvádí tuto filozofii jako šprohadák a dodává že e-learning není počítačový systém, který funguje po zakoupení a nainstalování sám od sebe. Zmíní uje, že jde o mnohem rozsáhlejší problém, který zahrnuje komplexní systém, založený na spolupráci lidí, počítačových sítí a počítačových sítí.

Jak již bylo výše zmíněno, existuje řada různých způsobů vymezení pojmu e-learning. Na které definice kladou důraz na dostupnost výukového softwaru pomocí moderních technologií, dále říká, že stejně jako prvkem je komunikace. Jsou i takové definice, které spíše zdůrazňují místo, kde probíhá proces učení (Zounek, 2009).

Nocar (2004) ve své práci popisuje e-learning jako "šmultimediální podporu vzdělávacího procesu využívající moderní informační a komunikační technologie pro zkvalitnění vzdělávání".

Pedagogický slovník (2003) uvádí: "E-learning označuje různé druhy učení podporovaného počítačem, zpravidla s využitím moderních technologických prostředků, především CD-ROM." Na této definici je zřejmé, že vymezení tohoto pojmu je opravdu náročné a to proto, jak už bylo výše zmíněno, že se spolehneme, a tím i tato problematika dynamicky vyvíjí. V tomto konkrétním případě zde bylo opomenuto zahrnutí síťových prostředků, které jsou jednou z důležitých znaků moderních technologií.

Na výše uvedených pojetích významu e-learningu Zounek (2009) postavil jeho komplexní vymezení: "E-learning zahrnuje jak teorii a výzkum, tak i jakýkoliv reálný vzdělávací proces (s různým stupněm intencionality⁶), v němž jsou v souladu s etickými principy používány informační a komunikační technologie pracující s daty v elektronické podobě. Způsob využívání prostředků ICT a dostupnost učebních materiálů jsou závislé především na vzdělávacích cílech a obsahu, charakteru vzdělávacího prostředí, potřebách a možnostech všech aktérů vzdělávacího procesu."

V oblasti školství Dostál (2007) konkretizuje definici na šformu vzdělávací činnosti, při níž vzdělávající a vzdělávání vstupují do určitých vztahů za aktivní účasti počítače jako technického prostředku pro dosažení stanoveného cíle.

⁶zaměstnanost v domě na světě jako podmínka myšlení (slovník-cizích-slov.abz.cz, 2014)

1.6 Konkrétní využití počítače ve výuce

Toto využití shrnuje (Dostál, 2007) do několika bodů :

1. multimediální programy
2. simulační programy, modelování
3. testovací programy
4. výukové programy
5. informační zdroje
6. videokonference
7. distanční formy výuky
8. virtuální realita

2 Výukový software

Je zřejmé, že počítač má při výuce zjevné místo. Rozvojem ICT technologií, a tím i rozvojem programovacích programů a uživatelského prostředí, vznikají možnosti jak zlepšit výuku. Začíná se pomalu rozmáhat využívání výukových programů.

Záměr využívat výukový aplikační software při vyučovacím procesu vychází z metody programového vyučování. Jedná se o tzv. šautodidaktický proces, při kterém se žák vzdělává podle svých schopností, tzn. tempo a náročnost práce si volí sám. Program má řídit funkci, tzn. žákově sdělit nové informace, ale zároveň kontrolovat jeho práci. Neméně důležité je, že pedagogovi poskytuje zpětné informace a vyhodnocuje žákovu práci (Klement, 2002).

Je třeba si ovšem uvědomit, že cílem není nahrazení učitele právě výukovým programem. Plně vzdělaný pedagog je samozřejmě nenahraditelný. Ale i přesto mu výukový program může v hodinách pomoci s expozicí učiva, procvičováním, při testování apod. Při takovéto kombinaci dochází k zefektivnění a optimalizaci vzdělávacího procesu (Dostál, 2009).

2.1 Historie výukových programů

Historie výukových programů se datuje do roku 1954 a je spojována s prof. B. F. Skinnerem.⁷ V té době vznikaly první programové učebnice a bouřlivý rozvoj způsobily osmibitové mikropočítače, které následně vystřídaly IBM kompatibilní PC⁸.

První aplikace vznikly na konci 60. let. Jednalo se o programy určené pro vyučovací automaty založené na principu programového učení podle Skinnera. Byl vynalezen automat zvaný Unitutor, který patřil u nás k těm nejlepším. Na každé straně programu byl text sloužící k výkladu a na jeho konci byla kontrolní otázka s výběrem odpovědi. Program tak poskytoval okamžitou zpětnou vazbu.

(Burianová, 2003, Matematický a výukový software)

⁷Skinner je autorem *programového učení*, které vyvinul na základě operantního podmínění. Programové učení je založeno na zpětné vazbě, kontrole učení, která probíhá bez učitele a je přenesena na žáka, dále na zásad postupné aproximace, tj. postupného směřování ke konečnému cíli. (etex.czu.cz, 2003)

⁸ IBM kompatibilní PC je v informatice označení pro počítače, které jsou do určité míry kompatibilní s původním IBM PC

2.2 Vymezení pojmu výukový software

Nefl se budeme v novat výukovým program m, je t eba si ujasnit tendenci nep esného uflívání tohoto pojmu. B flní uflivatelé si pod pojmem výukový software p edstaví nap . MS Word, který se ale za n j považovat nedá. Podobn bychom mohli uvařovat o r zných –kolních informa ních systémech. Ty jsou sice pro –kolu vytvá eny, ale nejsou spojeny s výukou a tudířl neplní didaktické funkce.

Podle Dostála (2009) je šVýukový software jakékoliv programové vybavení po íta e, které je ur eno k výukovým ú el m a dokářle plnit alespo n kterou z didaktických funkcí.õ

Slavík (1997) uvádí definici: šVýukovými programy rozumíme software, který p edkládá řákovi celek u iva a zaji– uje osvojení jeho obsahu řákem. Zaji–t ní osvojení řákem rozumíme zprost edkování zp tné vazby a výstup pro dal–í pot eby.õ

ernochová (1998) považuje termín výukový program za ekvivalent termínu didaktický program. Podle ní je to tedy program, který n co vyu uje.

Mazák (1988) ve svém díle zastává názor, ře švýukový program je kařdý programový produkt, který lze za adit do struktury didaktiky ur ítého p edm tu. Podle charakteru didaktické funkce programu m řeme ur ít jeho vyuřlití ve výuce.õ

Burianová (2003, Matematický a výukový software) uvádí: šVzd lávací software je program, jehořl struktura a obsah vedou uflivatele k osvojení nebo procvi ení si nových znalostí z ur íté oblasti, p i emřl p i tom vyuřřívá moderních didaktických metod.õ

V anglické literatu e se m řeme setkat s pojmem eduka ní software (Educational software), který je definován jako jakékoliv programové vybavení po íta e, které je p edur eno pro vyuřřití v situacích, kdy dochází k rozvoji osobnosti jedince (Dostál, 2009).

2.3 Principy innosti výukových program

Klement (2002) ve své práci uvádí základní principy innosti výukových program :

1. princip aktivní odpov di ó pofladuje se od řáka zjevná, nikoliv pouze my–řená odpov
2. princip zpevn ní ó za azovat do procesu u ení informace o jeho výsledku dostate n etn , bezprost edn a nep etřřit . Znalost správného výsledku je zpevn ní

3. princip malých kroků u učivo musí být rozděleno na velmi malé kroky, aby se princip zpevnění mohl projevit
4. princip samostatného tempa učení je efektivní jen tehdy, je-li přizpůsobeno tempu žáka
5. princip řízení učitel by měl co nejméně zasahovat do práce žáka s výukovým programem, program řídí celou práci žáka. Pokud toto nečiní, není výukovým programem.

2.4 Funkce výukového softwaru

V dnešní době existují různé výukové programy, které plní nejrozličnější funkce. Většinou jsou tvořeny pro výklady, cvičení nebo test. Existují dokonce i takové, které plní více funkcí najednou. Mezi základní funkce výukového programu řadíme:

2.4.1 Výklad učiva

Základní funkcí je zprostředkovat žákovi teoretické poznatky. Takto funkce byla v historii zcela první. Počítáme-li za úkol reprodukovat písemný záznam učitele. Je přirozené, že tento způsob výuky nepřinesl nic nového, pouze nahradil učebnice.

Dnešní doba natolik pokročila, že jsou programy založené na hypertextu, často ve formě WWW stránek. Znamnou výhodou této funkce je, že informace nejsou studentovi předávány jedna za druhou, naopak jsou propojeny pomocí odkazů. Jedná se o tzv. nelineární prezentaci učiva.

(Burianová, 2003, Využití aplikací programů ve výuce)

2.4.2 Simulace problémů, simulace hry.

Při určité problematice je mnohem efektivnější vysvětlit učivo pomocí simulace. Tak většinou ztvárňuje problém v grafické podobě. Správný program dokáže změnit parametry a tím názorně předkládat praktickou ukázkou vlivu těchto změn na výsledek. Žákovi je učivo

p edlofeno aktivn , a tudífl vede k efektivit vyu ovacího procesu (Burianová, 2003, Vyuffití aplika ních program ve výuce).

2.4.3 Procvi ování

Nej etn j-í jsou programy, které jsou vytvo eny k procvi ování látky. Aplikace p ebírá plnou kontrolu nad inností fláka a tím realizuje innost u itele. fláci hledají odpov di na otázky, které jim po íta p edloflil a své odpov di do n j vkládají. Program je sestaven tak, aby rozvíjel flákovi dovednosti, nau il ho pouffvat poznatky, procvi il p edlofenný text, a co nejvíce ho pomohl flákovi upevnit (Burianová, 2003, Vyuffití aplika ních program ve výuce).

2.4.4 Testování

asto se objevují programy, které plní testovací a zkou-ecí funkci nebo které ov ují osvojení u iva. Je ov-em nutné poznamenat, fle by takovýto program m l okamffit poskytnout flákovi zp tnou vazbu, tzn. docílit toho, aby se flák nenásilným zp sobem zároveň i u il. N který program, je nastaven tak, aby podle zadaných kritérií navrhl u iteli známku. Tím je pedagogovi uleh ena práce a je docílena spravedlnost p i hodnocení test více flák .

Testovací programy m fleme d lit podle zp sob odpov dí. N které programy nabízejí otázky s výb rovou odpov dí. Sloflit j-í zvládnou zhodnotit odpov tvo enou, tzn. takovou, kterou flák sám sestaví. Dále se nám vyskytují otázky p i azovací, uspo ádávající nebo umis ovací.

(Burianová, 2003, Vyuffití aplika ních program ve výuce)

2.5 Programové u ení

Burianová (2003, Matematický a výukový software) rozd luje programové u ení na kroky, neboli na malé ásti, které jsou se azeny v organickém sledu. Toto se azení zaji-uje, fle fláci postupn zvládnou obtífln j-í dovednosti a v domosti. Každá ást je zakon ena kontrolou, která objasní, zda flák danou problematiku zvládl i nikoliv. Podle návaznosti sled jednotlivých flákových krok rozeznáváme:

- a) Lineární programy
- b) Vtvené programy

2.5.1 Lineární výukové programy

Jsou spojeny s jifl zmi ovaným anglickým psychologem Skinnerem. Zastával názor, fe princip u ení vychází ze zpevoání a opakování. Jedná se o postup, ve kterém flákovi sd luje teorii a následn provádíme kontrolu, zda si u ivo doopravdy osvojil. K tomu volíme vhodné úkoly a otázky.

Na stejném principu jsou postaveny lineární výukové programy. V p ípad , fe flák odpoví chybn , program ho sice vede obsahem dál, ale s tím, fe je pot ebné u ivo v dal-í ásti n kolikrát znovu uvedeno a vysv tleno. Z toho vychází pojmenování, nebo flák je veden stále dál a dál. Program se nev tví, je lineární.

Tento druh programu je vhodný hlavn pro men-í výukové celky.

(Klement, 2002)

2.5.2 Vtvené výukové programy

Jsou spojeny s osobností pana Crowdera, jenfl se zabýval vzd láváním dosp lých. Zastával názor, fe chyba není negativní skute nost, ale fe je naopak jakýmsi pozitivním stimulem pro dal-í získání pot ebných poznatk .

Program stojí na principu chyby, jako tzv. prvku pro upevn ní a osvojení u iva. V programu se nacházejí kontrolní body, které mají funkci šk iflovatky. flákovi je p edlofena a vysv tlena teorie, následn ho program zavede do testovací ásti. V p ípad , fe flák tuto ást neabsolvuje, program ho dál nepustí. Uflivatel je programem vrácen zp t na za átek nebo se v jiné ásti programu pomocí teorie znovu vzd lává. Posléze je op t p iveden na kontrolní bod, ve kterém prokazuje osvojení znalostí. Celý proces se opakuje tak dlouho, dokud flák neusp je. Program by m l být situován tak, fe ím víckrát uflivatel opakuje teoretickou ást, tím podrobn ji by mu b la být p edkládána.

Tento druh programu je vhodný pro větší tematické celky, z čehož vyplývá doporučení spojit do výukového programu pro zpestření i nejrozličnější didaktické hry.

V tvorbě výukové programy jsou velmi náročné na konstrukci, a to jak v oblasti obsahové, tak i v oblasti strukturální. Je třeba dbát na to, aby byl účastník pravidelně informován o průběhu a postupu výuky.

(Klement, 2002)

2.6 Výhody a nevýhody výukových programů

Podle Dostála (2007) přináší výukové programy nespočet výhod. Bohužel je zřejmé, že se objevuje i řada nevýhod. Mezi pozitivní řadíme:

Pozitiva:

- rozvíjení kreativity studentů
- zábavnější forma učiva, u které si žáci často neuvědomují, že se vlastně učí
- žáci jsou více motivováni do práce, navíc se učí pracovat s výpočetní technikou
- dochází k lepšímu a jednoduššímu zapamatování díky názorné ukázce realizované pomocí počítače, kterou žáci vnímají zrakem a zároveň spojenou s výkladem vnímaným pomocí sluchu
- studenti si sami volí čas studia
- výuka je interaktivní, student se hned dozví, zda úkol splnil či ne
- forma a rychlost výuky se dá lépe přizpůsobit schopnostem i rytmu studenta
- výukový program nahrazuje řadu různých pomůcek, bývá zpravidla názornější než výuka

Negativa:

- zdlouhavá práce na počítači ohrožuje zdraví, dochází k jednostrannému pohybovému zatížení systému, zhoršení zraku, zdravotním problémům se zády, šíjí
- studenti spolu nekomunikují, zaostávají v sociálních dovednostech, chybí jim kontakt s vrstevníky, dochází značně ke snížení komunikačních a vyjadřovacích schopností

- v nich kterých případech se může student stát závislý na počítačích, ovlivuje to způsob trávení volného času mimo školu
- studenti ztrácejí motivaci číst, neumí jí vyhledávat literární zdroje, většinou jim obstará internet

2.7 Kategorizace výukových programů

Učitelé výukových programů se zabývali různými autory. Můžeme zde uvést například Burianovou Evu a její publikaci s názvem Matematický a výukový software (2003) nebo také knihu Počítačové výukové programy od pana Mazáka (1988). Ovšem dále budeme zmiňovat kategorizaci, kterou uvádí Dostál (2009) a která je výsledkem studia 148 výukových programů od dodavatelů jak domácích, tak zahraničních.

a) dle míry interaktivity:

Interaktivita je podstatná vlastnost, která spočívá v aktivním ovlivňování obsahu programu nejen jedincem. Uživatel tedy nepřijímá pouze pasivně vzdělávací obsah, ale může číst, může komunikovat s programem. Student je více motivován a učení se stává efektivní. Je zřejmé, že se vyskytují programy s různou hladinou interaktivity.

- s interaktivními prvky
- bez interaktivních prvků

b) dle úrovně vzdělávání:

Je celkem jasné, že když máme různé typy RVP pro různé stupně vzdělávání, máme stejné typy výukových programů. Pro každý stupeň je typický rozdílný didakticky transformovaný obsah. Přede vším jsou zohledněny výchovně-vzdělávací cíle, které jsou brány s ohledem na věkové rozdíly žáků. Je zřejmé, že i nejdokonalější výukový program v rukou žáka, který nemá dostatečnou úroveň psychického vývoje, nedokáže dobře plnit svoji funkci.

Výukové programy dělíme:

- pro mateřské školy
- pro základní školy

- pro střední školy
- pro vysoké školy

c) dle míry poskytování zpětné vazby

Student má právo z psychologického hlediska dostávat zpětnou vazbu. Výukový program ji zajišťuje v části cvičení, úkol nebo v oblasti testování. Důležitou roli hraje zpětná vazba především v oblasti korigování nesprávně osvojených poznatků. Bohužel sada výukových programů zpětnou vazbu nemá, proto je možné dle:

- se zpětnou vazbou
- bez zpětné vazby

d) dle organizovanosti vzdělávání

Znáhou výhodou výukových programů je, že nejsou omezeni pouze na použití v prostředí výuky, ale umožňují i samostudium a to jednak v rámci doplnění potřeby v domácnosti, ale také v oblasti zájmu a koníků. Podle toho, jestli student jde nebo ne jde sám sebe, dle výukové programy na:

- pro školní výuku
- pro samostudium

e) dle on-line x off-line funkcí

Výukový program realizovaný ve výuce je ve většině případů nainstalován přímo na počítačích, anebo na serveru školy. Vedle toho existují i programy, které jsou nainstalované na vzdáleném serveru a jsou poté on-line sdíleny pomocí internetu. Poslední možností je případ, kdy je výukový program nainstalován přímo na počítači, ale podporu lze získat pomocí internetu. Shrnutí výukové programy dle:

- off-line
- off-line s on-line podporou
- on-line

f) dle počtu uživatelů

Podle toho, kolik uživatelů sdílí ve stejném případě jeden výukový program, dělíme na:

- monouživatelský
- víceuživatelský

g) dle tematického rozsahu:

Podle toho, jestli výukové programy obsahují jeden nebo více tematických celků, dělíme na:

- monotematický
- polytematický

h) dle možností vnímání:

Dále dělíme výukový software dle toho, čím působí na uživatele. Jedná se o možnost působení pomocí vizuálních vjemů, anebo v kombinaci s vjemy auditivními. Je zřejmé, že pokud zapojíme do vnímání sluch, je program zajímavější, více motivující a tudíž lépe dokáže plnit svou funkci.

- vizuální
- audiovizuální

ch) dle jazykových mutací:

Většina výukových programů je psaná v mateřském jazyce uživatele, ovšem můžeme se setkat i s programy, které nabízejí současně mateřské i cizojazyčné rozhraní. Dělení:

- jednojazyčný
- vícejazyčný

i) podle verze:

Jistě jste se již setkali s pojmem demo verze. Většinou jde o programy různých antivirů, program pro úpravu fotek, počítačové hry apod. Stejně můžeme rozdělit i výukové programy. Prodejci, nebo můžeme napsat výrobci výukových programů, nabízejí možnost tzv. zkušební verze programu (neboli demoverze), která je sice funkční, ale na které její aplikace nejsou povoleny. Znamnou výhodou ovšem je to, že si program můžeme otestovat a zjistit, zda program odpovídá našim požadavkům. Opakem demo verze je plná verze, která je kompletní, tzn. bez omezení. Dělí:

- plná verze
- demo verze (omezená)

j) dle počtu didaktických funkcí

Abychom mohli program označit jako výukový, musí plnit nejméně jednu didaktickou funkci. Na trhu jsou často k dispozici programy s jednou didaktickou funkcí, tj. pouze například výkladové nebo testovací. Naproti tomu ovšem existují programy, které zastávají více funkcí souasně. Dělí:

- s jednou didaktickou funkcí (motivační, expoziční, fixační, verifikační)
- s více didaktickými funkcemi

k) dle zaměření na jednotlivé předměty:

Podle toho, zda programy prezentují určitý obsah jednoho předmětu, nebo jsou naopak zaměřeny na obsah mimo rámec vyučování, lze výukové programy dělit na:

- s předmětovým zaměřením (matematika, fyzika, dějepis, zeměpis, chemie, í)
- bez předmětového zaměření

(srov. Dostál, 2009)

2.8 Kritéria hodnocení výukových program

Každým pedagog, který p i své výuce bude využívat výukový program, stojí p ed nelehkým úkolem. Vybrat vhodnou aplikaci. V první ad by se m l u ítel zam ít na následující kritéria, které byl m l kvalitní výukový program splnit. Burianová (2003, Využití aplika ních program ve výuce) je sepisuje v následujících bodech:

1. Kvalitn provedená dokumentace

Nezáleží na tom, jestli u ítel program vytvo í nebo jen zakoupí. Ve v-ech p ípadech je nejd ležit j-í ástí manuál programu, který detailn popisuje, jak v programu pracovat. Vytvo ený výukový program samotným u ítelem m fle odpovídat nárok m jiných pedagog , kte í program neznají, ale p esto by ho cht li ve své výuce pouít. Proto by se na vytvo ení manuálu nem lo zapomínat.

D raz by m l být kladen na popis typických problém a chyb, které mohou nastat a kterých se práv m fle dopustit užívatel, který program je-t natolik nezná. P ípadn je výhodné, za lenit do programu text s adresou a kontaktem, na který se m fle užívatel v p ípad problém obrátit.

Taktéí užíte né je p íložit k programu tzv. metodický návod, který popisuje a radí jak program za lenit do výuky.

2. Užívatelsky p ív tivá instalace

Náro n j-í program, by m l obsahovat vlastní instalaci. Instalací program zajistí jednak dekomprimaci dat a jednak jejich následné uložení na konkrétní místo, které si sami zvolíme. Správn vytvo ený program by m l nabídnout užívateli i jeho odinstalaci, která odstraní v-echny soubory programu i zm ny v registru, které vznikly v opera ním systému p í instalaci.

Oba programy, jak instala ní tak i deinstala ní, by m li být užívatelsky jednoduché a nem ly by ínit velké problémy.

3. Jednoduché a pirozené ovládání

Výukové programy v t-ínou vyuffívají vedle uitel i fláci a rodi e, kte í nemusí mít velké znalosti a zku-enosti s prací na počíta i. Proto by m l uitel p i výb ru výukového programu zohlednit ovládání programu, práv pro cílovou skupinu uflivatel . Snaha p íjít na to, jak s programem pracovat, kde zjistit informace o programu a jaké funkce program nabízí, vede k neefektivit vzd lávacího procesu a program tudífl neplní funkce, pro které byl vytvo en.

4. Možnost zp tné vazby

P ítomnost zp tné vazby ve výukovém programu je nemén d leflitá. Kvalitní výukový program spl ující p edání u iva flákovi a kontrolu získané úrovn znalostí by m l vyhodnotit výsledek, a následn p edat informaci zp t uflivatel i. Výhodou z stává, fl e program p istupuje k flákovi individuáln , cofl ve v t-in p ípad není v moci u itele. Aplikace reaguje na flákovu chybu p edlofením dopl ujících informací, p ípadn vrácením na osvojovaný text a možnost znovu otestovat probíranou ást. Tento zp sob osvojování informací zaru í, fl e kaflký flák bude sice pracovat jiným tempem, ale na záv r programu se dostane i pr m rný flák.

5. Podpora multimediální technologie

Stále více jsou fládané programy, které nabízejí videosekvence, audiosekvence, animace, obrázky a které podporují b flné multimediální formáty.

6. ízení pr chodu programem na základn reakce uflivatele

Program lze hodnotit i z pohledu míry vlastního tempa fláka, nastavení asového limitu p í e-ní úloh nebo taky p i možnosti nastavení po tu kontrolních otázek i schopnosti programu výsledky archivovat.

7. Další kritéria jsou
- ovládní v mate ském jazyce
 - rozumná cena
 - pravidelná aktualizace programu výrobcem
 - dob e zpracovaná nápo v da

2.9 Výb r výukového programu

Po o-et ení vý-e uvedených kritériích je d leffité se p i výb ru výukového programu zam ít p edev-ím na následující body, a jim pod ídit rozhodování o výb ru programu.

- a) výukové cíle ó výukovým cíl m musí být pod ízeny v-echny prost edky pomáhající jejich dosažení, tudífl i výukový program
- b) v k a úrove psychického vývoje flák - je z ejmé, fle jinak budeme vybírat program pro fláky prvního stupn a jinak pro fláky vy-ích stup . Tzn., jiný obsah bude mít program na stejné téma, ale v r zných stupních eduka ního procesu.
- c) schopnost u itele integrovat programy do výuky ó kaflký z kantor má rozdílnou schopnost zasazení programu do výuky. Informa ní gramotnost by m la být sice jakousi profesní výbavou u itele, ov-em jejich míra se m fle zna n li-ít. Proto by u itel m l využívat takové výukové programy, které svou náro ností na obsluhu a obsah odpovídají jeho schopnostem.
- d) podmínky realizace ó nedílnou sou ástí p i rozhodování o pouffítí výukového programu bývají realiza ní podmínky, tzn. vybavenost u eben, dostupnost program a hardwarové vybavení po íta e

(Dostál, 2009)

3 Tvorba výukového programu

V dnešní době jifi sice na trhu existuje spousta výukových programů, ale z vlastní zkušenosti vím, že ne všechny se uitelům pro použití ve výuce hodí. Každý uitel, i když stejného předmětu a ročníku k němu postupuje jinak. Volí jiné metody a pomůcky. Rozdílný je i přístup k důležitosti osvojení konkrétních znalostí. Jeden uitel v hodinách matematiky klade důraz na zapamatování například vzorek, druhý třeba dává větší důraz na poítání z paměti. Naopak pro některé může být důležitější logické uvažování a aplikace poznatků. Bude zastávat názor, že vzorce najdou žáci v tabulkách a pro poítání mohou používat kalkulátory.

Pokud chceme do vyučovacího procesu zařadit i výukové programy, budeme mít dvě možnosti. Jedna představuje situaci, kdy budeme mít šanci a pro nás vyhovující výukový program získáme. Buď na něj jednoduše narazíme, anebo na něj získáme finanční prostředky. Druhá možnost, kterou budeme v textu dále rozebírat je, že jsme natolik zvládání počítače a jsme dostatečně schopni využívat výpočetní techniku, že si výukový program vytvoříme podle svých představ a požadavků výuky.

V této kapitole se především do tématu, jak postupovat k tvorbě výukového programu a hlavně zde bude zmíněno, proč je výhodné použít při tvorbě výukového programu Visual Basic, kterému bude samostatně věnována kapitola 5.

3.1 Softwarové prostředí

Tvorba výukových programů se odvíjí od toho, jak je chceme udělat náročnější, tj. od možnosti komplexní aplikace do vyučovacího procesu.

Prostředí dělíme na:

1. Programovací jazyky

Toto prostředí není nijak omezeno, proto umožní vytvořit komplexní aplikace. Vytvořená aplikace obsahuje jen ty prvky, které v nich chceme mít. Nevýhodou je jednoznačně nutnost teoretických znalostí a zkušeností při tvorbě aplikací v konkrétním jazyce.

2. Vývojová prostředí

Oproti programovacím jazykům je toto prostředí znatelně jednodušší. Nejsou do takové míry vyžadovány teoretické znalosti. Lze velmi snadno vytvářet jednoduché výukové programy. Znanou nevýhodou je však omezení tvorby programu. Prostředí umožňuje pouze skládání již předem definovaných částí k sobě, bez možnosti vytvářet nové části.

(Klement, 2002)

3.2 Koncepce výuky programovacích jazyků

Pro běžného uživatele je potřeba znalosti aspoň jednoho programovacího jazyku stále aktuální. Proto není divu, že se za jiná rozmáhá snaha vychovávat uživatele, kteří budou schopni použít získané znalosti a dovednosti jak z oblasti ICT technologií, tak z oblasti pedagogiky a didaktiky především k vytvoření vlastních aplikací.

Na našich pedagogických fakultách se vychovávají uživatelé především v rámci jejich aprobace, nicméně potřeba znalosti výpočetní techniky je stále významnější a žádanější. Nebylo by tedy od věci, zařadit do náplně studia na vysokých pedagogických školách výuku programování. Ta by seznamovala studenty s tím kterým programovacím jazykem, který by jim umožnil vytvořit efektivní výukový program.

(Klement, 2002)

3.3 Volba programovacího jazyku

Nejrozšířenější objektový jazyk (vysvětleno v kapitole Visual Basic), který je zároveň vhodný a zdaleka ne tak náročný jako jsou ostatní jazyky je Visual Basic. Tento jazyk umožňuje funkční aplikace, které bez problému poběží na počítači s operačním systémem Windows od firmy Microsoft.

Zvolení tohoto jazyka má řadu výhod:

- osvojení práce s Visual Basic dává uživateli výhodu, že po krátkém seznámení, může začít pracovat v jiném objektově orientovaném programovacím jazyce

- Visual Basic nabízí programovací rozhraní zvané forApplications (VBA), ve kterém se dají vytvářet různé makra a jiné doplňky
- znalost práce ve Visual Basic dává uživateli větší prostor vytvářet WWW stránky v HTML pomocí Visual Basic Skript

(Klement, 2002)

3.4 Přístupy k tvorbě výukových programů

Přístupy, o kterých se budeme zmínovat, navrhl autor, který je pedagogem na Univerzitě Palackého v Olomouci, v rámci výuky na Katedře technické a informační výchovy.

3.4.1 Jednou úroveň výukový program

Jedná se o programy, které jsou tvořeny za účelem demonstrace vřady pevně daného výukového celku. Tím je myšleno, že po sestavení programu, již do něj není možný další zásah, například doplnění informací. Úroveň domstí programátora nemusí být v tomto případě natolik rozsáhlá. Náročná není ani příprava.

Při tvorbě tohoto druhu programu jsou důležité jisté aspekty, které Klement (2002) uvádí v jednotlivých krocích:

1. návrh uživatelského rozhraní
2. návrh a posouzení základních nabídek a formulářů
3. propojení jednotlivých formulářů a procedur na úrovni API funkcí
4. tvorba systémové logiky programu
5. příprava souborů pro potřeby distribuce programu

Ad1. Návrh uživatelského rozhraní

Je třeba, aby si programátor uvědomil, jací uživatelé budou program používat. Konkrétně se teď bavíme o třídách a uživatelských základech, proto by programátor měl vytvořit aplikaci s podobným uživatelským rozhraním, na kterou jsou zvyklí. Jak již bylo zmíněno, je výhodné použít programovací jazyk Visual Basic, který je tvořen s grafickým

rozhraním API systému Windows. Je tedy vhodné, navrhovat programy s grafickým a uživatelským rozhraním operačního systému Microsoft Windows od verze 95 a novější.

Ad 2. Návrh základních formulářů

V návaznosti na předchozí odstavec je výhodné vytvořit formuláře, se kterými jistě mají větší zkušenost. Tím je myšleno, že je formulář pro uživatele povodný, přehledný a snadno pochopitelný. K tomu jsou využívána právě grafická prostředí systém Windows postavená na 32 bitové architektuře.

Ad 3. Propojení jednotlivých procedur na úrovni API funkcí

API funkce výrazně zrychlují běh programu a hlavně zmenšují nároky na hardware. Tento krok vyřazuje nejnáročnější část práce na celém programu. Právě zde se projeví, jestli vynaložené úsilí při tvorbě přineslo očekávané výsledky.

Ad 4. Tvorba systémové logiky

Tvorba systémové logiky vychází z požadavků na výukový software. Při deklaraci proměnných je třeba použít vhodný datový typ, a nespolehat tak na datový typ Variant (vysvětleno v kapitole 5), který je univerzální. Program pak využívá centrální srovnávací a kontrolní modul.

Ad 5. Připravení prostředí pro potřeby distribuce programu

Závěrem tvorby aplikace je důležité, aby konečná verze programu byla upravena pro fungování na počítačích s různou konfigurací hardware. Hlavní je třeba, aby byla odlišná od konfigurace počítače, kde byl program tvořen. Problém je nutno vyřešit vytvořením základního modulu, jehož úkolem je zaznamenat velikost aktuálního rozlišení a následně převést programový formulář do odpovídajícího rozměru.

(Klement, 2002)

3.4.2 Univerzální výukový program

Vzhledem k tomu, že ne každý učitel je zbyhlý ve vytváření výukových programů, lze pro tyto případy sestavit výukový software, který sice bude obsahově prázdný, ale i tak bude plnit konkrétní požadavky. Tyto univerzální výukové systémy je možno poté libovolně doplňovat potřebnými informacemi. Jejich vytvoření je sice velmi náročné a problematické, ovšem jejich využití je naopak snadné a zaručuje vysokou flexibilitu vyučovacího procesu.

Znanou nevýhodou ovšem zůstává, že tento program neumí mít strukturu prezentace, nicméně ho lze naplnit pokáždě jiným obsahem.

(Klement, 2002)

Jak již bylo zmíněno, tvorba tohoto programu je značně obtížná. K sestavení se využívá modulový systém, jehož problematika je natolik obsahově náročná, že již není dále zmíněna v této práci.

4 Stereometrie na ZTM

Pro vytvoření výukového programu na úrovni týkající se stereometrie jsem, jifi z vý-
zmín ných d vod , zvolila jazyk Visual Basic. Nejl p ejdeme k samotnému výukovému
programu a k popisu uřivatelského prost edí aplikace, která slouřfi pro jeho tvorbu, budeme
se chvíli v novat práv stereometrii. Následný text bude zahrnovat její za azení do RVP ZV
a rozepsání jednotlivého u iva stereometrie v konkrétních ro nících na základní -kole.
Povařluji v-ak za vhodné nejprve uvést její definici.

Stereometrie neboli geometrie v prostoru, se zabývá útvary, které nejde jifi umístit
do roviny. Do t chto útvar m řeme zahrnout t lesa jako je krychle, kvádr, jehlan, koule
apod. (maths.cz, 2010).

Stereometrie se řídí n kolika axiomy:

1. Dva r zné body ur ují práv jednu p ímku, která jimi prochází.
2. P ímka, která je ur ená dv ma body z roviny , leřfi rovn řl v této rovin .
3. Pokud je bod A společným bodem rovin a , potom mají tyto roviny společnou
p ímku, která bodem prochází. Krom této p ímky jifi ob roviny dal-í společné body
nemají.
4. Leřfi-li bod A na p ímce p a leřfi-li p ímka p v rovin , potom bod A leřfi v rovin .
5. Ke kařdé p ímce lze daným bodem vést práv jednu rovnob řku.
6. Libovolná p ímka d řlí prostor na dva vzájemn opa né poloprostory. Tato p ímka je
jejich hrani ní rovinou.
7. Rovina je jednozna n ur ena:
 - p ímkou a bodem, který na ní neleřfi
 - dv ma r znými rovnob řkami
 - dv ma r znými r znob řkami
 - t emi r znými body, které neleřfi na p ímce

(stereometrie.webz.cz, 2011)

4.1 Stereometrie v RVP ZV

RVP ZV zachycuje toto téma ve vzdělávací oblasti *Matematika a její aplikace*. Tato oblast je v dokumentu dle obsahu vzdělávacího obsahu pro první a pro druhý stupeň. Nás ovšem bude dále zajímat pouze druhý stupeň, který je zde dle obsahu vzdělávacího obsahu rozdělen na několik tematických okruhů. Jedná se o okruhy: *Číslo a proměnná; Závislosti, vztahy a práce s daty; Geometrie v rovině a v prostoru a Nestandardní aplikační úlohy a problémy*. Problematika stereometrie spadá tedy do vzdělávacího obsahu druhého stupně pod tematický okruh *Geometrie v rovině a v prostoru*.

Dokument RVP ZV (2013) uvádí tyto očekávané výstupy žáka:

- zdvojnásobuje a vyúsťuje a vyúsťuje polohové a metrické vlastnosti základních rovinných útvarů a jejich úloh a jednoduchých praktických problémů; vyúsťuje potřebnou matematickou symboliku
- charakterizuje a určuje základní rovinné útvary
- určuje velikosti úhlu měřením a výpočtem
- odhaduje a vypočítá obsah a obvod základních rovinných útvarů
- vyúsťuje pojem množina všech bodů dané vlastnosti k charakteristice útvaru a k měření polohových a nepolohových konstrukčních úloh
- namaluje a sestrojí rovinné útvary
- uskloňuje argumentaci a používá výpočty o shodnosti a podobnosti trojúhelníků
- namaluje a sestrojí obraz rovinného útvaru ve středové a osové souměrnosti, určuje osové a středové souměrné útvary
- určuje a charakterizuje základní prostorové útvary (tělesa), analyzuje jejich vlastnosti
- odhaduje a vypočítá objem a povrch těles
- namaluje a sestrojí síť základních těles
- namaluje a sestrojí obraz jednoduchých těles v rovině
- analyzuje a řeší aplikační geometrické úlohy s využitím osvojeného matematického aparátu

4.2 Stereometrie ve výuce na ZTM

(erpáno ze TMkolního vzdávacího programu základní školy Heyrovského v Olomouci, p íloha . 2 zobrazuje ást vzdávací oblasti Matematika a její aplikace, konkrétn osnovy p edm tu)

6. t ída

íáci se poprvé setkají s geometrií v prostoru v 6. t íd . U ívo nese název *Povrch a objem krychle a kvádru*. Zahrnuje:

- kvádr, krychle, zobrazování, síť t les
- povrch krychle, kvádru
- jednotky objemu
- objem krychle, kvádru

O ekávané výstupy íáka:

- ur uje a charakterizuje základní prostorové útvary (t lesa ó krychle a kvádr), analyzuje jejich vlastnosti
- na rtne a sestrojí obraz jednoduchých t les (krychle a kvádr) v rovin
- na rtne a sestrojí síť krychle a kvádru
- odhaduje a vypo ítá objem a povrch krychle a kvádru
- analyzuje a e-í aplika ní geometrické úlohy s vyuíitím osvojeného matematického aparátu

Díl í výstupy:

- charakterizuje jednotlivá t lesa (kvádr, krychle)
- na rtne a narýsuje síť a z ní t leso vymodeluje
- na rtne a sestrojí obraz krychle a kvádru ve volném rovnob ílném promítání
- vypo ítá povrch krychle a kvádru
- uíívá jednotky objemu a vzájemn í je p evadí
- odhaduje a vypo ítá objem krychle a kvádru

7. téma

V sedmém tématu je stereometrii v novém prostoru v učivu *Povrch a objem hranolu*.

Toto učivo zahrnuje

- opakování kvádra a krychle o objem a povrch
- pojem hranol
- povrch a objem hranolu

Očekávané výstupy:

- určí a charakterizuje základní prostorové útvary o tělesa, analyzuje jejich vlastnosti
- odhaduje a vypočítá objem a povrch hranolu
- na rovině a sestrojí obraz hranolu v rovině
- na rovině a sestrojí síť hranolu

Dílejší výstupy:

- dokáže rozpoznat a pojmenovat různé typy hranolů
- je schopen vyjmenovat jejich vlastnosti
- na rovině a narýsuje obrazy různých hranolů v rovině
- dokáže si představit a narýsovat síť různých hranolů v rovině
- odhaduje a vypočítá objem a povrch hranolu, správně určí jednotky výsledků

8. téma

V osmém tématu se žáci setkávají s *válcem*. Náplní tohoto učiva je:

- pojem válec
- síť válce
- povrch a objem válce
- slovní úlohy

Očekávané výstupy:

- určí a charakterizuje základní prostorový útvar (těleso - válec)
- analyzuje jeho vlastnosti

- odhaduje a vypočítá objem a povrch válce
- namontuje a sestrojí síť válce

Dílčí výstupy:

- pozná válec a popíše ho jako těleso
- dokáže sestrojit síť válce
- vypočítá povrch a objem válce, určí správné jednotky výsledku
- analyzuje a řeší slovní úlohy na objem a povrch válce

9. tělesa

V devátém třídě je prostor v nově zbyvajícím tělesem. Učivo *Tělesa* tedy zahrnuje:

- kufel
- jehlan
- koule
- povrch a objem těles

Očekávané výstupy:

- určí a charakterizuje základní prostorové útvary (tělesa), analyzuje jejich vlastnosti
- řeší úlohy na prostorovou představivost, aplikuje a kombinuje poznatky a dovednosti z různých tematických a vzdělávacích oblastí

Dílčí výstupy:

- namontuje a sestrojí síť jehlanu a kufele
- odhaduje a vypočítá objem a povrch těles

5 Visual Basic

Tato kapitola je vnována jak samotnému programovacímu jazyku, tak i nástinu prostředí programu, který využívá jazyk Visual Basic. Zvolení tohoto jazyka je objasneno již v dřívejších kapitolách. Dříve ovšem neř se mu za neme vnovat, pokládám za dlefité, objasnit pojmy, které s ním bezpochybně souvisejí, a bez kterých by mohl být další text nesrozumitelný. Proto jsou první dvě podkapitoly vnovány problematice programovacího jazyka a pro VB dlefitému bodu, a to objektově orientovanému programování.

5.1 Program

Programem nazýváme sestavenou aplikaci, ve které je ve formě zdrojového kódu zapsán soubor instrukcí. Tento zdrojový kód má podobu textového souboru, který obsahuje algoritmy programu. Abychom mohli mluvit o programu, je potebné zmínit dlefitou vlastnost, a to, že aplikace musí být spustitelná na počítači (Klement, 2002).

5.1.1 Programovací jazyky a příklady

Programovací jazyk Klement (2002) definuje jako: šumělý jazyk vytvořený pro zápis počítačového programu. Tyto jazyky mají přesně definovanou syntaxi, což je systém symbolů a pravidel, kterými se řídí formální zápis programu, a sémantiku, která určuje význam programu.

Dle Klementa (2002) programovací jazyky dleíme podle stupně abstrakce na:

- jazyky strojově orientované
- vyšší programovací jazyky

Dle běžným pojmem je prekladač neboli compiler, který převádí zdrojový kód programu z programovacího jazyka do strojového kódu. Prekladač má dle Klementa (2002) podobu:

a) kompilátor

- prekladá všechny příkazy najednou, spustit ho lze až po ukončení celého prekladu
- používá se u programovacích jazyků FORTRAN, PASCAL, ADA, aj.

b) interpret

- příkazy zdrojového kódu zpracovává jednotlivě, tzn., při každém spuštění je program znovu přeložen, z toho vyplývá nevýhoda jeho nízké rychlosti
- používá se u programovacích jazyků BASIC

Pouze pro doplnění a vytvoření přehledu uvádím konkrétní jazyky jednotlivých stupňů abstrakce. Text je převzat z Klementa (2002):

a) jazyky strojově orientované

- o Assembler (jazyk symbolických adres)

b) vyšší programovací jazyky

- o FORTRAN (Formula Translation),
- o BASIC (Beginner's All-Purpose Symbolic Instruction Code),
- o COBOL (Common Business Oriented Language),
- o Algol (Algorithmic Language),
- o PL/1 (Programming Language 1),
- o Lisp (List Processing),

- ó C (C Language),
- ó Pascal a Turbo Pascal,
- ó ADA, í .

5.2 Objektov orientované programování

Objektov orientované programování (dále jen OOP) patří mezi moderní metody. Tato metoda klade důraz na vlastnost řešení problému tím, že plně podporuje modularitu, abstrakci dat, srozumitelný zápis a v neposlední řadě spolehlivost programu.

Princip OOP je takový, že je potřeba vyvinout funkční tvůrčí vzory, které se jejich vlastností přenášejí na nový objekt. Tato myšlenka je rozvinuta o spojení dat s algoritmy, tzn., objekty jsou popisovány jejich vzájemné vztahy.

V dnešní době většina programovacích jazyků pracuje na principu OOP. Tato metoda je zvláště vhodná pro použití u aplikací, které využívají grafické prostředí typu MS Windows (dnes nepoužívaný operační systém), takže se předpokládá, že za pár let bude OOP využívat většina programátorů.

(Klement, 2002)

5.3 Úvod do jazyka Visual Basic

Programovací jazyk Basic vytvořili v roce 1963 John G. Kemeny a Thomas E. Kurtz z Dartmouthské univerzity. Do jeho nástupu se museli programátoři soustředít na metody a algoritmizace, které vyřadili konkrétní hardware po fázi sestavení a ladění programu. Ovšem právě se vznikem jazyka Basic tato nutnost zmizela a programátor se mohl soustředít na metody a algoritmy, které byly potřebné k vyřešení úlohy (Klement, 2002).

Postupem času vznikl Visual Basic, který byl vyvinutý firmou Microsoft a který byl právě založen na jazyku Basic. VB byl prvním jazykem, který poskytoval graficky programovatelné prostředí pro vývoj uživatelského rozhraní. Místo toho, aby se programátor zabýval syntaktickými detaily, přidává značné množství kódu pouhým přetažením objektu

pomocí myši. Jedná se například o tlačítka a dialogové okna, u kterých definujeme jejich vzhled a chování. VB je založený na filozofii objektově orientovaného programování, i když ho takovým jazykem v pravém slova smyslu nazvat nemůžeme. Spíše se užívá event-driver jazyk, což vyjadřuje, že každý objekt reaguje na nějakou událost, jako je například kliknutí myši (webopedia.com, 2014).

Klement (2002) definuje programovací jazyk Visual Basic jako: š programovací systém vytvořený firmou Microsoft pracující v operačním systému Windows a sloužící k vývoji a správě aplikací pod Windows.

Tím, že lze do takové míry využít operační systém, odpadá programátorovi řada rutinních úloh. Tvoří se mu v nové tvorbě podstaty aplikace a nemusí ji programovat vzhled nebo chování. Zkrátka lze tvrdit, že se VB orientuje na práci s objekty definované již programem nebo operačním systémem. Z toho plyne značná výhoda tzv. švisual- produktů, že začínající a nemotorný programátor může vytvořit poměrně efektivní aplikaci (Klement, 2002).

5.4 Přehled vývojových verzí jazyka VB

V dnešní době je nejvíce využívána verze 6 (VBA⁹) a rovněž verze .NET. Pro přehled o vývoji jsou níže vyjmenovány další verze.

1. Verze 4.0 umožnila vytvořit jak 16 - bitové aplikace (pro Windows 3.1), tak i 32 - bitové aplikace (pro Windows 95/NET)
2. Verze 5.0 oproti verzi 4.0 má lepší grafické prostředí, navíc obsahuje lokální nápovědy. Lze v ní vytvářet pouze 32 - bitové aplikace (pro Windows95/NET)
3. Verze 6.0 o klasická verze, umožnila využít VBA (Word, Excel, Access)
4. Verze .NET (verze 7.1, 8, 9, 10)

(Klement, 2002)

⁹ Neboli Visual Basic for Application. Více informací na <http://www.jakprogramovat.cz/lekce-vba/uvod-do-visual-basic-application-vba>

5.4.1 VisualBasic .NET

Jelikož vývojové prostředí, které je popsáno níže, a ve kterém je tvořen výukový program, je psáno v tomto jazyce, je dalších pár řádků v nově jeho problematice.

Vznik této verze je popisován nespokojeností mnoha uživatelů. Proto kolem roku 2000 začal Microsoft vyvíjet .NET Framework¹⁰ 1.0. Jednalo se o velmi šilně prostředí. Když se uvažovalo, ke kterému programovacímu jazyku ho přidat, zvolil Microsoft jazyk Visual Basic. Je zřejmé, že ho musel přizpůsobit, takže původní VB6 byla změněna. Byla zrušena sada funkcí, které nahradili funkce .NETu. Zapomínalo to, že se VisualBasic .NET stal velmi plnohodnotným programovacím jazykem kompletně podporující objektově orientované programování (dotnetportal.cz, 2014).

5.5 Vývojové prostředí Microsoft Visual Studio 2010 Express

Pro tvorbu výukového programu popisovaného v praktické části jsem zvolila vývojové prostředí Microsoft Visual Studio 2010 Express, které využívá programovací jazyk VisualBasic .NET. Jedná se o novější verzi, uživatelsky přijatelnou a přehlednou. Nyní se zaměříme na popis prostředí této verze, její znalost je nezbytná jak pro orientaci v programu, tak i jeho samotném vytvoření.

Každé okno této aplikace má několik částí:

1. *Záhlaví okna s ovládacími prvky.* Toto je společné pro všechny programy firmy Microsoft. Nachází se v pravém horním rohu a obsahuje tlačítka pro minimalizaci, maximalizaci, obnovení a ukončení okna.
2. *Ovládací menu s příkazy,* které jsou rozloženy podle druhu operace. Podle jazykové verze programu zde můžeme najít:
 - o nabídku File (Soubor) sloužící k práci s projekty a formuláři (název pro soubory ve VB),
 - o nabídku Edit (Úpravy) umožňující úpravu formulářů,
 - o nabídku View (Zobrazit) pro zobrazení projektu,

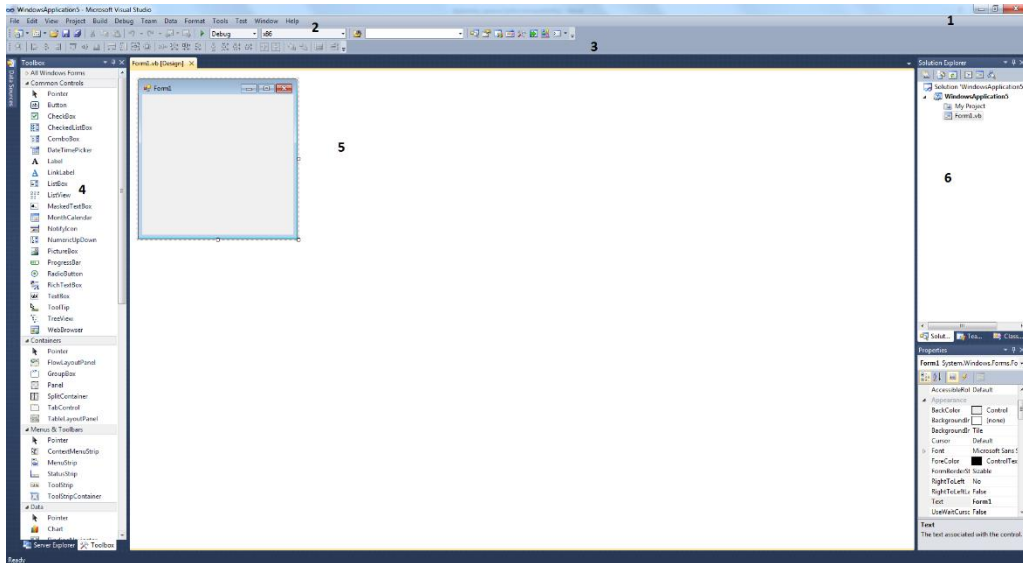
¹⁰.NET Framework je prostředí, ve kterém se vyvíjejí aplikace. Ty pak ke svému spuštění potřebují mít nainstalované knihovny z tohoto prostředí (cs.vsb.cz, 2007).

- ó nabídku Project (Projekt) sloužící pro vkládání různých částí projektu,
 - ó nabídku Format (Formát) umožňující upravování objektů ve formulářích,
 - ó nabídku Debug (Lazení) určenou k nastavení chování aplikace,
 - ó nabídku Build (Spustit) sloužící ke spuštění programu,
 - ó nabídku Tools (Nástroje) umožňující vkládání do vývojového prostředí prvky ActiveX,
 - ó nabídka Windows (Okno) dovolující zobrazit více projektů naráz a nakonec
 - ó nabídka Help (Nápověda) obsahující potřebné informace pro práci ve Visual Basic.
3. *Nástrojové lišty.* Ty jsou opět typické pro různé aplikace, liší se v konkrétních, nejčastěji používaných tlačítkách aplikace.
 4. *Plochy pro zobrazení projektu a formuláře.* Jedná se o pracovní plochu programu. Zde se zobrazuje program, který vytváříme.
 5. *Informační lišty.* Zde se nacházejí informace o právě tvořeném i upravovaném projektu nebo formuláři. Tyto lišty se skládají z:
 - a. Project Explorer (Správce projektu)
 - b. Properties (Vlastnosti)

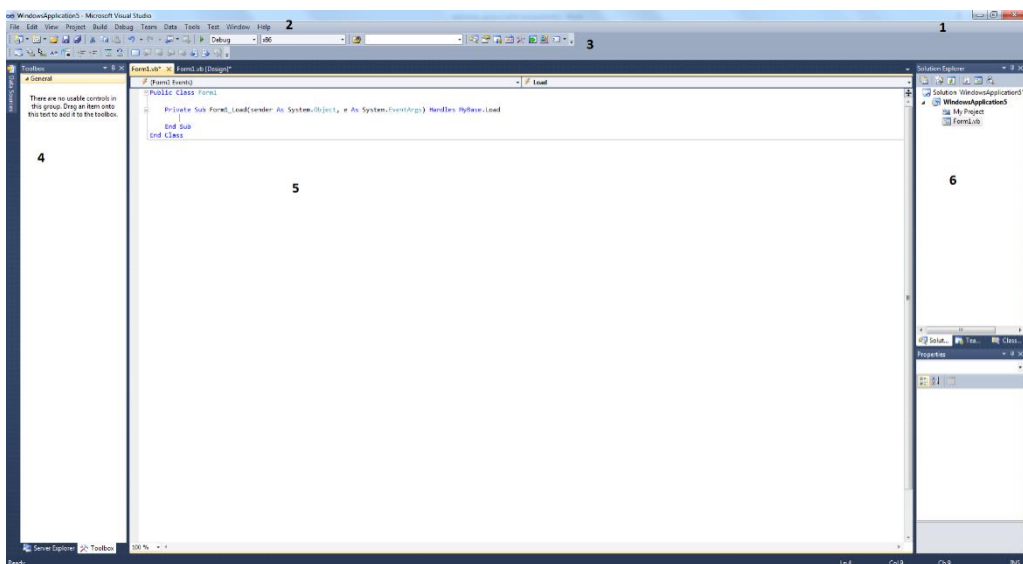
(Klement, 2002)

Při vytváření aplikace pracujeme se dvěma okny. Jedno slouží k vytvoření vizuální podoby, tzn. rozmístění tlačítek, textových polí, apod. Druhé, významnější, slouží pro vytvoření funkčního programu zapsaného programovacím jazykem ve formě příkazů. Toto okno zobrazíme, když klikneme na formulář zobrazený na obrázku číslo 2. Mezi okny přepínáme pomocí záložky, která se nachází nad pracovní plochou.






















Obě okna jsou zobrazena na obrázcích číslo 2 a 3. Obrázek číslo 4 pak tyto okna popisuje dopodrobna.



Obrázek 2: Okno MS Visual Studio 2010 Express p i grafické tvorbě programu



Obrázek 3: Okno MS Visual Studio 2010 Express p i psaní kódu

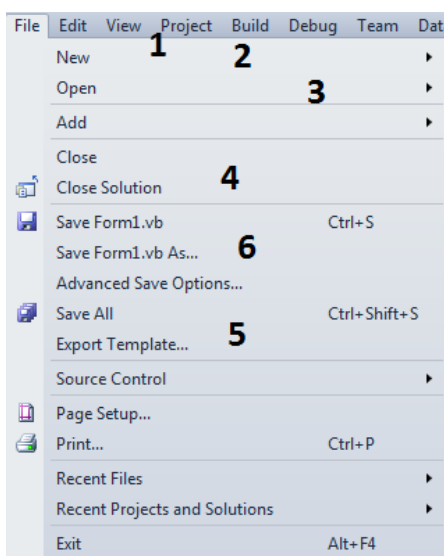
1	<p>Záhlaví okna aplikace Visual Basic s ovládacími prvky okna.</p> <ul style="list-style-type: none"> - Minimalizovat - Maximalizovat / Obnovit - Zavřít
2	<p>Ovládací menu. Nejdůležitější z položek jsou vysvětleny v dalších následujících kapitolách.</p>
3	<p>Nástrojová lišta. Obsahuje tyto funkce:</p> <ul style="list-style-type: none">  Tlačítko NOVÝ PROJEKT (vytvoří nový projekt Visual Basic).  Tlačítko OTEVŘÍT PROJEKT (otevře vytvořený projekt MS Visual Basic).  Tlačítko PŘIDAT NOVOU POLOŽKU K PROJEKTU (Form, Modul, Class Modul, User Control).  Tlačítko ULOŽIT AKTUÁLNÍ FORMULÁŘ (Form1.vb).  Tlačítko ULOŽIT VŠE (Uloží všechny soubory související s aktuálním projektem - WindowsApplication1.sln, FormX.vb ...).  Tlačítko VYJMOUT (vyjme TEXT, OBJEKT).  Tlačítko KOPÍROVAT (zkopíruje TEXT, OBJEKTY do systémové schránky).  Tlačítko VLOŽIT (vloží TEXT, OBJEKTY ze systémové schránky).  Tlačítko HLEDAT (najde ve zdrojových kódech hledané slovo, slovní spojení). <i>Pozn. Lze přepnout do módu umožňujícího nahrazování textu.</i>  Tlačítko PŘIDEJ KOMENTÁŘ k aktuálnímu řádku (přidá komentář k aktuálnímu řádku, komentáře jsou výhradně psané zelenou barvou pro snadné odlišení od zdrojových kódů).  Tlačítko ODEBER KOMENTÁŘ (odebere komentář od aktuálního řádku).  Tlačítko ZPĚT (vrátí zpět provedené změny).  Tlačítko DALŠÍ (posune se k provedeným změnám).  Tlačítko PŘEPNI NA DALŠÍ ZÁLOŽKU (přepíná mezi záložkami daným směrem).  Tlačítko PŘEPNI NA DALŠÍ ZÁLOŽKU (přepíná mezi záložkami daným směrem).  Tlačítko SPUSTĚNÍ APLIKACE (spustí aktuálně vytvářenou aplikaci pro ladění)  Tlačítko ZAPAUZUJ SPUŠTĚNOU APLIKACI (zastaví vytvářenou aplikaci pro ladění)  Tlačítko ZASTAV SPUŠTĚNOU APLIKACI (úplně zastaví vytvářenou aplikaci pro ladění)  Tlačítko KROKOVÁNÍ PROJEKTU (skok na další pozici)  Tlačítko KROKOVÁNÍ PROJEKTU (skok na další vybranou pozici)  Tlačítko KROKOVÁNÍ PROJEKTU (skok na předchozí pozici)
4	<p>Nástrojový panel. Obsahuje jednotlivé nástroje či objekty, které je možné umísťovat do formulářů.</p>
5	<p>Plocha pro zobrazení projektu a jeho součástí (formulářů, kódů, tříd nebo modulů). V této ploše se zobrazuje vlastní formulář, který vytváříme. Zobrazení je možné nastavit podle individuálního nastavení a požadavků uživatele. Formulář se zobrazuje jako šedý panel. Okno příslušného kódu se potom zobrazuje jako stránka ve Wordu.</p>
6	<p>Okno s vlastnostmi (Properties). V něm nastavujeme vlastnosti právě navrhovaného formuláře nebo vytvářeného ovládacího prvku. Například u tlačítka zde volíme text zobrazený na tlačítku, použité písmo (font), stav tlačítka a další.</p> <p>Okno s projektem. Obsahuje přehled všech komponent, ze kterých se skládá aplikace. Velmi jednoduché aplikace mají jen jednu komponentu - formulář. Složitější se běžně skládají z více komponent.</p>

Obrázek 4: Popis oken v MS Visual Studio (Klement, 2002)

5.5.1 Nabídka FILE (Soubor)

V této záložce najdeme všechny potřebné příkazy, které používáme při globální práci s projektem. Dále jsou zde zakomponovány příkazy spojené s nastavením celého projektu nebo formulářů (Klement, 2002).

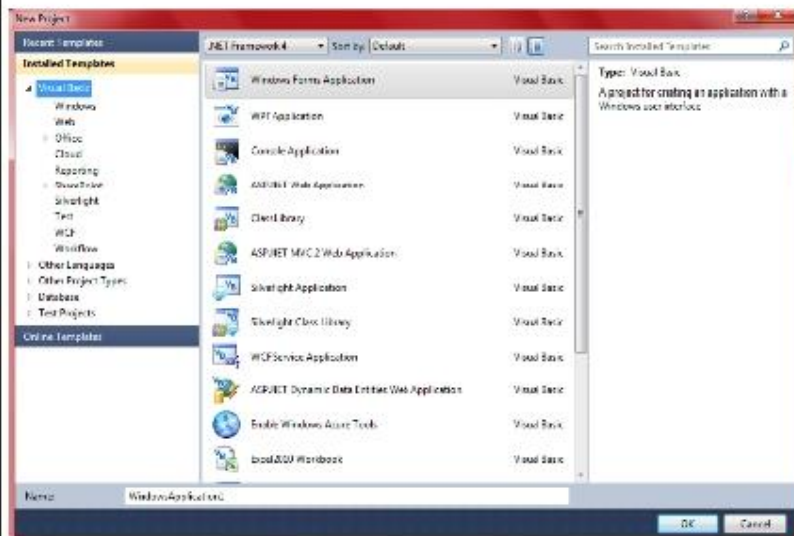
Obrázek 5 znázorňuje nabídku záložky FILE, obrázek 6 popisuje jednotlivé části.



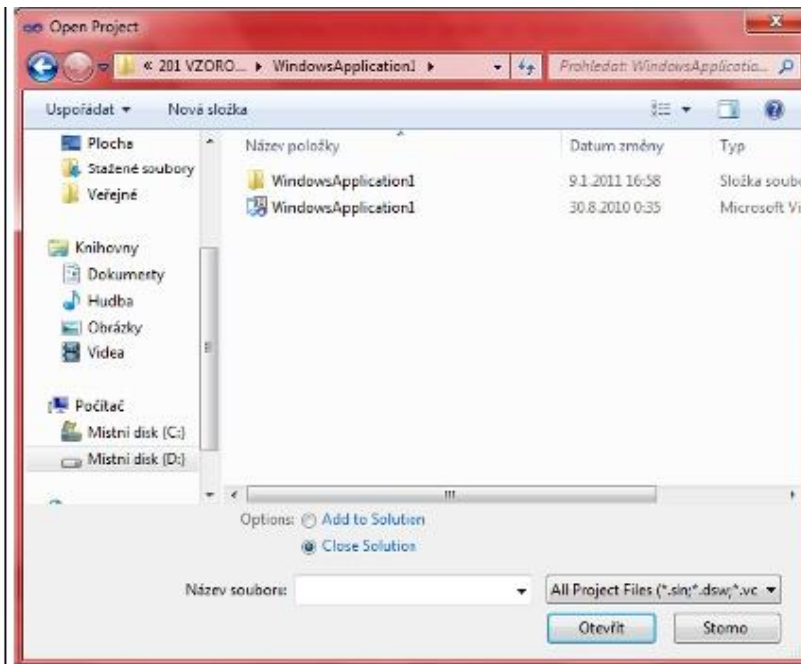
Obrázek 5: Nabídka FILE

- 1 Polozka ovládacího menu FILE (Soubor)
- otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši
- dojde k otevření položky. Pohyb v ní je shodný s pohybem v hlavním menu.

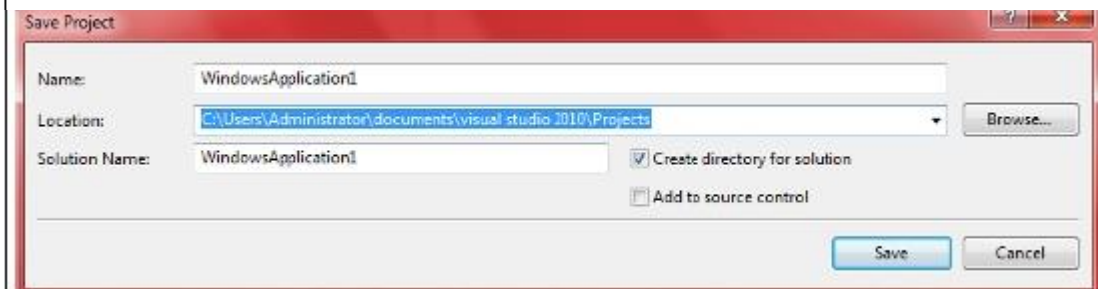
- 2 Volba NEW PROJECT (Nový projekt). Spustí průvodce pro vytvoření nového projektu.



- 3 Volba OPEN PROJECT (Otevřít projekt). Spustí průvodce, který umožní otevřít existující projekt, jenž je uložen na některém z paměťových médií.



- 4 Volba CLOSE PROJECT (Zavřít projekt). Slouží k uzavření aktuálně zobrazeného projektu. Nedojde k jeho odstranění z pevného disku pouze k jeho uzavření (odstranění z operační paměti RAM).
- 5 Volba SAVE ALL (Uložit všechno). Spustí průvodce pro uložení aktuálně zobrazeného projektu a příslušných komponent (form, module atd.) pod určitým jménem.
- 6 Volba SAVE FORM.FRM AS (Uložit formulář jako). Spustí průvodce, který vám umožní uložit formulář pod určitým jménem. Projekt je zpravidla tvořen jedním nebo několika formuláři. Pokud chcete uložit celý projekt, budete vyzváni k uložení i formulářů.

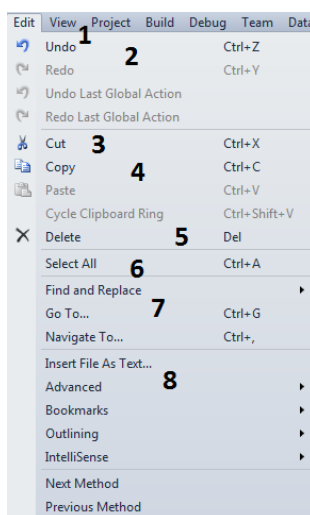


Obrázek 6: Popis nabídky FILE (Klement, 2002)

5.5.2 Nabídka EDIT (Úpravy)

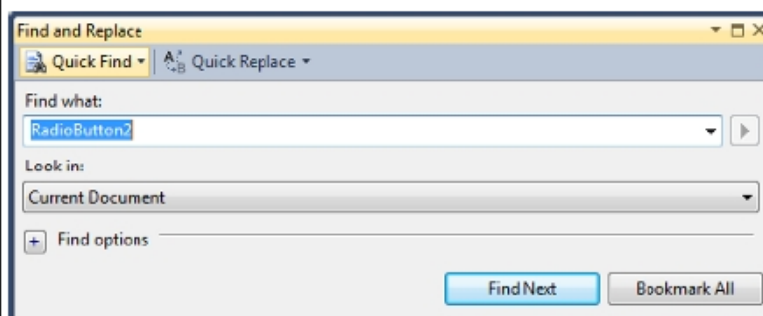
Zde nalezneme příkazy týkající se práce s obsahem konkrétního formuláře, například kopírování či vkládání části kódu, objektů apod. (Klement, 2002).

Obrázek 7 znázorňuje nabídku záložky EDIT, obrázek 8 popisuje jednotlivé části.



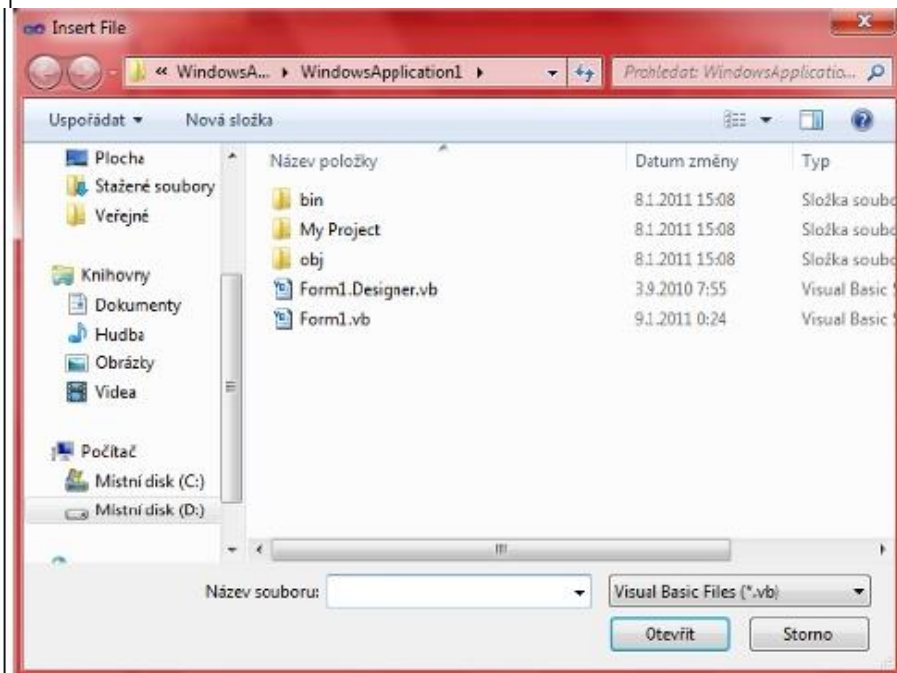
Obrázek 7: Nabídka EDIT

- | | |
|---|--|
| 1 | <p>Položka ovládacího menu (EDIT) Úpravy.</p> <ul style="list-style-type: none"> - otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši - dojde k otevření položky. Pohyb v ní je shodný s pohybem v hlavním menu. |
| 2 | <p>Volba UNDO (Zpět) a REDO (Opakovat). Tato tlačítka vám umožní vracet nebo opakovat poslední změny, které jste v projektu nebo formuláři provedli. Pokud například vymažete kód a následně zjistíte, že jej potřebujete, stačí stlačit tlačítko zpět a kód se znovu vrátí do textu.</p> <p>Program si pamatuje pouze ty změny, které se udály od posledního otevření projektu. Pokud projekt uložíte, nebude již tato volba dostupná!!!</p> |
| 3 | <p>Volba CUT (Vymnout). Tato volba vyjme z formuláře označenou část kódu nebo jakýkoli objekt, který je označen. Nelze vymout celý formulář, ale pouze jeho část. Tato vyjmutá část kódu nebo objekt se automaticky přesune do schránky, odkud může být znovu vyvolána.</p> |
| 4 | <p>Volba COPY (Kopírovat). Uloží označenou část textu či jiný objekt vložený do formuláře, do schránky. Volba PASTE (Vložit). Pomocí tohoto tlačítka můžete zkopírovaný text nebo objekt ze schránky umístit do libovolné části formuláře nebo dokonce i do jiného dokumentu. Schránka může vždy obsahovat pouze jednu položku. Pokud do schránky umístíte další položku, původní položka se vymaže!!!</p> <p>Neplatí pro Office 20XX, kde je možno zkopírovat do schránky až 20 různých položek.</p> |
| 5 | <p>Volba DELETE (Vymazat). Pomocí této volby se vymaže vybraná část kódu nebo objekt. Pokud jednou objekt vymažeme, tento se již neumístuje do schránky. Proto jej můžeme obnovit pouze pomocí tlačítka Zpět.</p> |
| 6 | <p>Volba SELECT ALL (Vybrat vše). Označí text, kódy a objekty v celém formuláři.</p> |
| 7 | <p>Volba QUICK FIND (Vyhledat). Spustí průvodce, který umožňuje vyhledat jednotlivé části kódu či dokonce součásti objektů a jejich vlastnosti. Pozn. V rámci formuláře se můžete přepnout i do režimu umožňujícího nahrazování části textu.</p> |



8

Volba INSERT FILE (Vložit soubor). Umožňuje vkládat do aktuálního projektu či formuláře obsah externích souborů nebo již dříve vytvořených formulářů a projektů.

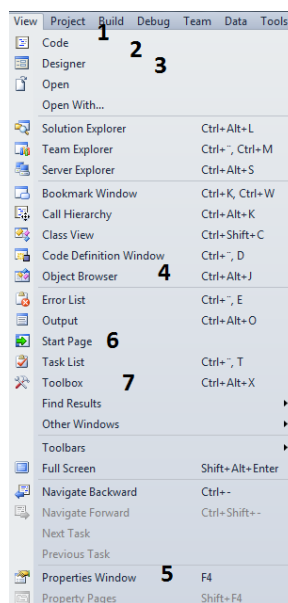


Obrázek 8: Popis nabídky EDIT (Klement, 2002)

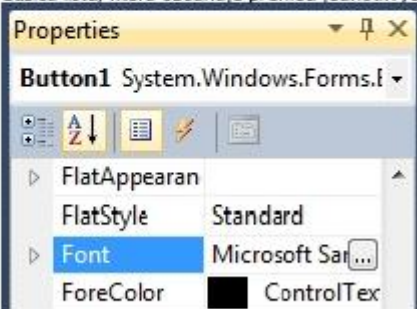
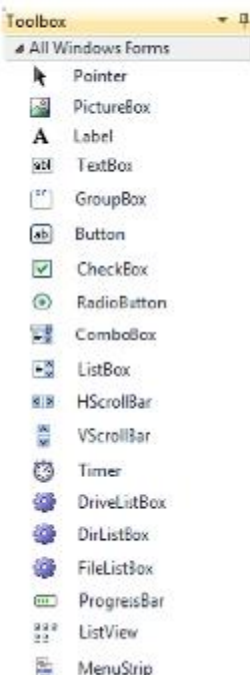
5.5.3 Nabídka VIEW (Zobrazit)

Zde najdeme příkazy, které se používají při nastavení zobrazení jednotlivých formulářů. Taktéž zde patří i příkazy jako je pravítko, nástrojové lišty apod. (Klement, 2002).

Obrázek 9 znázorňuje nabídku záložky VIEW, obrázek 10 popisuje jednotlivé části.



Obrázek 9: Nabídka VIEW (Zobrazit)

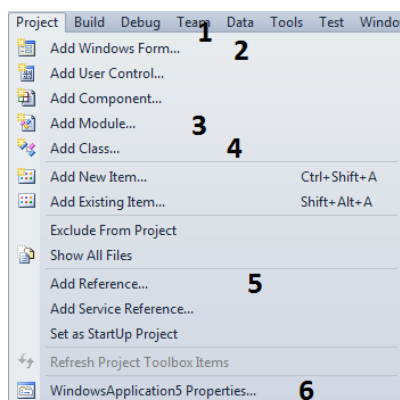
1	Položka ovládacího menu VIEW (Zobrazit) <ul style="list-style-type: none"> - otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši, - dojde k otevření položky. Pohyb v ní je shodný s pohybem v hlavním menu.
2	Volba nastavení zobrazení CODE (Zdrojový Kód). Zobrazí zdrojový kód aktuálně otevřeného formuláře či projektu. Pokud přidáme na formulář určitý objekt (např.: tlačítko), chceme, aby se při jeho aktivaci (stlačení apod.) stala nějaká činnost. Tuto činnost potom generuje právě obslužný (zdrojový) kód.
3	Volba nastavení zobrazení DESIGNER (Návrhový mód). Zobrazí okno s designovým návrhem aplikace. V tomto režimu sestavujeme grafickou podobu aplikace.
4	Tlačítko OBJECT BROWSER (Správce objektů). Toto tlačítko spustí průvodce, který obsahuje veškeré příkazy a definované funkce, které Visual Basic obsahuje.
5	Tlačítko PROPERTIES WINDOW (Vlastnosti). Pomocí tohoto tlačítka zobrazíte v pravé části okna Visual Basicu lištu, která obsahuje přehled jednotlivých volitelných vlastností objektů či formulářů. 
6	START PAGE, zobrazí titulní stránku aplikace MS Visual Basic .NET
7	Tlačítko TOOLBOX (Nástrojový panel). Pomocí tohoto tlačítka zobrazíte nebo skryjete svislý nástrojový panel v pravé části okna Visual Basicu. 

Obrázek 10: Popis nabídky VIEW (Klement, 2002)

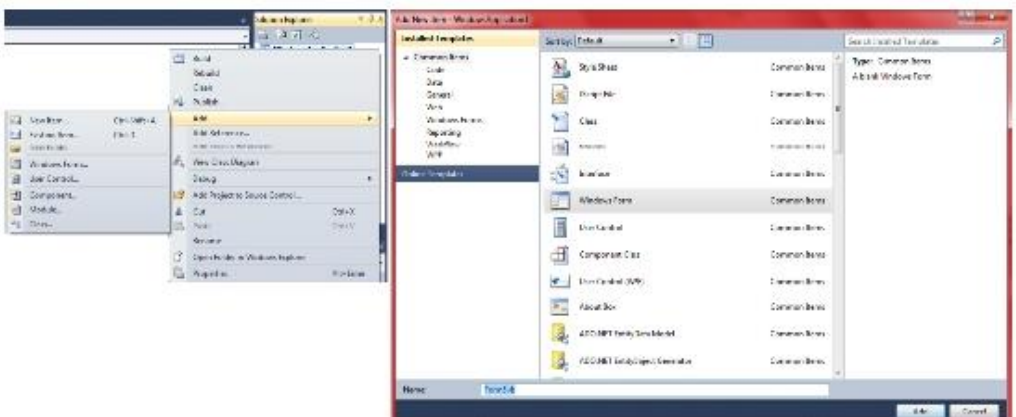
5.5.4 Nabídka PROJECT (Projekt)

V této nabídce jsou všechny předdefinované komponenty projektu, například prázdné formuláře, existující formuláře, moduly, třídy, soubory atd. (Klement, 2002).

Obrázek 11 znázorňuje nabídku záložky PROJECT, obrázek 12 popisuje jednotlivé části.

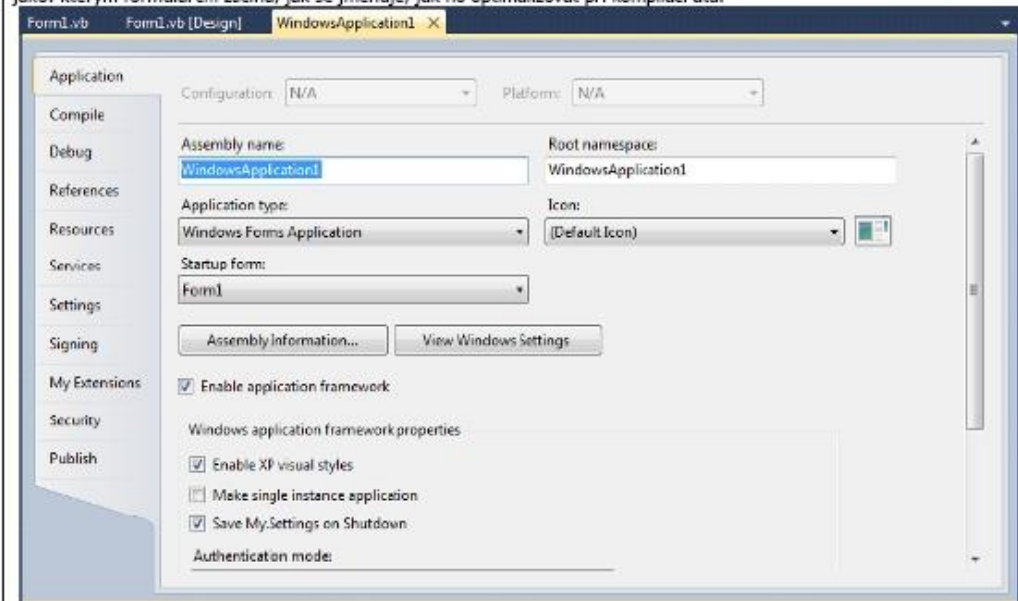


Obrázek 11: Nabídka PROJECT

- | | |
|---|--|
| 1 | <p>Položka ovládacího menu PROJECT (Projekt).</p> <ul style="list-style-type: none"> - otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši, - dojde k otevření položky. Pohyb v ní je shodný s pohybem v hlavním menu. |
| 2 | <p>Volba ADD FORM (Vložit formulář). Pomocí této volby můžete do aktuálního projektu vložit nový formulář. Formulář se automaticky umístí uže za poslední existující formulář projektu. Pokud mu nezadáte jiné jméno, bude jeho název <i>FormX.vb</i> (<i>X</i> je číslo o jedno vyšší než číslo posledního existujícího formuláře).</p>  |
| 3 | <p>Volba ADD MODULE (Vložit modul). Pomocí této volby můžete do aktuálního projektu vložit modul. Modul slouží k deklaracím funkcí a proměnných, které využíváme na dvou a více místech projektu. Deklarovaným proměnným na modulu se říká globální proměnné.</p> |
| 4 | <p>ADD CLASS MODULE (Vložit modul třídy). Pomocí této volby vložíte do aktuálního projektu class modul. Stručně řečeno je třída nebo modul formulář, který se nezobrazuje a je stále zaveden v paměti, může obsahovat takové objekty a funkce, které jsou společné pro několik formulářů a je tedy zbytečné je dávat do každého formuláře zvlášť.</p> |
| 5 | <p>ADD REFERENCE (Komponenty). Pomocí této volby spustíte seznam všech objektů, které můžete přidat na nástrojový pruh (ToolBox). Visual Basic obsahuje přibližně 200 různých objektů a nástrojů, které se běžně ve Windows nacházejí (přehrávač multimédií, textová pole atd.). Pokud potřebujete některý z objektů, který není obsažen na standardním nástrojovém pruhu, vložit do svého projektu, použijte tohoto průvodce.</p> |

6

PROPERTIES (Vlastnosti projektu). Pomocí této volby můžete zobrazit vlastnosti celého projektu včetně zvláštních funkcí jako: kterým formulářem začíná, jak se jmenuje, jak ho optimalizovat při kompilaci atd.

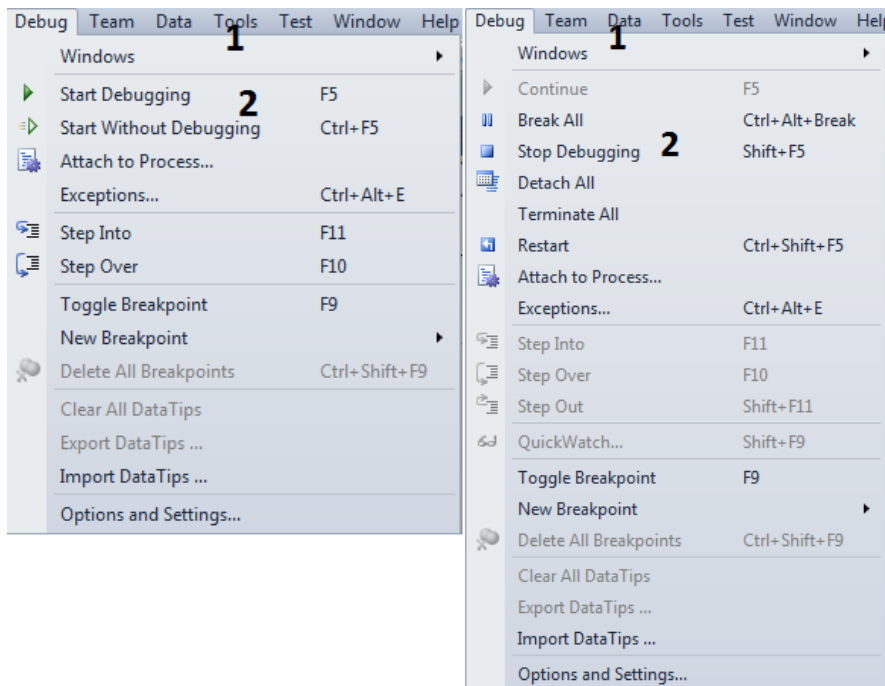


Obrázek 12: Popis nabídky PROJECT (Klement, 2002)

5.5.5 Nabídka DEBUG (Spustit)

V nabídce jsou přítomny příkazy, které se týkají spuštění aplikace. Spuštění aplikace se chová jako jistý reálný program, na kterém ověřujeme jeho funkčnost. Velký vliv to má na nalezení chyby v aplikaci (chybně zapsaný kód, špatná instrukce atd.). Při zaznamenání chyby přejde program do ladícího režimu Debug a v kódu se označí řádek, ve kterém je chyba. VB proto dává uživateli oproti ostatním programům značnou výhodu, když umožní ujet spuštění projektu, který ještě nebyl zcela dokončen (Klement, 2002).

Obrázek 13, který je rozdělen na dvě části, znázorňuje nabídku záložky PROJECT. První část zobrazuje nabídku, když chceme program spustit, druhá část naopak ukazuje nabídku jistého spuštěného programu. Obrázek 14 pak popisuje jednotlivé části.



Obrázek 13: Nabídka DEBUG

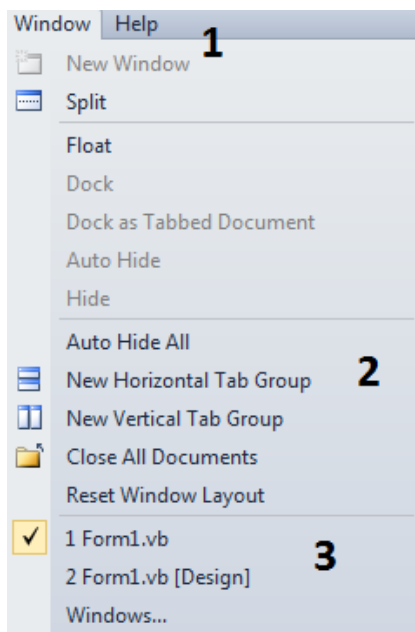
1	<p>Položka ovládacího menu DEBUG (Spustit)</p> <ul style="list-style-type: none"> - otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši - dojde k otevření položky. Pohyb v ní je shodný s pohybem v hlavním menu.
2	<p>Volba START DEBUGGING (Spustit aplikaci). Pomocí této volby (přejdete do režimu RUN) spustíte projekt a ten se bude chovat jako reálná aplikace a můžete tak vyzkoušet její funkčnost. Pokud dojde za běhu takto spuštěné aplikace k chybě (špatně volaná instrukce, chybně zapsaný kód, syntaktická chyba, volání objektu, který neexistuje atd.), přejde aplikace do ladícího režimu Debug, zobrazí řádek kódu, ve kterém se vyskytuje chyba.</p> <p>Volba STOP DEBUGGING (Konec ladění). Pomocí této volby se opět vrátíte do návrhového režimu (režim DESIGN) a můžete pracovat na dalších úpravách aplikace.</p>

Obrázek 14: Popis nabídky DEBUG (Klement, 2002)

5.5.6 Nabídka WINDOW (Okno)

Záložka slouží k rozvržení okna v prostoru. Navíc díky této záložce jde vidět, které projekty jsou otevřeny, a ve kterých se pracuje (Klement, 2002).

Obrázek 15 znázorňuje nabídku záložky WINDOW, obrázek 16 popisuje jednotlivé části.



Obrázek 15: Nabídka WINDOW

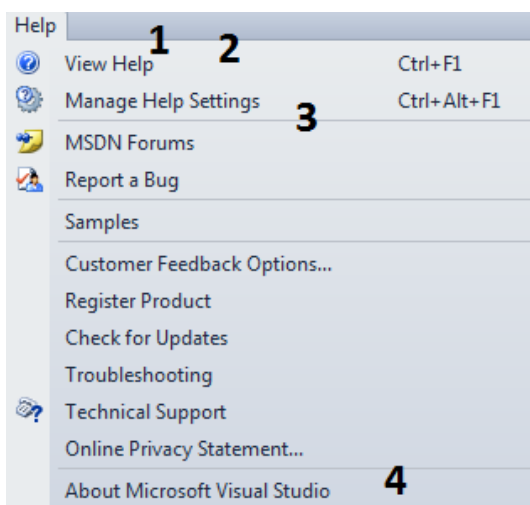
1	Položka hlavního menu WINDOW (Okno). - otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši, - dojde k otevření položky. Pohyb v ní je shodný s pohybem v hlavním menu.
2	Volba TILE HORIZONTALLY – VERTICALLY (Nové horizontální – vertikální okno). Tato volba otevře nové okno, ve kterém zobrazí stávající otevřený projekt.
3	Informační panel. Tento panel vás informuje o tom, kolik a jakých projektů je aktuálně otevřeno. U projektu, který je právě aktivní (je s ním právě pracováno), je před jeho názvem zvýrazněno zatržení.

Obrázek 16: Popis nabídky WINDOW (Klement, 2002)

5.5.7 Nabídka HELP (Nápověda)

Tato záložka slouží k tomu, aby uživateli poradila, kde hledat nebo jak použít některou funkci nebo kde získat potřebné informace. Nápověda rovněž zahrnuje i *Průvodce*, který má za úkol provést programátora celým problémovým řešením. Nápovědu je možné získat umístěním kurzoru na vybraný prvek a stisknutím tlačítka F1. To způsobí otevření nápovědy (Klement, 2002).

Obrázek 17 znázorňuje nabídku záložky HELP, obrázek 18 popisuje jednotlivé části.



Obrázek 17: Nabídka HELP

1	Položka hlavního menu HELP (Nápověda). - otevření položky: přesuňte kurzor myši nad tuto položku a stlačte jednou levé tlačítko myši, - dojde k otevření položky. Pohyb v ní shodný jako pohyb v hlavním menu.
2	Nápověda pro Microsoft VISUAL BASIC .NET 2010. Tato volba spustí průvodce aplikací Visual Basic – Pomocníka. Tento pomocník se snaží sledovat vaši práci a radit vám možné alternativy řešení problému (případně vyhledat pomoc v síti internet).
3	Nastavení nápovědy (možnosti on-line atp.)
4	Informace o aplikaci Visual Basic. Zde jsou informace o verzi programu MS Visual Basic .NET 2010, servisních balíčcích, oprávněném uživateli licence a výrobci daného produktu.

Obrázek 18: Popis nabídky HELP (Klement, 2002)

5.6 Proměnné

Při vytváření programu je v první řadě nutné definovat a deklarovat proměnné. Proměnné v VB představují držitele prostoru, do kterého zanecháváme hodnoty podle naší potřeby. Představte si, píšete program, který zobrazuje určitá data pro různé zadání. Například po výzvě vybrat uživatele, například Petra Nováka, se spustí program, kde bude napsán Petr Novák a budou se mu výsledky z úloh zapisovat do jeho listu v tabulkovém editoru. Abychom nemuseli psát jinou aplikaci například pro Marii Konečnou, deklaruje v aplikaci proměnnou, která uchová jméno uživatele zapsané do připraveného textového okna po spuštění programu. Tuto proměnnou program následně uloží a bude ji zobrazovat při práci Petra Nováka ve zvolených formulářích.

Promenná obsahuje jméno a hodnotu. Např. promenná *Uživatel* může nabývat hodnoty šAnnao, nebo promenná *Sleva* může nabývat hodnoty 0,5. *Uživatel* a *Sleva* jsou jména promenných a šJosefo a 0,5 jsou jejich hodnoty. Platí pravidlo, že pokud je hodnota promenné text nebo lišit, zapisujeme ji do uvozovek.

(Petroutsos, 1999)

POZOR! Dle příslušného pravidlem pro zápis promenných je ten, že jméno nesmí začínat číslicí ani nesmí obsahovat mezery, čárky a znaky jako je @, #, &, ~, ! a %. Navíc mohou mít délku jen do 255 znaků (Klement, 2002).

Zvláštnosti při zápisu promenných:

- Celá čísla může zapsat v hexadické soustavě - například místo 1000 lze napsat &H3E8
- Desetinná čísla (např. 1.57) může zapsat v semilogaritmickém tvaru (157E-2)
- Logické funkce *False* a *True* může nahradit jejich interními číselnými obnohami. *False* je 0, *True* je -1.
- Zápis komentářů je pomocí uvozovek (šš). Pro zápis poznámky se používá apostrof ('). Poznámka slouží pouze pro povysvětlení kódu. Program ji nevykonává.

(Klement, 2002)

5.6.1 Deklarování promenných

Promenné deklarujeme před jejich použitím. Pro deklaraci používáme slovo *Dim*, za kterým následuje název promenné a její typ. Ukázka deklarace může vypadat například takto:

Dim Součet As Integer

Slovo *As* je klíčovým slovem a přichází vždy po jménu promenné, po něm následuje typ deklarace.

(Petroutsos, 1999)

Jazyk VB používá tyto datové typy (viz obrázek . 19)

Název typu	Délka v bytech	Popis	Hodnoty
<i>Byte (#)</i>	1	Celé číslo bez znaménka	0...255
<i>Boolean</i>	2	Logické hodnoty	False, True
<i>Integer (%)</i>	2	Celé číslo	-32768...32767
<i>Long (&)</i>	4	Celé číslo (větší rozsah)	-1,04 ³⁸ ...1,08 ⁴²
<i>Single</i>	4	Číslo v pohyblivé desetinné čárce	
<i>Double</i>	8	Číslo v pohyblivé desetinné čárce (větší rozsah)	
<i>String (\$)</i>	1-65400	Řetězec znaků	
<i>Date</i>	8	Datum a čas	
<i>Object</i>	8	Univerzální typ (nemusí se definovat)	

Obrázek 19: Datové typy (Klement, 2002)

Ukázka:

Dim po et As Integer, Cena As Integer

*Dim d As String, p, Dk As String*15*

Pokud není u promenné uveden typ (název promenné *p*, viz ukázka), má typ Object. Promenná *Dk* je, ve výše uvedeném příkladu, etzec s pevnou délkou znaků a to 15. Promenná *d* je pouze etzec promenné délky. Slovo *Dim* se píše pouze na začátku řádku.

(Klement, 2002)

VB umožnil také deklarovat řádky zná pole, která jsou určena horní a dolní mezí. Tyto meze jsou od sebe odděleny pomocí slova *To* (Klement, 2002).

Pro ukázkou si uvedeme příklad deklarace jednorozměrného pole *číslo* a trojrozměrného pole *Matice*:

Dim číslo (15) As Long, Matice (1 To 5, 1 To 10, 1 To 15) As Single

Jak je z výše uvedené deklarace zřejmé, pro zápis pole se používají kulaté závorky. Pokud je uvedena pouze jedna mez (viz ukázka deklarace promenné *číslo*), je automaticky hodnota dolní meze 0. Takovou deklaraci můžeme přepsat:

Dim číslo (0 To 15) As Long

S proměnnými se můžeme ve vývojovém prostředí programu setkat na různých místech. Jejich umístění se odvíjí od jejich rozsahové platnosti. Tí úrovní platnosti proměnných jsou:

- *dostupná v celé aplikaci* - musí být deklarována v modulu Basic, místo *Dim* se používá *Global*
- *dostupná v celém formuláři i nebo modulu* - deklarace se uvádí v sekci *General* příslušného formuláře nebo modulu
- *lokální v proceduře i nebo funkci* - deklaraci je možné zapsat na libovolné místo uvnitř funkce. Její platnost je pak v rozsahu od místa deklarace až po konec funkce.

(Klement, 2002)

5.7 Operátory

Při psaní zdrojového kódu je leckdy potřeba vyjádřit určitý vztah proměnných, či zapsat vzorec pro výpočet funkce. K tomuto VB používá 4 kategorie operátorů, které Klement (2002) dělí na:

- *Aritmetické* - zde patří mocnina, opačná hodnota, násobení a dělení, zbytek, součet a rozdíl
- *Srovnávací* - méně, méně nebo rovno, větší, větší nebo rovno, je rovno, není rovno, je, srovnání se vzorem
- *Spojení* *et zc* - &, +.
- *Logické* - konjunkce, disjunkce, ekvivalence, implikace, negace, nonekvivalence.

5.8 Příkazy

Vytvoření aplikace je závislé na vkládání příkazů. VB používá dva druhy. Podle počtu potrubných znaků v příkazu je dle Klementa (2002) na:

1. Řádkové příkazy

Jejich rozsah je pouze na jednom řádku. Příkladem takových příkazů je například *GoTo* (příkaz skok), nebo řádková verze příkazu *If*. Je-li příkaz na řádek příliž dlouhý, lze odskočit na další řádek vložením znaku podtržení (_).

2. Blokované příkazy

Tyto příkazy jsou psány na více řádků. Patří sem například blokovaná verze příkazu *If*, příkazy cyklu (*For* a *Do*) a příčina (*Select Case*). Příkaz *If* musí být ukončen klíčovými slovy *End If*.

<pre>If v < > 0 Then v=1/v Else v=1</pre>	<pre>If v < > 0 Then v=1/v Else v=1 End If</pre>
---	--

Obrázek 20: Ukázka příkazu *If* v řádkové a blokované verzi (Klement, 2002)

Konec cyklu *For* je vyznačen klíčovým slovem *Next*. Konec cyklu *Do* je vyznačen klíčovým slovem *Loop*. Tento druhý cyklus je řízen podmínkou, která může být zapsána buď na začátku (u *Do*), nebo na konci (u *Loop*). Podmínka je dvojího druhu: *While* (cyklus probíhá, dokud podmínka platí), nebo *Until* (cyklus probíhá, dokud podmínka nezačne platit) (Klement, 2002).

V následujícím příkladu, který pro ilustraci uvádí Klement (2002), je zobrazen součet čísel 1 až 10. K výpočtu jsou použity různé druhy cyklu.

<pre>s=0 For i=1 To 10 s=s+i Next i</pre>	<pre>s=0: i=1 Do While i<=10 s=s+i i=i+1 Loop</pre>	<pre>s=0: i=1 Do s=s+i i=i+1 LoopWhile i<=10</pre>	<pre>s=0: i=1 DoUntil i>10 s=s+i i=i+1 Loop</pre>	<pre>s=0: i=1 Do s=s+i i=i+1 LoopUntil i>10</pre>
---	--	---	--	--

Obrázek 21: Výpočet součtu pomocí různých druhů cyklu

5.9 Procedury a funkce

VB používá dva typy programových jednotek:

- Procedury
- Funkce

Každý parametr je volán bu hodnotou (v hlavi ce funkce stojí p ed parametrem klí ové slovo *ByVal*) nebo odkazem (p ed parametrem nestojí klí ové slovo nebo je p ed ním *ByRef*).

(Klement, 2002)

1. Procedury

Procedury za ínají v kódu klí ovým slovem *Sub* a kon í slovem *End Sub*.

Klement (2002) ve své práci uvádí jednoduchý p íklad takové procedury. Jedná se o proceduru *Nastav*, která nastaví všechny prvky na zvolenou hodnotu.

```
Sub Nastav (Pole() As Integer, ByVal Hodnota As Integer)
```

```
    Dim i As Integer
```

```
    For i = LBound(Pole) To UBound(Pole)
```

```
        Pole (i) = Hodnota
```

```
    Next i
```

```
End Sub
```

2. Funkce

Funkce v kódu za ínají klí ovým slovem *Function* a kon í slovem *End Fuction*.

Klement (2002) ve své práci uvádí, jednoduchý p íklad takové funkce. Jedná se o funkci, která po ítá odmocninu ísla typu *Single*.

```
FunctionOdmocnicna (ByVal x As Single) As Single
```

```
    If < 0 Then Odmocnina = 0
```

```
    Odmonicna = Sgr(x) funkceSgr po ítá druhou odmocninu
```

```
End Function
```

5.10 Ošetření chyb při výpočtu

Při chodu aplikace může dojít k chybě. Pokud se tak stane, aplikace se zastaví. VB umožňuje tvůrci, aby si chyby sami ošetřili. V části programu, ve které tučíme chybu, vložíme příkaz *On Error*, a to buď ve tvaru *On Error GoTo ResumeNext* (příkaz s chybou se přeskočí) nebo ve tvaru *On Error GoTo* (při chybě skočí program na návští *Chyba*). Na konci sekvence je potřeba zapsat jeden ze tří možných zápisů *Resume*:

- *Resume 0* ó dojde k vrácení programu tam, kde došlo k chybě
- *ResumeNext* ó pokračování programu příkazem, který stojí za příkazem s chybou
- *ResumeNáv* ó pokračování programu na místě *Náv*

Příkaz *On Error* je třeba ukončit a to vložením zápisu *On Error GoTo 0*, který se vkládá na konec.

(Klement, 2002)

Klement (2002) uvádí příklad ošetření chyby funkce, která počítá výraz $a^2 + 1$. Program je postaven tak, aby při případné velké hodnotě neznámé a , a tedy při výrazném přetečení výrazu při výpočtu, vrátil funkci maximální hodnotu typu *Integer* (tj. 32767).

Function Výraz (ByVal a As Integer) As Integer

Dim v As Integer 'deklarace lokální proměnné v

On Error GoTo Chyba

$v = a * a + 1$

On Error GoTo 0 'ukončí působení příkazu *On Error GoTo Chyba*

Výraz = v

Exit Function 'návrat z funkce

Chyba: v = 32767

ResumeNext

End Function

PRAKTICKÁ ÁST

Praktická ást této diplomové práce je zam ena na popis výukového programu, který byl vytvo en v prostředí aplikace Microsoft Studio 2010 Express za pomoci jazyka VB. D vodem této volby jazyka byla snaha o vytvo ení uřivatelsky jednoduché a p ív tivé aplikace, která bude polyfunk ní, tzn. takové, která bude plnit více didaktických funkcí. Dalším cílem bude vytvo ení manuálu k výukovému programu, který bude p íložen jako p íloha na CD-ROM.

6 Výukový program Prostorová t lesa

Výukový program *Prostorová t lesa* je vytvo en na procvi ení a upevn ní u iva stereometrie. Je rozd len do t ech hlavní ástí, z nichř kařdá je vytvo ena tak, aby plnila rozdílné funkce.

První ást je zam ena na teorii nazvanou *Výukový text*. Zde jsem vytvo ila okna, která zobrazují jednak jednotlivá t lesa a jejich vzorce pro výpo et objemu a povrchu, a jednak praktické ukázky řt lesů v b říném řivot .

Druhá ást programu je zam ena za kontrolní neboli zp tnovazební funkci, která je nazvaná jako *Test*. Tato ást slouří jak pro řáký jako řáze procvi ovací a kontrolní, tak i pro u itele jako sbírka p íklad ů a správných výsledk ů.

T etí ást aplikace jsem vytvo ila za ú elem jakéhosi řodpo inutíř od procesu u ení. K této problematice jsem zvolila didaktickou hru zvanou *Pexeso*. Hra je konstruována tak, aby p edávala uřivateli nenásilným zp sobem pot ebné informace. Zna nou výhodou je, ře řák aniř by si to uv domoval, vstupuje do procesu u ení. Je to vlastn řakási obdoba testu, ve které jde o p íazení vzorc ů konkrétním t les m.

V-echny tyto ásti budou níře detailn ě popsány.

Vytvo ený výukový program a manuál pro jeho pouřítí jsou k dispozici jako p íloha na CD-ROM.

Nefl se ov-em za neme zabývat samotnou aplikací, pokládám za d leflité uvést technické pofladavky programu a zp sob jeho spu-t ní. Proto první dv kapitoly budou v novány této problematice.

6.1 Systémové pofladavky

Výukový program *Prostorová t lesa* je 32-bitová aplikace ur ená pro opera ní systém Microsoft Windows. Program je tudífl spustitelný na kterémkoliv po íta i, který má opera ní systém typu Windows XP, Windows Vista, Windows 7 a Windows 8.

Chod aplikace vyfladuje rozhraní .NET Framework 4.0., tudífl bez jeho nainstalování fungovat nebude.

6.2 Spu-t ní a ukon ení výukového programu

Ke spu-t ní aplikace není pot eba instala ního programu. Výukový program je typu *.exe*, tudífl k jeho aktivaci sta í dvojí kliknutí levým tla ítkem my-í na p íslu-nou ikonu. Jediné co po íta ke spu-t ní vyfladuje, je rozhraní .NET Framework 4.0, takže pokud jífl v po íta i není, je t eba ho nainstalovat. Instala ní program je voln dostupný na internetu.

Pro ukon ení programu slouffí červený k ífek nacházející se v pravé horní ásti okna. Krom oddílu s názvem *Výukový text*, kde tento k ífek slouffí k navrácení do hlavního okna, m fle být aplikace tímto zp sobem ukon ena v jakékoliv její ásti.

6.3 Popis výukového programu

Tato kapitola, *Prostorová t lesa*, detailn popisuje prost edí výukového programu. Po jejím nastudování, by m l být kaflký uflivatel schopen program pln vyuflívat. Následný text tudífl m fle slouffit i jako manuál k aplikaci. Ten bude je-t zvlá- p ilofen na CD-ROM.

6.3.1 Hlavní okno

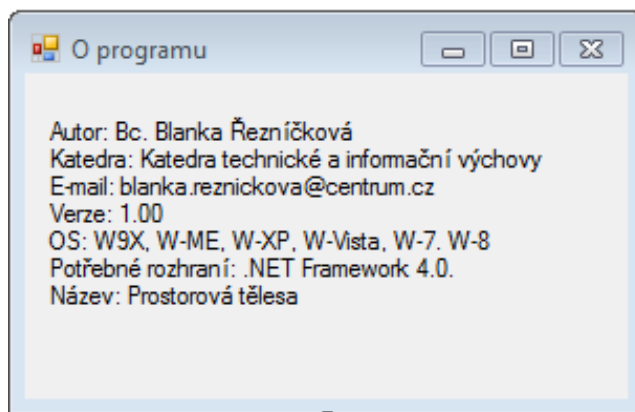
Po spuštění programu se na obrazovce objeví okno, které je na obrázku . 22. Zde jsou vidět tři oddíly zvané *Výukový text*, *Test* a *Pexeso*. Každý oddíl představuje jedno tlačítko, které spustí požadovanou část programu.

Pro vypnutí aplikace slouží jednak záhlaví okna, které je společné pro všechny aplikace firmy Microsoft. Toto záhlaví se nachází v pravém horním rohu a obsahuje tlačítka pro minimalizaci, maximalizaci, obnovení a ukončení okna. K ukončení aplikace tedy dojde po stisknutí tlačítka v pravém horním rohu. Další variantou je stisknutí tlačítka v pravém dolním rohu, které má podobu rozřezaného křížku.



Obrázek 22: Hlavní okno výukového programu Prostorová tělesa

Dále se na úvodním okně nachází pás karet, který nabízí záložku nazvanou *O programu*. Ta obsahuje základní informace o aplikaci. Dleřitým odkem je kontakt na programátora, který uživatel programu ocení, pokud dojde v aplikaci k chybě. Tato záložka je zobrazena na obrázku . 23.



Obrázek 23: Zálófkka O programu

6.3.2 Výukový text

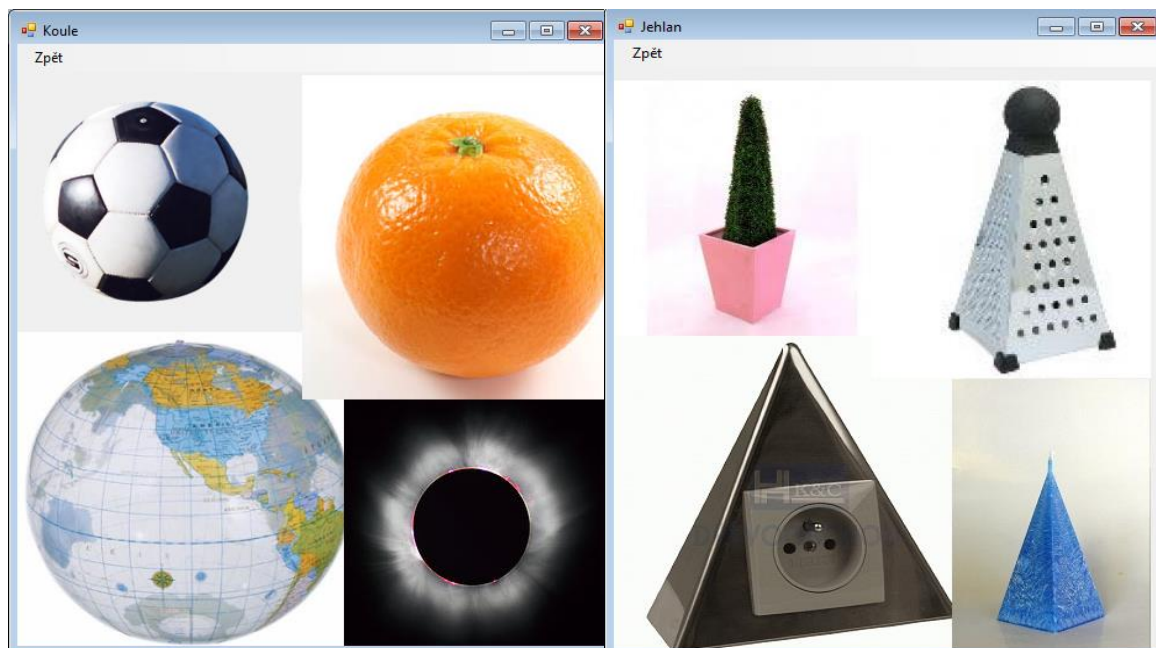
Po kliknutí na tlačítko *Výukový text* se uživateli zobrazí okno, které je vidět na obrázku . 24. V levém horním rohu je zobrazeno, v které části programu se zrovna uživatel nachází.

Okno je děleno do pomyslných dvou sloupců. V levém sloupci nalezneme tlačítka pro zobrazení těles v reálném světě (obrázky . 25 a 26). V pravé části se nacházejí tlačítka k zobrazení pořadované teorie jednotlivých těles (obrázek . 27). Jednotlivé okna se ukoní po kliknutí na tlačítko *Zpět*, které je umístěno v levém horním rohu, v páse karet.

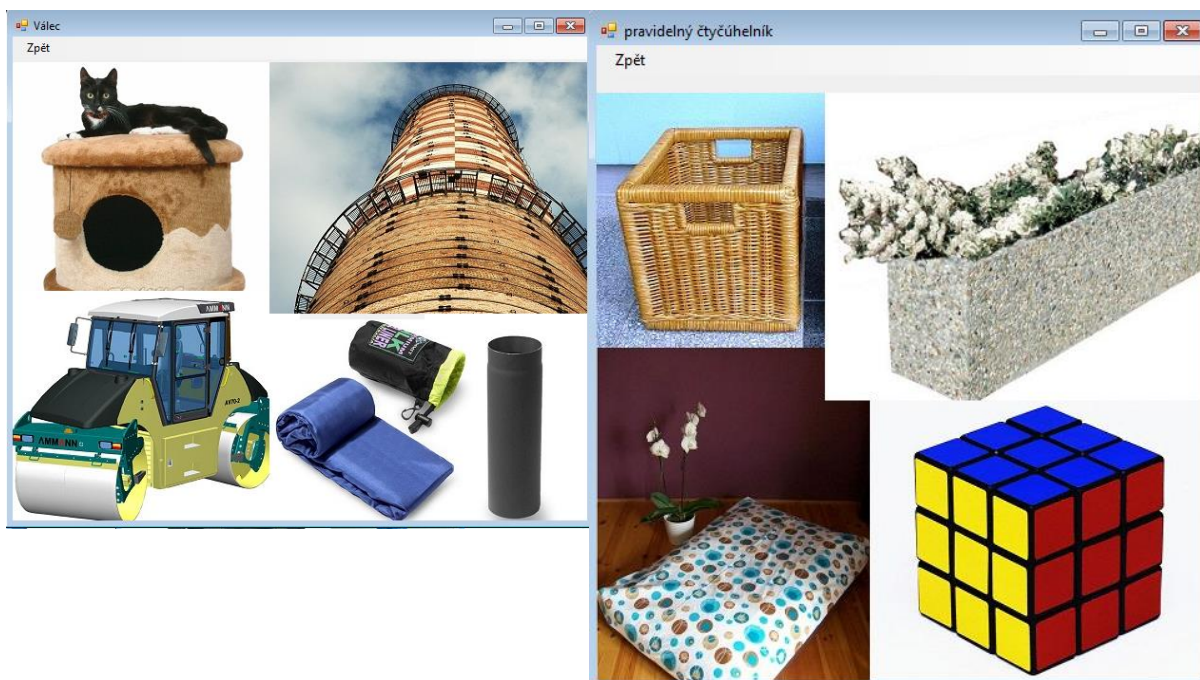
K vrácení z *Výukového textu* na úvodní okno stačí, když uživatel použije klikáček umístěný pro změnu v pravém horním rohu.



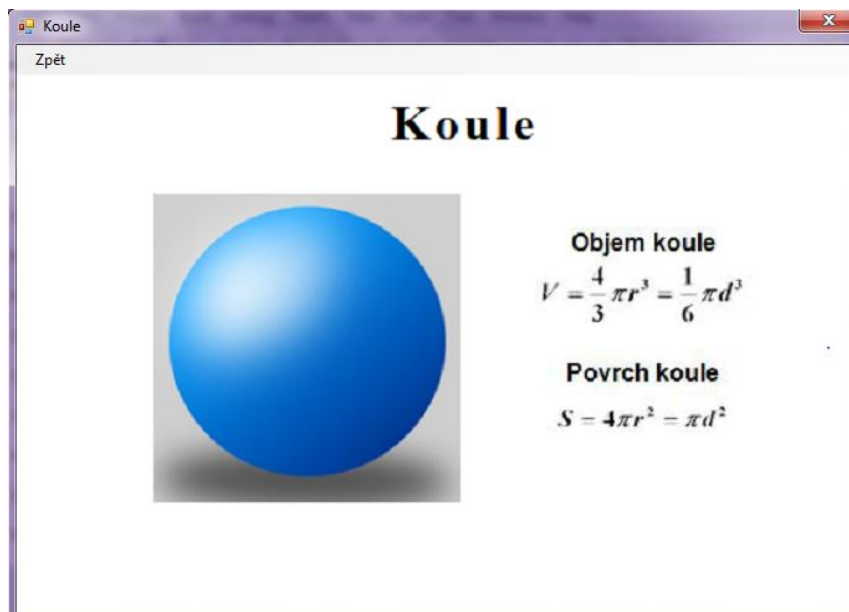
Obrázek 24: Okno výukového textu



Obrázek 25: Ukázky jehlanu a koule



Obrázek 26: Ukázky válce a pravidelného čtyřúhelníku

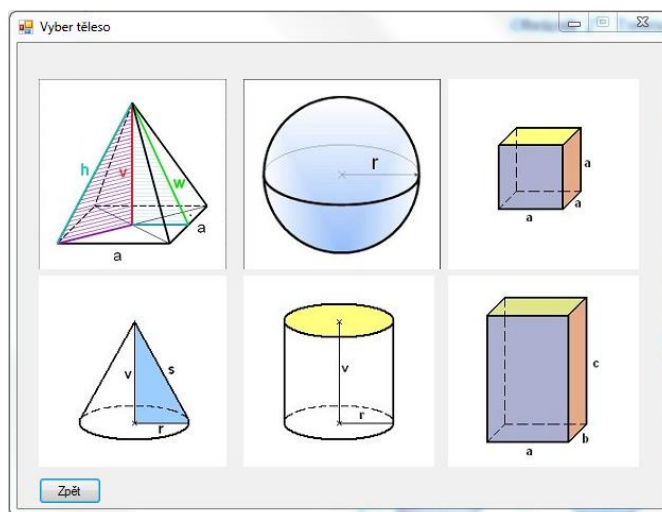


Obrázek 27: Teorie koule

Ostatní teorie těles je zobrazena v programu stejným způsobem.

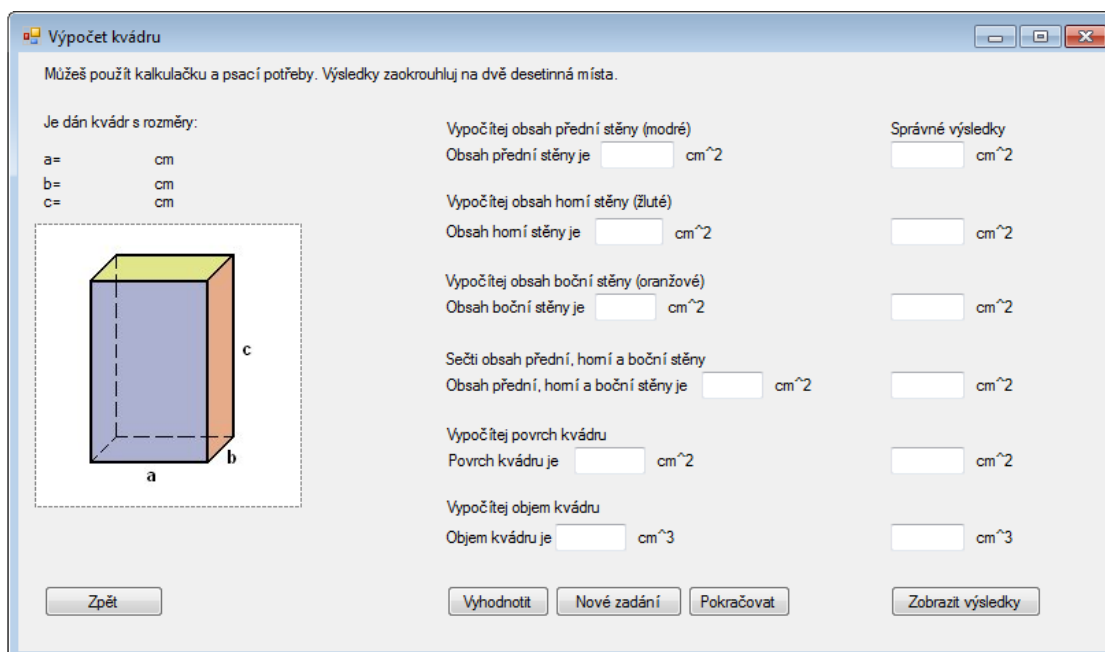
6.3.3 Test

Volbou tlačítka s názvem *Test*, je uživateli zobrazeno okno totožné s obrázkem 28. Toto okno nese název *Vyber těleso* a dává tak uživateli možnost volby testovacího okruhu. Tlačítko *Zpět* potom vrátí uživatele na hlavní okno.



Obrázek 28: Vyber těleso

Kliknutím na příslušný obrázek (např. kvádr) se uživatel dostane do testovacího prostředí (výpočet kvádrů, obrázek číslo 29).



Obrázek 29: Výpočet kvádrů

V levé horní části je uvedena informace o názvu zobrazené části. Dále jsou zde viditelné pokyny pro uživatele programu.

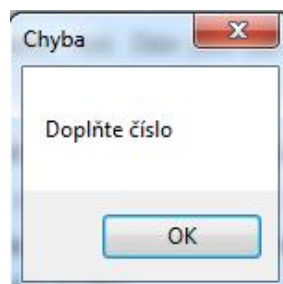
Po kliknutí na tlačítko *Nové zadání* se ke stranám kvádrů připíou náhodná celá čísla. Ty program vybírá v rozmezí od 1 do 9 včetně. Nízká čísla intervalu jsou zvolena z důvodu zapisování malých výsledků do textových polí. Uživatel, který umí pracovat s programem Microsoft Visual Basic 2010 Express může tento interval dle svých potřeb upravit.

Úkolem uživatele je zapsat do připravených textových polí výsledky odpovídající konkrétním rozměrům kvádrů. Tlačítko *Vyhodnotit* potom výsledky zkontroluje. Pokud uživatel zapsal správný výsledek, program toto číslo zbarví do zelena. Pokud je ovšem výsledek chybný, číslo zmizí a zůstane červená 0. Konkrétní ukázka je zobrazena na obrázku číslo 30.



Obrázek 30: Ukázka výpočtu konkrétního kvádrů

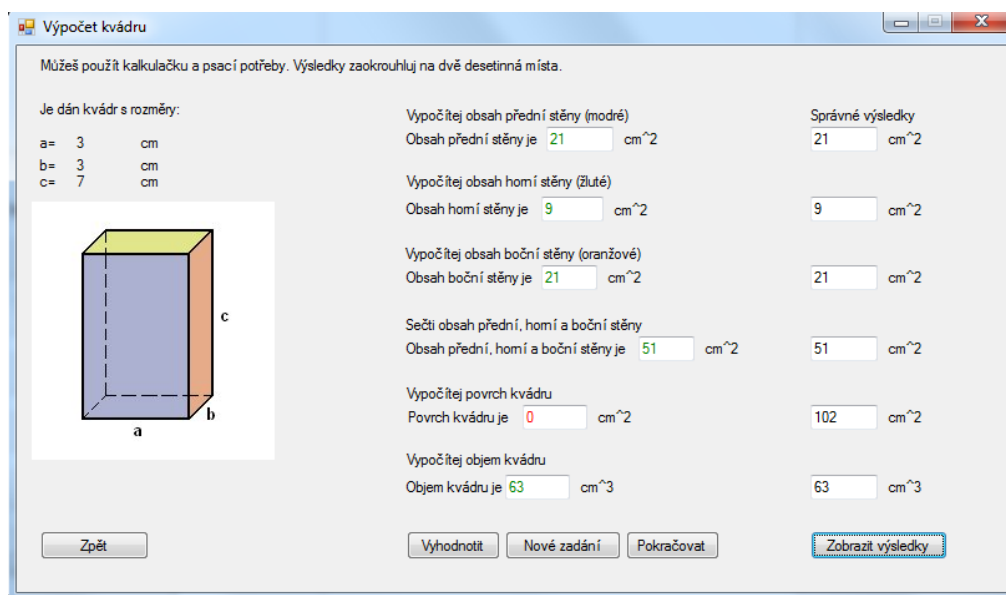
Program je navíc ošetřen proti zápisu jiných znaků než číslic (písmena, diakritika, í). Dále aplikace kontroluje, zda je v každém textovém poli zapsáno nějaké číslo. Pokud tyto podmínky splněné nejsou, vyskočí chybové hlášení zobrazené na obrázku číslo 31.



Obrázek 31: Chybové hlášení

V případě, že se uživatelé nebudou moci doplnit správného čísla, nabízí aplikace funkci *Zobrazit výsledky*. Po kliknutí na toto tlačítko se do připravených textových polí vypíší správné odpovědi. Bohužel se dá zde polemizovat o tom, jestli je tato část pro fláky vhodná. Je zřejmé, že mohou nastat dvě situace. První se bude týkat případu, kdy se opravdu bude flák snažit přijít ke správnému výsledku, ale bohužel se mu to opakovaně nebude dařit. Zde je potom zobrazení výsledku na místě, protože program tímto zajistí požadovanou

kontrolu a zpětnou vazbu. Ve druhém případě ale může nastat situace, kdy si fláči zobrazí výsledky za účelem pouhého šopsání výsledků do domácí úlohy. Bohužel výběr varianty je pouze na uživateli a tento konkrétní program ji nedokáže sám ošetřit. Je to navíc i z důvodu, že je zobrazení výsledků vytvořeno i pro uživatele jako ulehčení práce při opravování úloh. Takže by nemělo význam programu navštívit, aby výsledky zobrazil ať už po určitém počtu opakovaných vyhodnocení. Konkrétní zobrazení výsledku pro výše vymodelovaný příklad výpočtu kvádrů je znázorněno na obrázku 32.



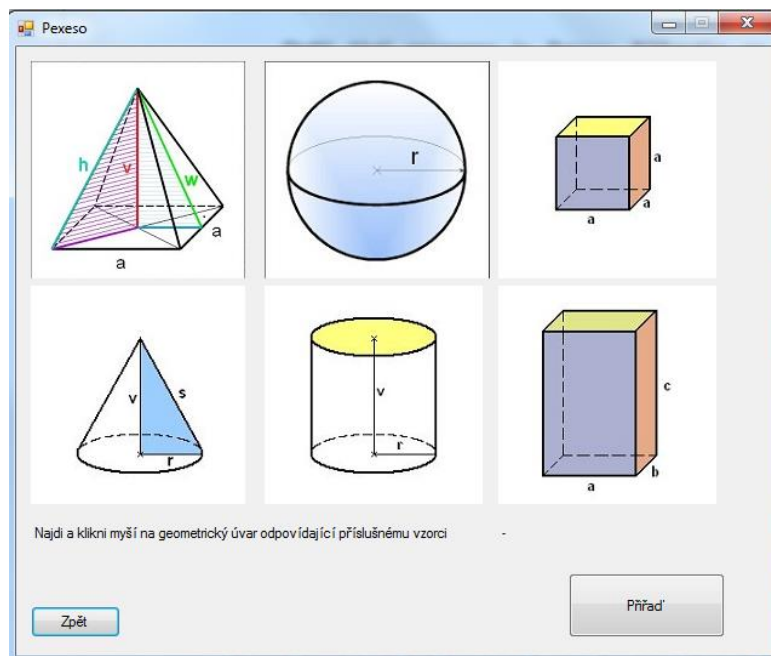
Obrázek 32: Výpočet kvádrů se zobrazenými výsledky

V okně pro výpočet tles se nacházejí ještě další tlačítka, a to: *Zpět* a *Pokračovat*. Tlačítko *Zpět* zavede uživatele zpátky na okno *Vyber tles*. Naopak tlačítko *Pokračovat* vrátí uživatele na hlavní okno, kde si může vybrat jiný oddíl aplikace.

Všechny ukázky výpočtu tles z důvodu velkého počtu stran této diplomové práce dále již uvádět nebudu. Fungují však všechna na stejném principu a uživatelské prostředí jednotlivých oken je konstruováno stejným způsobem.

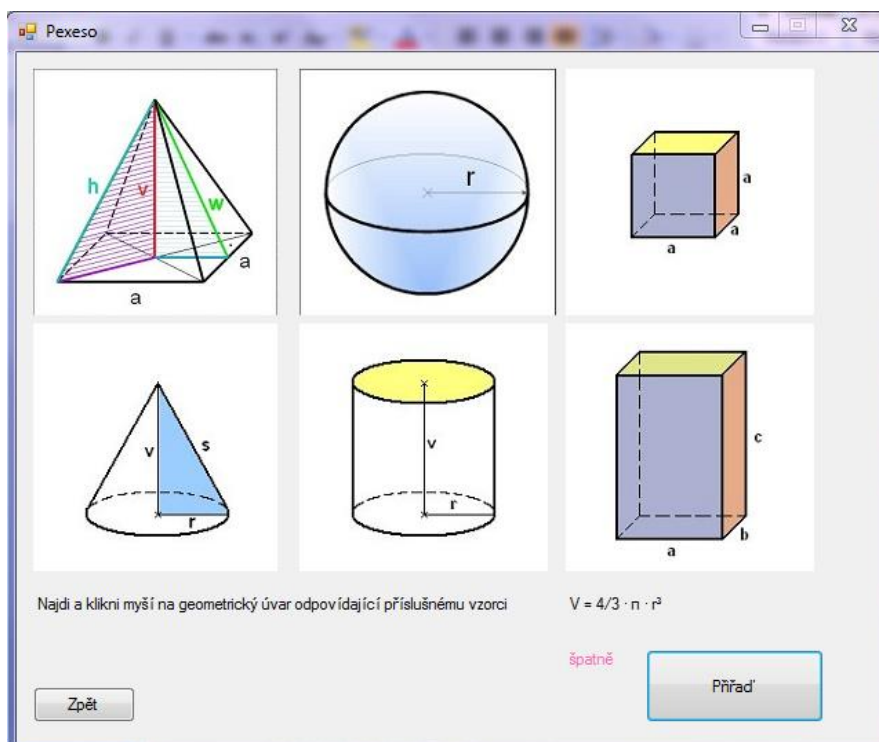
6.3.4 Pexeso

Další částí programu je *Pexeso*. Kliknutím na příslušné tlačítko umístěné na hlavním okně se uživateli zobrazí okno totožné s obrázkem číslo 33.

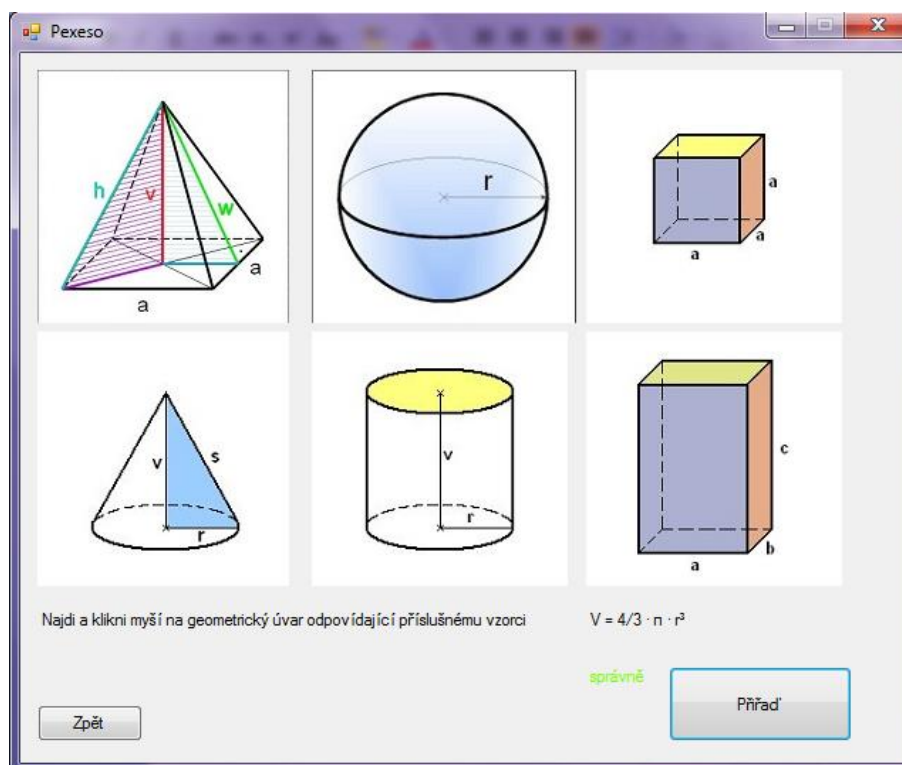


Obrázek 33: Pexeso

Obrázek znázorňuje 6 různých těles. Po kliknutí na tlačítko *Přihodit*, se vedle textu s instrukcemi objeví vzorec pro výpočet objemu i povrchu určitého tělesa. Úkolem uživatele je po zobrazení vzorce kliknout na odpovídající obrázek. Pokud vybraný obrázek tělesa odpovídá zobrazenému vzorci, objeví se v okně zelené slovo *správně*, v opačném případě program napíše slovo *nesprávně*, které bude mít červenou barvu. Tyto varianty znázorňuje obrázek číslo 34 a 35. Správný obrázek lze hledat tak dlouho, dokud uživatel znovu neklikne na tlačítko *Přihodit*, čímž zadá programu instrukci pro zobrazení dalšího vzorce.



Obrázek 34: Pexeso se špatným výb rem obrázku



Obrázek 35: Pexeso se správným výb rem obrázku

Vzorci jsou programem vybírány náhodně. K jejich zobrazení je potřeba vždy kliknout na tlačítko *Přít*. Aplikace není omezena pouze na řešení příkladů, ale může být spuštěna i bez zadání vzorců. Během řešení stále dokola, takže je jen na uživateli, kdy hru ukončí.

Dalším tlačítkem, které zde program nabízí je *Zpět*. Jeho zvolením vrátí program uživatele na hlavní okno.

6.4 Didaktický přínos a výhody programu

Didaktický přínos vytvořeného výukového programu spoívá v tom, že lze program využít jak při zpestření hodin matematiky, tak i při domácím používání, kdy si žáci sami procvičí a doplní potřebné znalosti. Program pojímá uivo stereometrie komplexně a hlavně poskytuje žákovi okamžitou zpětnou vazbu. Díky zajímavé grafické úpravě je aplikace pro žáky poutavá. Zábavným způsobem tak předává uživateli potřebné informace a žáci jsou pak motivováni k práci s programem. Přínosem je tedy procvičení nespočetného množství příkladů a tudíž upevnění potřebných znalostí.

Výhoda výukového programu *Prostorová tělesa* je, že jde o nenáročnou aplikaci spustitelnou na běžném počítači s operačním systémem Windows (verze 95 a výše) od firmy Microsoft. Navíc další nezanedbatelnou výhodou je, že program v testovací části nabízí uživateli náhodně generované příklady. Tato funkce vede k odbourání stereotypu vepisování opakovaně stejných výsledků úloh.

Závěr

Diplomová práce se zabývala vytvořením komplexní aplikace, která by měla formu výukového programu. Cílem bylo vytvořit program, který bude zahrnovat jak část týkající se potěbné teorie, tak i části potěbné k testování osvojeného učiva. Tohoto cíle bylo dosaženo. S pomocí programového jazyka Visual Basic vznikl program s názvem *Prostorová t lesa*, který zahrnuje komplexní problematiku stereometrie. Tento program je vytvořen pro použití v hodině matematiky, kdy může učitel zvýšit efektivitu vyučovacího procesu pomocí jeho záležitosti. Dále pak může sloužit i žákům k domácímu procvičení a upevnění učiva.

Vytvořená aplikace je rozdělena na několik částí. První zahrnuje teoretickou oblast, která je doplněna praktickými ukázkami. Druhý okruh zahrnuje testovací část, kde jsou uživatelům nabídnuty okruhy testování. Tím je myšleno, že uživatel může zobrazit test jen pro ty tělesa, které si zvolí. Program tedy nevyžaduje otestování všech okruhů. Tato oblast je v nověna didaktické hře zvané *Pexeso*. Ta slouží k předání potěbných informací šňenásilným způsobem. Jedná se vlastně o motivací část programu. Zmíněný program je popsán v praktické části této práce.

Teoretická část diplomové práce splňuje druhý cíl, který jsem si v úvodu stanovila. Tím je sepsání potěbných teoretických poznatků o výukovém softwaru. Jsou zde taktéž zmíněny kapitoly pojednávající o záležitosti stereometrie do RVP ZV. Dále se zde vyskytuje kapitola v nověna programovacímu jazyku Visual Basic a hlavně popsání uživatelského prostředí aplikace MS Visual Studio 2010 Express, ve které je výukový program *Prostorová t lesa* vytvořen.

Posledním cílem, bylo sepsání uživatelské dokumentace. Ta je nezbytnou součástí každého kvalitního výukového programu. Slouží jako návod pro ty uživatele, kteří program nevytvářejí, ale kteří by v něm rádi pracovali. Tohoto cíle bylo rovněž dosaženo a vytvořený manuál je přiložen jako příloha na CD-ROM.

Ověření programu *Prostorová t lesa* proběhne ať již v mé učitelské profesi. Rovněž mám v úmyslu program rozšířit o další úlohy. Konkrétně mám na mysli příklady, kde jsou již zadány povrchy i objemy těles a úkolem žáka je vypočítat stranu, výšku, průměr atd. Posléze

bych chtěla program upravit tak, aby kantor mohl proměnné v testovací úloze ručně vpisovat do zadání příkladu.

V budoucnosti se plánuji dále zabývat vytvářením výukových programů pomocí programovacího jazyka Visual Basic. Mým cílem je vytvořit aplikace pro různé oblasti matematiky. Je zřejmé, že se bude jednat sice o časově náročnější práci, ale dle mého názoru tato snaha přinese do budoucna efektivitu vyučovacího procesu. Zapojením výukových aplikací do hodin matematiky i do jiných humanitních předmětů přispívá k názornosti a ke zvýšení motivace a aktivity žáků.

Zdroje

1. Axiomy stereometrie. *Stereometrie.webz* [online]. [cit. 2014-04-16]. Dostupné z: <http://stereometrie.webz.cz/axiomy.html>
2. BURIANOVÁ, Eva. *Matematický a výukový software*. Vyd. 1. Ostrava: Ostravská univerzita, 2003, 74 s. Systém celoživotního vzdělávání Moravskoslezska. ISBN 80-704-2867-8.
3. BURIANOVÁ, Eva. *Využití aplikačních programů ve výuce*. Vyd. 1. Ostrava: Ostravská univerzita, 2003, 146 s. Systém celoživotního vzdělávání Moravskoslezska. ISBN 80-704-2858-9.
4. CIHLÁR Jiří, ZELENKA Milan, NOCAR David. *Využití informačních a komunikačních technologií ve vyučování matematice na 2. stupni ZŠ*. In *Podíl užití matematiky ZŠ na tvorbu TWP : Studijní materiály k projektu*. 1. vyd. Praha: JCMF, 2006. 53 s. ISBN 80-7015-085-8.
5. ERNOCHOVÁ, Miroslava a Jaroslav NOVÁK. *Využití počítačové a písmo v vyučování: náměty pro práci dříve s počítačem*. Vyd. 1. Praha: Portál, 1998, 165 s. ISBN 80-717-8272-6.
6. DOSTÁL, Jiří. *Počítač ve vzdělávání*. Vyd. 1. Olomouc: Votobia Olomouc, 2007, 2 sv. ISBN 978-80-7220-295-912.
7. DOSTÁL, Jiří. *Výukový software a počítačové hry - nástroje moderního vzdělávání*. *Journal of Technology and Information Education*. [on-line.] 2009, ročník 1, číslo 1, s. 24 - 28. [cit. 2014-02-18] Dostupné z http://www.jtie.upol.cz/clanky_1_2009/vyukovy_software_a_didakticke_pocitacove_hry_-_nastroje_moderniho_vzdelavani.pdf. ISSN 1803-6805
8. ICT - význam zkratky. *IT Slovník.cz* [online]. [cit. 2014-04-16]. Dostupné z: <http://it-slovník.cz/pojem/ict>
9. Jak je to s verzemi jazyka Visual Basic. *Dotnetportal.cz* [online]. [cit. 2014-04-16]. Dostupné z: <http://www.dotnetportal.cz/blogy/3/Tomas-Herceg/586/Jak-je-to-s-verzemi-jazyka-Visual-Basic>

10. KLEMENT, Milan. *Základy programování v jazyce Visual Basic*. 1. vyd. Olomouc: Univerzita Palackého, Pedagogická fakulta, 2002. ISBN 80-244-0472-9.
11. KROPÁČEK, Jiří a Miroslav CHRÁSKA. *Výchova v obecně technických předmětech*. 1. vyd. Olomouc: Univerzita Palackého v Olomouci, 2004, 190 s. ISBN 80-244-0897-X.
12. MAZÁK, E. *Podívejme se na výukové programy a metodika jejich tvorby*. 1. vyd. Praha: Ústav národních informací, 1988. 119 s.
13. NOCAR, D. a kol. *E-learning v distančním vzdělávání*. Olomouc: UP, 2004. ISBN 80-244-0802-3.
14. SLAVÍK, Jan a Jaroslav NOVÁK. *Podívejme se na učební pomůcky u počítače: efektivní práce s informacemi ve škole*. Vyd. 1. Praha: Portál, 1997, 119 s. ISBN 80-717-8149
15. PETROUTSOS, Evangelos. *Visual Basic 6: pro vodce programátora*. Vyd. 1. Praha: Grada, 1999, 638 s. ISBN 80-716-9801-6.
16. Pojem intencionalita. *ABZ.cz: slovník cizích slov* [online]. [cit. 2014-04-16]. Dostupné z: <http://slovník-cizich-slov.abz.cz/web.php/slovo/intencionalita>
17. PRŮCHA, Jan, Eliška WALTEROVÁ a Jiří MAREŠ. *Pedagogický slovník*. 4., aktualiz. vyd. Praha: Portál, 2003. ISBN 80-717-8772-8
18. *Rámcový vzdělávací program pro základní vzdělávání* [online]. Praha: Ministerstvo školství, mládeže a tělovýchovy, 2014 [cit. 2014-02-18]. Dostupné z: <http://www.msmt.cz/vzdelavani/zakladni-vzdelavani/upraveny-ramcovy-vzdelavaci-program-pro-zakladni-vzdelavani>
19. Úvod do stereometrie. *Matematika pro každého* [online]. [cit. 2014-04-16]. Dostupné z: <http://maths.cz/clanky/uvod-do-stereometrie.html>
20. Visual Basic. *Webopedia* [online]. [cit. 2014-04-16]. Dostupné z: http://www.webopedia.com/TERM/V/Visual_Basic.html
21. Základy teorie učení. *SMEP: systém multimediální elektronické publikace* [online]. [cit. 2014-04-16]. Dostupné z: http://etext.czu.cz/php/skripta/kapitola.php?titul_key=64&idkapitola=95
22. ZOUNEK, Jiří. *E-learning - jedna z podob učení v moderní společnosti*. Vyd. 1. Brno: Masarykova univerzita, 2009. ISBN 978-80-210-5123-2.

23. .NET framework. *Katedra informatiky FEI V^TB-TUO* [online]. [cit. 2014-04-16].
Dostupné z: <http://www.cs.vsb.cz/behalek/vyuka/pcsharp/text/ch01.html>

Seznam obrázk

Obrázek 1: Vyuffití po íta e ve vzd lávání.....	11
Obrázek 2: Okno MS Visual Studio 2010 Express p í grafické tvorb programu.....	45
Obrázek 3: Okno MS Visual Studio 2010 Express p í psaní p íkaz	45
Obrázek 4: Popis oken v MS Visual Studio (Klement, 2002).....	46
Obrázek 5: Nabídka FILE	47
Obrázek 6: Popis nabídky FILE (Klement, 2002)	48
Obrázek 7: Nabídka EDIT.....	49
Obrázek 8: Popis nabídky EDIT (Klement, 2002).....	50
Obrázek 9: Nabídka VIEW (Zobrazit).....	50
Obrázek 10: Popis nabídky VIEW (Klement, 2002)	51
Obrázek 11: Nabídka PROJECT	52
Obrázek 12: Popis nabídky PROJECT (Klement, 2002)	53
Obrázek 13: Nabídka DEBUG	54
Obrázek 14: Popis nabídky DEBUG (Klement, 2002)	54
Obrázek 15: Nabídka WINDOW.....	55
Obrázek 16: Popis nabídky WINDOW (Klement, 2002).....	55
Obrázek 17: Nabídka HELP.....	56
Obrázek 18: Popis nabídky HELP (Klement, 2002).....	56
Obrázek 19: Datové typy (Klement, 2002)	58
Obrázek 20: Ukázka p íkazu <i>If</i> v ádkové a blokové verzi (Klement, 2002).....	60
Obrázek 21: Výpo et sou tu pomocí r zných druh cyklu	60
Obrázek 22: Hlavní okno výukového programu Prostorová t lesa	65
Obrázek 23: Zálolka O programu.....	66
Obrázek 24: Okno výukového textu	66
Obrázek 25: Ukázky jehlanu a koule	67
Obrázek 26: Ukázky válce a pravidelného ty úhelníku.....	67
Obrázek 27: Teorie koule.....	68
Obrázek 28: Vyber t leso.....	68
Obrázek 29: Výpo et kvádru.....	69

Obrázek 30: Ukázka výpočtu konkrétního kvádru	70
Obrázek 31: Chybové hlášení.....	70
Obrázek 32: Výpočet kvádru se zobrazenými výsledky	71
Obrázek 33: Pexeso	72
Obrázek 34: Pexeso se špatným výběrem obrázku.....	73
Obrázek 35: Pexeso se správným výběrem obrázku.....	73

Seznam příloh

Příloha . 1: CD-ROM

Příloha . 2: Vzdělávací oblast Matematika a její aplikace a osnovy (převzato z ^{TMP} základní školy Heyrovského v Olomouci)

Příloha .1 o CD- ROM

Na přiloženém CD-ROM se nacházejí následující soubory:

Diplomová práce

- Diplomova_prace.pdf

Výukový software

- Prostorova_telesa.exe

Manuál k programu

- Manual_k_programu.pdf

Příloha . 2: Vzdělávací oblast Matematika a její aplikace – osnovy (převzato z TMVP základní školy Heyrovského v Olomouci)

Vzdělávací oblast Matematika a její aplikace

Vyučovací předmět: **Matematika**

Ročník: **6.**

Očekávané výstupy	Dílčí výstupy	Učivo	Průřezová témata, mezipředmětové vztahy
- provádí písemné a početní operace v oboru přirozených čísel	- čte, zapisuje, porovnává, zaokrouhluje - provádí zpaměti i písemně početní operace - využívá matematických zákonů, určuje správné pořadí výpočtů - provádí odhady a kontrolu výpočtů - řeší jednoduché slovní úlohy z praktického života vedoucí na 1-2 početní úkony	Rozšířené opakování o přirozená čísla. - četění a zápis v desítkové soustavě - zobrazení na číselné ose - porovnávání, zaokrouhlování - početní operace, užití matematických zákonů, pořadí výpočtů - slovní úlohy	
- narýsuje a znázorní základní geometrické útvary (tvar, obdélník, trojúhelník a kružnici), užití jednoduché konstrukce	- užití a rozlišení pojem přímka, polopřímka, úsečka - rýsuje lineární útvary - provádí jednotky délky, hmotnosti a času	Rozšířené opakování o geometrické útvary v rovině. - rovina, bod, úsečka, přímka, polopřímka - kružnice, kruh	Fórmální délky

	<ul style="list-style-type: none"> - vypočítá obvod tverce, obdélníku a trojúhelníku - provádí jednotky obsahu - zdvořuje a využívá polohové a metrické vlastnosti základních rovinných útvarů při řešení úloh a jednoduchých praktických problémů - charakterizuje a určuje základní rovinné útvary 	<ul style="list-style-type: none"> - převody jednotek oboru přirozených čísel - obvody tverce, obdélníku, trojúhelníku - obsahy tverce, obdélníku - jednotky obsahu - obsahy složitějších obrazců 	
<ul style="list-style-type: none"> - určuje velikost úhlu měřením a výpočtem 	<ul style="list-style-type: none"> - rozumí pojmu úhel - narýsuje a změřený daný úhel - graficky představí úhel a sestrojí jeho osu - rozliší a pojmenuje druhy úhlů - provádí početní operace s velikostmi úhlů (ve stupních a minutách) - pozná dvojice úhlů vedlejších a vrcholových, využívá jejich vlastnosti - sestrojí úhly ($90^\circ, 60^\circ, 45^\circ, 30^\circ$) pomocí kružítka - rozumí pojmu mnohoúhelník - sestrojí pravidelný trojúhelník a osmiúhelník 	<p>Úhel a jeho velikost.</p> <ul style="list-style-type: none"> - pojem, rýsování a měření úhlu - osa úhlu - jednotky velikosti úhlu a měření velikosti úhlu - ostrý, tupý, pravý a přímý úhel - početní operace s velikostmi úhlů - vrcholové a vedlejší úhly - konstrukce úhlů pomocí kružítka - mnohoúhelníky o pojem, pravidelný trojúhelník a osmiúhelník (konstrukce, obvod) 	<p>EV 2 o Základní podmínky flivota na Zemi Z o ur ovaní zem pisné polohy, zemská osa a její sklon, otá ení Zem</p>

<ul style="list-style-type: none"> - provádí početní operace v oboru racionálních čísel - zaokrouhluje a provádí odhady s danou přesností, účelně využívá kalkulátor - analyzuje a řeší jednoduché problémy, modeluje konkrétní situace, uvažuje různé způsoby kvantitativního vyjádření vztahu celek a část desetinným číslem 	<ul style="list-style-type: none"> - čísla zapisuje desetinná čísla - zobrazí desetinné číslo na číselné ose - porovnává a zaokrouhluje desetinná čísla - provádí početní operace s desetinnými čísly - umí vypočítat aritmetický průměr - převádí jednotky - zaokrouhluje a provádí odhady s danou přesností - využije v praktických úlohách početní výkon - provádí odhady a kontrolu řešení úloh 	<p>Desetinná čísla.</p> <ul style="list-style-type: none"> - čtení a zápis v desítkové soustavě - zobrazení na číselné ose - porovnávání - zaokrouhlování - početní operace - aritmetický průměr - převody jednotek - slovní úlohy 	<p>VMEG 1 o Evropa a svět nás zajímá</p> <p>F o řešení početních úloh</p> <p>Z - rozdílnost mezi dvěma místy na Zemi, časová pásma, místní a světový čas</p>
<ul style="list-style-type: none"> - narýsuje a sestrojí obraz rovinného útvaru v osové souměrnosti - určí osově souměrný útvar 	<ul style="list-style-type: none"> - pozná shodné geometrické útvary - pozná útvary osově souměrné o ústředí ose souměrnosti - sestrojí obraz rovinného útvaru v osové souměrnosti 	<p>Osová souměrnost.</p> <ul style="list-style-type: none"> - shodnost geometrických útvarů - osová souměrnost - osa souměrnosti - osově souměrné útvary 	<p>EV 2 o Základní podmínky života na Zemi</p>
<ul style="list-style-type: none"> - modeluje a řeší situace s využitím dělitelnosti v oboru přirozených čísel, uvažuje logickou úvahu a kombinaci úsudků při řešení úloh a problémů a nalézá různé řešení předkládaných nebo zkoumaných situací 	<ul style="list-style-type: none"> - zná pojem násobek, dělitel - používá znaky dělitelnosti - rozumí pojmu prvo číslo, číslo složené - rozloží číslo na součin prvo čísel - určí a uvažuje násobky a dělitele v etně nejmenšího společného násobku a největšího společného 	<p>Dělitelnost přirozených čísel</p> <ul style="list-style-type: none"> - násobek, dělitel, znaky dělitelnosti - prvo číslo, číslo složené - společný násobek, dělitel 	

	<p>d litele</p> <ul style="list-style-type: none"> - modeluje a e-í situace s využitím d litelnosti p írozených ísel 		
<ul style="list-style-type: none"> - zd vod uje a využívá polohové a metrické vlastnosti trojúhelníku p í e-ení úloh a jednotlivých praktických problém , využívá pot ebnou matematickou symboliku 	<ul style="list-style-type: none"> - ur í a znázorní r zné druhy trojúhelník a zná jeho vlastnosti - pojmenuje, znázorní a správn užíává základní pojmy (strana, vý-ka, vnit ní a vn j-í úhly) - sestojí t fnice, st ední p í ky, vý-ky trojúhelníku - sestojí trojúhelníku kružnici opsanou a vepsanou 	<p>Trojúhelník</p> <ul style="list-style-type: none"> - pojem, rozd lení - vnit ní a vn j-í úhly trojúhelníku - vý-ky, t fnice, st ední p í ky trojúhelníku - kružnice opsaná, vepsaná 	<p>F ó t fli-t t lesa</p>
<ul style="list-style-type: none"> - ur uje a charakterizuje základní prostorové útvary (t lesa), analyzuje jejich vlastnosti - na rtne a sestojí obraz jednoduchých t les v rovin -na rtne a sestojí sí t základních t les -odhaduje a vypo ítá objem a povrch t les - analyzuje a e-í aplika ní geometrické úlohy s využitím osvojeného matematického aparátu 	<ul style="list-style-type: none"> - charakterizuje jednotlivá t lesa (kvádr, krychle) - na rtne a narýsuje sí a z ní t leso vymodeluje - na rtne a sestojí obraz krychle a kvádrů ve volném rovnob řném promítání - vypo ítá povrch krychle a kvádrů - užíává jednotky objemu a vzájemn je p evádí - odhaduje a vypo ítá objem krychle a kvádrů 	<p>Povrch a objem krychle a kvádrů.</p> <ul style="list-style-type: none"> - kvádr, krychle, zobrazování, sí t les - povrch krychle, kvádrů - jednotky objemu - objem krychle, kvádrů 	<p>EV 2 ó Základní podmínky flivota na Zemi</p>
		<p>Záv re né opakování.</p>	

Vzdělávací oblast: Matematika a její aplikace

Vyučovací předmět: **Matematika**

Ročník: 7.

Hlavní výstup	Dílčí výstupy	Učivo	Průřezová témata mezipředmětové vztahy.
<ul style="list-style-type: none">- provádí početní operace v oboru racionálních čísel- zaokrouhluje, provádí odhady s danou přesností, účelně využívá kalkulátor- určí velikost úhlu měřením i výpočtem	<ul style="list-style-type: none">- čte a zapisuje desetinná čísla- zobrazí desetinná čísla na číselné ose- porovnává a zaokrouhluje desetinná čísla- provádí početní operace s desetinnými čísly- uznává zákon komutativní a asociativní- převádí jednotky délky a hmotnosti- rýsuje úhel dané velikosti, uznává jednotky stupě a minuta- umí změnit velikost úhlu úhlovým měřítkem- dokáže odhadnout velikost úhlu	<p><u>Opakování ze 6. ročníku</u> <u>Desetinná čísla</u></p> <ul style="list-style-type: none">- čtení a zápis čísel v desítkové soustavě- zobrazení čísel na číselné ose- početní operace s desetinnými čísly- porovnávání a zaokrouhlování- aritmetický průměr- převody jednotek- slovní úlohy <p>-</p> <p><u>Úhel a jeho velikost</u></p> <ul style="list-style-type: none">- rýsování úhlu dané velikosti, osa úhlu- měření úhlu, jednotky velikosti úhlu- druhy úhlů	

<ul style="list-style-type: none"> - provádí početní operace v oboru racionálních čísel - uvažuje kvantitativního vyjádření vztahu celkové část zlomkem a desetinným číslem - analyzuje a řeší jednoduché problémy, modeluje konkrétní situace, v nichž využívá aparát racionálních čísel 	<ul style="list-style-type: none"> - modeluje a zapisuje zlomkem část celku - převádí zlomky na desetinná čísla a naopak - porovnává zlomky, zobrazuje je na číselné ose - provádí početní operace s racionálními čísly - převádí zlomky na základní tvar - dokáže upravit smíšené číslo na zlomek - uvažuje zlomky při řešení praktických situací - řeší slovní úlohy z praxe - upraví složený zlomek na jednoduchý 	<p><u>Racionální čísla kladná -zlomky</u></p> <ul style="list-style-type: none"> - tení a zápis zlomku - vztah mezi zlomky a desetinnými čísly - zobrazení na číselné ose - převrácený zlomek - smíšené číslo - početní operace - slovní úlohy - složený zlomek 	
<ul style="list-style-type: none"> - analyzuje a řeší jednoduché problémy, modeluje konkrétní situace, v nichž využívá matematický aparát v oboru celých i racionálních čísel 	<ul style="list-style-type: none"> - rozlišuje kladná a záporná čísla - zobrazí kladná a záporná čísla na vodorovné i svislé číselné ose - chápe pojem opačné číslo - určí absolutní hodnotu daného čísla a chápe její geometrický význam - ovládá početní operace s celými a racionálními čísly - vyřeší jednoduché slovní úlohy v oboru celých i racionálních čísel 	<p><u>Celá čísla. Racionální čísla.</u></p> <ul style="list-style-type: none"> - tení a zápis čísla - zobrazení na číselné ose - opačné číslo - absolutní hodnota - početní operace s celými čísly - početní operace s racionálními čísly - slovní úlohy 	<p>Fyzika 6 teplota na teploměru</p>

<ul style="list-style-type: none"> - uflívá k argumentaci a p i výpo tech v ty o shodnosti trojúhelník - na rtne a sestrojí rovinný útvar 	<ul style="list-style-type: none"> -uflívá v ty o shodnosti trojúhelník v po etních a konstruk ních úlohách -sestrojí trojúhelník podle r zného zadání daných prvk -dbá na kvalitu a p esnost rýsování 	<p><u>Trojúhelník</u></p> <ul style="list-style-type: none"> -opakování u íva o trojúhelníku -shodnost trojúhelník .trojúhelníková nerovnost -konstrukce trojúhelníku 	
<ul style="list-style-type: none"> - na rtne a sestrojí obraz rovinného útvaru ve st edové soum rnosti -ur í st edov soum rný útvar 	<ul style="list-style-type: none"> - pozná st edov soum rné útvary - najde st ed soum rnosti - sestrojí obraz rovinného útvaru ve st edové soum rnosti 	<p><u>St edová soum rnost</u></p> <ul style="list-style-type: none"> - st ed soum rnosti - st edov soum rné útvary 	
<ul style="list-style-type: none"> - charakterizuje a t ídí základní rovinné útvary - odhaduje a vypo ítá obvody a obsahy základních rovinných útvar - na rtne a sestrojí rovinné útvary 	<ul style="list-style-type: none"> - charakterizuje pojem rovnob fníku - rozli-uje r zné typy rovnob fníku, zná jejich vlastnosti - na rtne a sestrojít rovnob fník - odhaduje a vypo ítává obvod a obsah rovnob fníku - odhaduje a vypo ítává obsah trojúhelníku - dokáfle sestrojít vý-ky rovinných 	<p><u>Rovnob fníky</u></p> <ul style="list-style-type: none"> - pojem - vlastnosti - rozd lení - konstrukce - obvod a obsah rovnob fníku - obsah trojúhelníku - vý-ky rovinných útvar - slovní úlohy 	<p>Fyzika ó skládání sil</p>

	<p>útvary</p> <ul style="list-style-type: none"> - řeší slovní úlohy z praxe na obvody a obsahy rovnoběžníků - odhaduje, zaokrouhluje s danou přesností, - účelně využívá při výpočtech kalkulačtor 		
<ul style="list-style-type: none"> - charakterizuje a určuje základní rovinné útvary - namaluje a sestrojí rovinný útvar - odhaduje a vypočítá jeho obvod i obsah - analyzuje a řeší aplikativní geometrické úlohy s využitím osvojeného matematického aparátu 	<ul style="list-style-type: none"> - rozpozná a pojmenuje lichoběžník, dokážkou na rýtnout - sestrojí lichoběžník - vypočítá obvod a obsah lichoběžníku, správně určí jednotky výsledků - řeší slovní úlohy z praxe na obvod a obsah lichoběžníku 	<p><u>Lichoběžník</u></p> <ul style="list-style-type: none"> - pojem - konstrukce - obvod a obsah - slovní úlohy 	

<ul style="list-style-type: none"> - uflívá kvantitativní vyjádění vztahu celek a část poměrem - e-í modelováním i výpočtem situace vyjáděné poměrem, pracuje s měřítky map a plán - určuje vztah přímé nebo nepřímé úměrnosti - vyjádění funkčního vztahu tabulkou, rovnicí, grafem 	<ul style="list-style-type: none"> - vyjádění poměru mezi danými hodnotami - zvětšuje a zmenšuje veličiny v daném poměru - dílčí celek na části v daném poměru - zjednodušuje daný poměr krácením - chápe problematiku měřítek map a plán - rozumí a využívá pojmu úměrnosti - dokáže rozeznat rozdíl mezi přímou a nepřímou úměrností - dokáže sestavit tabulku a narysovat graf podle dané rovnice - dokáže zakreslit bod do pravouhlé soustavy souřadnic (vyčíst z grafu souřadnice bodu) - e-í slovní úlohy na přímou a nepřímou úměrnost - využívá trojúhelníku při řešení slovních úloh 	<p><u>Poměry. Přímá a nepřímá úměrnost</u></p> <ul style="list-style-type: none"> - pojem - zvětšení a zmenšení v daném poměru - rozdělení hodnoty v daném poměru - měřítko - úměrnosti - přímá a nepřímá úměrnost - trojúhelník - pravouhlá soustava souřadnic - slovní úlohy 	<p>OSV 1 - Rozvoj schopnosti poznání 5 - Kreativita</p> <p>Zeměpisové měřítko map, plán míst Přímá míchání surovin v daném poměru</p>
<ul style="list-style-type: none"> - uflívá kvantitativního vyjádění vztahu celek a část procentem - e-í aplikační úlohy na procenta i pro případ, že procentová část je větší než celek - zaokrouhluje a provádí odhady s danou přesností 	<ul style="list-style-type: none"> - chápe pojem 1 % - uflívá základní pojmy procentového počtu - určuje, jakou část celku tvoří daný počet procent - dokáže určit celek z dané části a daného počtu procent - vypočítá, kolik procent je daná 	<p><u>Procenta</u></p> <ul style="list-style-type: none"> - pojem - základ, procentová část, počet procent - slovní úlohy - promile - úrok 	<p>Zeměpisová ekonomika, hospodářství, křehké látky v ovzduší</p>

	<p>část z celku</p> <ul style="list-style-type: none"> - řeší slovní úlohy na procenta (i pro případ, že procentová část je v tučně neřešitelná) - chápe pojem promile - rozumí pojmu úrok - je schopen provádět výpočty s procenty pomocí kalkulačky 		
<ul style="list-style-type: none"> - určí a charakterizuje základní prostorové útvary tělesa, analyzuje jejich vlastnosti - odhaduje a vypočítá objem a povrch tělesa - narýsuje a sestrojí obraz jednoduchých těles v rovině - narýsuje a sestrojí síť základních těles 	<ul style="list-style-type: none"> - dokáže rozpoznat a pojmenovat různé typy hranolů - je schopen vyjmenovat jejich vlastnosti - narýsuje a sestrojí obrazy různých hranolů v rovině - dokáže si představit a narýsovat síť různých hranolů v rovině - odhaduje a vypočítá objem a povrch hranolu, správně určí jednotky výsledků 	<p><u>Povrch a objem hranolu</u></p> <ul style="list-style-type: none"> - opakování kvádru a krychle o objem a povrch - pojem hranol - povrch a objem hranolu 	
		<p><u>Závěrečné opakování</u></p>	

Vzdělávací oblast: Matematika a její aplikace

Vyučovací předmět: **Matematika**

Ročník: **8.**

Hlavní výstupy	Dílčí výstupy	Učivo	Průřezová témata mezipředmětové vztahy.
- provádí početní operace v oboru celých a racionálních čísel	<ul style="list-style-type: none">- ovládá početní výkony s racionálními čísly- převádí zlomek na desetinné číslo a opačně- zobrazuje zlomky na číselné ose- řeší slovní úlohy - ovládá výpočty s celými čísly- rozumí početním operacím s racionálními čísly- dokáže zobrazit celá a racionální čísla na číselné ose- řeší slovní úlohy s celými a racionálními čísly	<ul style="list-style-type: none">- <u>Opakování učiva ze 7. ročníku</u> <u>Racionální čísla</u><ul style="list-style-type: none">- početní operace- vztah mezi zlomky a desetinnými čísly- zobrazení na číselné ose <u>Celá čísla</u><ul style="list-style-type: none">- početní operace s celými čísly- početní operace s racionálními čísly- slovní úlohy	

<p>- provádí početní operace v oboru celých a racionálních čísel, užívá ve výpočtech druhou mocninu a odmocninu</p>	<p>- určí druhou mocninu a odmocninu výpočtem, pomocí tabulek, pomocí kalkulačky</p> <p>- pracuje s druhou mocninou a odmocninou ve výpočtech</p> <p>- řeší slovní úlohy z praxe s využitím druhé mocniny a odmocniny</p> <p>- zapíše číslo ve tvaru mocniny</p> <p>- provádí početní operace s mocninami s p irozeným mocnitelem</p>	<p><u>Druhá mocnina a odmocnina</u></p> <p>- pojem</p> <p>- tení a zápis druhých mocnin a odmocnin</p> <p>- určení druhých mocnin a odmocnin</p> <p>- slovní úlohy z praxe</p> <p><u>Mocniny s p irozeným mocnitelem</u></p> <p>- tení a zápis mocnin</p> <p>- zápis čísla pomocí mocnin deseti</p> <p>- početní operace s mocninami s p irozeným mocnitelem</p>	<p>Fyzikální zápis jednotek fyzikálních veličin</p>
<p>- zdvodňuje a využívá polohové a metrické vlastnosti základních rovinných útvarů při řešení úloh a jednoduchých polohových problémů, využívá potřebnou matematickou symboliku</p> <p>- analyzuje a řeší aplikativní geometrické úlohy s využitím osvojeného matematického aparátu</p>	<p>- rozliší odvození a p epony</p> <p>- rozumí odvození vzorce Pythagorovy v t</p> <p>- zná Pythagorovu v t</p> <p>- využívá poznatků při výpočtu délek stran pravoúhlého trojúhelníku</p> <p>- užívá Pythagorovu v t při řešení slovních úloh z praxe</p> <p>- zaokrouhluje a provádí odhady s danou přesností</p>	<p><u>Pythagorova v t</u></p> <p>- strany pravoúhlého trojúhelníka</p> <p>- pojem Pythagorova v t</p> <p>- výpočet délek stran v pravoúhlém trojúhelníku</p> <p>- užití Pythagorovy v t ve slovních úlohách</p>	<p>Fyzika</p> <p>- výslednice kolmých sil</p>

<ul style="list-style-type: none"> - zdvoduje a vyúslovává polohové i metrické vlastnosti základních rovinných útvarů při řešení úloh a jednoduchých praktických problémů, vyúslovává matematickou symboliku - odhaduje a vypočítává obvod a obsah základních rovinných útvarů 	<ul style="list-style-type: none"> - určí vzájemnou polohu přímky a kružnice - sestrojíte kružnici v daném bodě kružnice i z daného bodu ležícího vně kružnice - určí vzájemnou polohu dvou kružnic - vypočítává obvod a obsah kruhu, délku kružnice - umí vyznačit tečtu a vypočítat její délku - řeší slovní úlohy na výpočet obsahu a obvodu kruhu, délky kružnice 	<p><u>Kruh, kružnice</u></p> <ul style="list-style-type: none"> - vzájemná poloha přímky a kružnice - tečna z bodu ke kružnici - tečtu kružnice - vzájemná poloha dvou kružnic - délka kružnice - obsah kruhu - slovní úlohy 	<p>Zeměpis</p> <p>poledníky a rovnoběžky zeměkoule</p>
<ul style="list-style-type: none"> - matematizuje jednoduché reálné situace s využitím proměnných, určí hodnotu výrazu, sčítá a násobí mnohočleny, provádí rozklad mnohočlena na součiny pomocí vzorce a vytýkáním 	<ul style="list-style-type: none"> - rozumí pojmu výraz - určí hodnotu číselného výrazu - zapíše pomocí výrazu s proměnnou slovní text - dokáže se číst a vynásobit mnohočleny - dosadí do výrazu s proměnnou - správně vytkne před závorku při rozkladu výrazu na součiny - užívá vzorce $(a + b)^2$, $(a - b)^2$ a $a^2 - b^2$ ke zjednodušení výrazu 	<p><u>Výrazy</u></p> <ul style="list-style-type: none"> - číselné výrazy - proměnné - výrazy s proměnnou - početní výkony s výrazy - rozklad mnohočlenu na součiny - užití vzorce 	

<ul style="list-style-type: none"> - určuje a charakterizuje základní prostorový útvar (t lesa) - analyzuje jeho vlastnosti - odhaduje a vypočítá objem a povrch - načrtne a sestrojí sí základního t lesa 	<ul style="list-style-type: none"> - pozná válec a popíše ho jako t lesa - dokáže sestrojít sí válce - vypočítá povrch a objem válce, určí správné jednotky výsledku - analyzuje a řeší slovní úlohy na objem a povrch válce 	<p><u>Válec</u></p> <ul style="list-style-type: none"> - pojem - sí válce - povrch válce - objem válce - slovní úlohy 	
<ul style="list-style-type: none"> - formuluje a řeší reálnou situaci pomocí rovnic <ul style="list-style-type: none"> - analyzuje a řeší jednoduché problémy, modeluje konkrétní situace, v nichž využívá matematický aparát v oboru celých a racionálních čísel 	<ul style="list-style-type: none"> - rozumí vztahu rovnosti - chápe ekvivalentní úpravy pí e-ení rovnic a dokáže je správně aplikovat - provádí zkoušku správnosti e-ení - umí vypočítat hodnotu neznámé ze vzorce po dosazení číselných hodnot všech daných veličin <ul style="list-style-type: none"> - matematizuje jednoduché reálné situace - vyřeší daný problém aplikací získaných matematických poznatků a dovedností - řeší slovní úlohy a příklady z praxe pomocí lineárních rovnic 	<p><u>Lineární rovnice</u></p> <ul style="list-style-type: none"> - rovnost - lineární rovnice - výpočet neznámé ze vzorce <p><u>Slovní úlohy</u></p> <ul style="list-style-type: none"> - slovní úlohy řešené rovnicemi - geometrické slovní úlohy řešené rovnicemi 	

<ul style="list-style-type: none"> - vyučívá pojem množina v-ech bod dané vlastnosti - k charakteristice útvaru a k e-ení polohových a nepolohových konstruk ních úloh 	<ul style="list-style-type: none"> - dodrfluje základní pravidla p esného rýsování - umí sestrojít jednoduché konstrukce - rozumí pojmu množina bod dané vlastnosti a dokáffe ji aplikovat v konstruk ních úlohách - chápe pojem Thaletova v ta - učívá Thaletovy krufnice v konstruk ních úlohách - vyučívá poznatk (vý-ka, t fnice, Thaletova krufnice, st ední p í ka) v konstruk ních úlohách 	<p><u>Konstruk ní úlohy</u></p> <ul style="list-style-type: none"> - p esnost rýsování - jednoduché konstrukce - množiny v-ech bod dané vlastnosti - Thaletova v ta - Thaletova krufnice - konstrukce trojúhelník a ty úhelník 	
<ul style="list-style-type: none"> - vyhledává, vyhodnocuje a zpracovává data - porovává soubory dat 	<ul style="list-style-type: none"> - te a sestavuje jednoduché tabulky a diagramy - zaznamená výsledky jednoduchých statistických -et ení do tabulek - vyhledá a vyhodnotí jednoduchá statistická data v grafech a tabulkách 	<p><u>Shromaflování, t íd ní a vyhodnocování statistických údaj</u></p> <ul style="list-style-type: none"> - základní statistické pojmy - základní charakteristiky statistického souboru 	<p>Zem pis, p írodopis ó srovnávání obyvatel stát , údaje z pr mysly, zdravotnictví ó diagramy, grafy, pr m ry, údaje v médiích, stav ovzdu-í atd.</p> <p>VMEG 2 ó Objevujeme Evropu a sv t - EV 2 - Základní podmínky flivota - MV 8 - Kritické vnímání mediálních sd lení</p>
		<p><u>Záv re né opakování</u></p>	

Vzdělávací oblast: Matematika a její aplikace

Vyučovací předmět: **Matematika**

Ročník: **9.**

Očekávané výstupy	Dílčí výstupy	Učivo	Průřezová témata. mezipředmětové vztahy
- určí hodnotu výrazu	- určí podmínky, za kterých má daný lom. výraz smysl - krátí a rozšiřuje lomené výrazy - provádí početní operace s lom. výrazy	Lomený výraz (opakování) - pojem lomený výraz - krácení a rozšiřování lom. výraz - úpravy a operace lomených výraz	
- formuluje a řeší reálnou situaci pomocí rovnic a jejich soustav	- řeší jednoduché lineární rovnice s neznámou ve jmenovateli - řeší slovní úlohy	Rovnice s neznámou ve jmenovateli	

<p>- uflívá k argumentaci a p i výpo tech v ty o podobnosti trojúhelník</p>	<ul style="list-style-type: none"> - e-í soustavu dvou lineárních rovnic se dv ma neznámými (metoda s ítací a dosazovací) - provádí zkou-ku e-ení - e-í slovní úlohy pomocí soustav lin. rovnic <p>provádí zkou-ku e-ení</p> <ul style="list-style-type: none"> - rozli-í shodné a podobné útvary - ur uje podobné útvary v rovin - ur uje a pouflívá pom r podobnosti - uflívá k argumentaci a p i výpo tech v ty o podobnosti trojúhelník - sestruje rovinný obraz podobný danému - rozd luje úse ku dané délky v daném pom ru - uflívá pom r podobnosti p i práci s plány a mapou 	<p>Soustavy rovnic</p> <ul style="list-style-type: none"> - soustava dvou lineárních rovnic se dv ma neznámými - slovní úlohy e-ené pomocí soustav lineárních rovnic <p>Podobnost</p> <ul style="list-style-type: none"> - podobnost - pom r podobnosti - podobnost trojúhelník - v ty o podobnosti trojúhelník - d lení úse ky v daném pom ru - technické výkresy, plány a mapy 	
---	--	--	--

<ul style="list-style-type: none"> - vyjádří funkční vztah tabulkou, rovnicí, grafem - matematizuje jednoduché reálné situace s využitím funkčních vztahů 	<ul style="list-style-type: none"> - chápe pojem funkce - rozeznává funkční vztah od jiných vztahů - zakreslí bod v PSS - definítní obor funkce a množinu hodnot funkce - vyjádří funkční vztah tabulkou, rovnicí, grafem - sestavuje graf lineární funkce, kvadratické funkce $y=ax^2$, nepřímá úměrnost $y = \frac{k}{x}$ - řeší graficky soustavu dvou lineárních rovnic - uvolňuje probrané funkce při řešení úloh z praxe 	<p>Funkce</p> <ul style="list-style-type: none"> - pravoúhlá soustava souřadnic - pojem funkce - lineární funkce - grafické řešení soustav dvou lineárních rovnic - kvadratická funkce $y=ax^2$ a její graf - nepřímá úměrnost $y = \frac{k}{x}$ a její graf 	<p>OSV</p> <p>1- Rozvoj schopnosti poznávání, řešení problémů a rozhodovací dovednosti řešení z grafů, jízdních řádů, spotřeby pohonných hmot</p>
<ul style="list-style-type: none"> - matematizuje jednoduché reálné situace s využitím funkčních vztahů 	<ul style="list-style-type: none"> - sestavuje grafy funkcí sinus a tangens pro hodnoty úhlů v intervalu $\langle 0^\circ; 90^\circ \rangle$ - uvolňuje goniometrické funkce sinus a tangens při výpočtech 	<p>Goniometrické funkce</p> <ul style="list-style-type: none"> - goniometrické funkce jako poměry stran v pravoúhlém trojúhelníku - funkce sinus, kosinus, tangens - uvolnění 	

<ul style="list-style-type: none"> - určuje a charakterizuje základní prostorové útvary (tělesa), analyzuje jejich vlastnosti - řeší úlohy na prostorovou představitelost, aplikuje a kombinuje poznatky a dovednosti z různých tematických a vzdělávacích oblastí 	<ul style="list-style-type: none"> - nártne a sestrojí sí jehlanu a kufle - odhaduje a vypočítá objem a povrch těles 	<p>Tělesa</p> <ul style="list-style-type: none"> - kufle - jehlan - koule <p>- povrch a objem těles</p>	<p>OSV</p> <p>1- Rozvoj schopnosti poznávání, 10 řešených problémů a rozhodovací dovednosti o spotřebě materiálu, stavebnictví</p>
<ul style="list-style-type: none"> - určívá různé způsoby kvantitativního vyjádření vztahu celek - část procentem 	<ul style="list-style-type: none"> - vypočítá úrok z dané jistiny za určité období - řeší úlohy z praxe na jednoduché úrokování 	<p>Základy finanční matematiky</p> <ul style="list-style-type: none"> - základní pojmy finanční matematiky 	<p>OSV</p> <p>1- Rozvoj schopnosti poznávání, 10 řešených problémů a rozhodovací dovednosti - plat, srážky, úroky, í</p>

Anotace

Jméno a příjmení:	Bc. Blanka Štejnčíková
Katedra:	Katedra matematiky
Vedoucí práce:	Mgr. David Nocar, Ph.D.
Rok obhajoby:	2014

Název práce:	Využití Visual Basic ve stereometrii na Z TM
Název v angličtině :	The Use of Visual Basic in Stereometry at an Elementary School
Anotace práce:	<p>Teoretická část shrnuje základní poznatky o výukovém softwaru. Dále se zabývá zejména stereometrií do RVP ZV a obsahuje základní informace o jazyce Visual Basic. Hlavní částí je popis uživatelského prostředí aplikace MS Visual Studio 2010 Express ve které je vytvořen výukový software Prostorová tělesa.</p> <p>V praktické části je popisován výukový program Prostorová tělesa.</p>
Clíčová slova:	Výukový software, stereometrie, Visual Basic, MS Visual Studio 2010 Express.
Anotace v angličtině :	<p>The theoretical part summarizes basic information about educational software. It further studies the incorporation of stereometry within the framework educational programme for elementary education (RVP ZV) and contains basic information about the Visual Basic language. The main part describes the user interface of MS Visual Studio 2010 Express where the educational software Prostorová tělesa is created.</p> <p>Educational software Prostorová tělesa is described in the practical part.</p>
Clíčová slova v angličtině :	Educational software, stereometry, Visual Basic, MS Visual Studio 2010 Express.
Přílohy vázané v práci:	CD-ROM
Rozsah práce:	82 stran
Jazyk práce:	český

