

Obsah

Úvodem	9
Komu je kniha určena	9
Forma výkladu	9
Konkrétní postup výuky	10
Příklady ke knize	11
Kapitola 1	
Co je to počítačové programování	13
Co je to program a jak ho vytvořit	13
Nádražní automat	13
Fotbal se hraje týmově	15
Volba správné varianty	15
Digitální budík – pravidelně opakovaná, jednotvárná činnost	16
Stavíme odrazový můstek programování ze čtyř základních kamenů	17
Kdo je programátor	18
Za prvé – Rozbor problému	18
Za druhé – Návrh funkčního a efektivního řešení	20
Za třetí – Realizace řešení	24
Za čtvrté – Sledování funkčnosti a vyladění řešení	25
Závěrem k práci programátora	25
Příkazy	26
Co je to příkaz a proč je tak důležitý	26
Příkazy samy o sobě nejsou všemocné	30
Náš první příkaz pro počítač	31
Algoritmus	32
Co musí splňovat algoritmus a proč	33
K čemu se hodí nesprávně fungující algoritmus	35
Složitost algoritmu	35
Tvoříme algoritmy s nízkou složitostí	37
K čemu je dobrý algoritmus s vysokou složitostí	38
Důležité algoritmy – seřadíme fotbalové hráče podle velikosti	39

Jak to funguje v počítači	41
Dorozumívání programátora s počítačem	42

Kapitola 2

První krůčky velkého programátora **43**

Stažení a instalace překladače	43
---------------------------------------	-----------

Spuštění vývojového prostředí	46
-------------------------------	----

Zdrojový kód a jeho překlad	46
------------------------------------	-----------

Jak to celé funguje	47
---------------------	----

Klíčová slova	47
---------------	----

Seznámení s vývojovým prostředím jazyka Pascal	48
---	-----------

První pokus	49
-------------	----

Provádění programu	51
--------------------	----

Co když uděláme chybu	52
-----------------------	----

Krojujeme program	53
-------------------	----

Pracujeme s okny vývojového prostředí	54
---------------------------------------	----

Hledáme zadaný text ve všech souborech adresáře	56
---	----

Změna aktuálního adresáře pro ukládání a vyhledávání	57
--	----

Kopírování textu z/do Windows	57
-------------------------------	----

Píšeme úhledně	58
-----------------------	-----------

Zarovnání	58
-----------	----

Komentáře	60
-----------	----

Kapitola 3

Jak využít paměť počítače **63**

Proměnná	63
-----------------	-----------

Paměť počítače je jako knihovna	64
---------------------------------	----

Výpis obsahu proměnné	65
-----------------------	----

I uživatel může udělit proměnné hodnotu	65
---	----

Jak nejlépe pojmenovat proměnnou	66
----------------------------------	----

Proč se proměnná jmenuje proměnná	67
-----------------------------------	----

K čemu ještě využijeme proměnné	71
---------------------------------	----

Datový typ	71
-------------------	-----------

Co je to datový typ a k čemu je dobrý	71
---------------------------------------	----

Řetězce (datový typ String)	72
-----------------------------	----

Krátce k jednotlivým znakům (datový typ char)	74
---	----

Celá čísla (datový typ integer)	75
---------------------------------	----

Desetinná čísla (datový typ real)	76
-----------------------------------	----

Pravda versus Nepravda (datový typ boolean)	77
---	----

Složené datové typy	79
---------------------	----

Kapitola 4

Rozhodujeme v podmínkách

85

Využíváme výsledky	85
Příklady v jazyce Pascal	86
Opačný případ	87
Elegantní řešení dvou možností	87
Více příkazů u jedné podmínky	88
Vnořený if – jde to i bez něho	89
Přísná logika	90
Výběr z několika možností	92

Kapitola 5

Opakujeme v cyklech

95

Horká voda na tři nápoje	96
Jak to funguje	98
Jak vypadá chybně napsaný cyklus	99
Zkusme cyklus vzhůru nohama	100
Určíme počet průchodů cyklem	102
Možnosti kombinací: žlutý nebo červený	104
Spojení několika cyklů dohromady	107

Kapitola 6

Jednoduché i složitější programy

109

Hry s řetězci	110
Obrácení řetězce	110
Hledání řetězce v textu	115
Náhrada textu za jiný – delší i kratší	120
Čtení ze souboru a zápis do souboru	126
Vytváříme textový soubor a zapisujeme do něho	126
Čteme z textového souboru	129
Hry s čísly	132
Rekurze a králíci	132
Náhoda je ...	137
Prvočísla	140
Složitost algoritmu	144
Závěrem ke kapitole plné praktického programování	146

Kapitola 7

Využíváme složené datové typy **147**

Pracujeme s polem **147**

- Las Vegas v české kotlině 148
- Seřadíme prvky pole podle velikosti 153
- Seřadíme písmena podle abecedy 157
- Řecký zlatokop prvočísel 161

Tvoříme jednoduchou databázi **168**

- Co je to databáze a jak ji v Pascalu vytvořit 169
- Malá půjčovna filmů 171
- Vyhledávání v databázi 174

Ukazatele – směrovky k opravdovým hodnotám **175**

- K čemu jsou ukazatele dobré 176
- Nicku, máš prázdný zásobník 180
- Postav se do fronty na náboje 184

Kapitola 8

Využití dříve napsaných programů **189**

Jarda umí sečíst dvě čísla – využíváme funkce **189**

- Krátká odbočka: argumenty versus parametry 192

I Karel umí sčítat – využíváme procedury **192**

Globální a lokální proměnné **193**

- Svážeme proměnné dohromady pomocí volání odkazem 194

Voláme funkci – ta volá funkci – ta volá... **196**

- Faktoriál 196
- Králík znovu zasahuje 197
- Šifrujeme zprávy 199

Využíváme funkce jiných programátorů **202**

- Zastavit losování 203
- Sestavíme si vlastní knihovnu 207
- Všechny dostupné funkce a procedury překladače Free Pascal 216

Závěr a další osudy programátorů **217**

Po odložení této knihy – nejen další jazyky **217**

- Algoritmy a datové struktury 217
- Lepší zdrojové kódy 218
- Do praxe 218

Příloha A

Druhy programovacích jazyků a doporučená literatura **219**

Mít či nemít překladač **219**

Jazyk C	219
C++	220
Jazyk C#	220
Visual Basic	221
Java	221

Jde to i bez překladu **222**

Shell a Bash	223
Python	223
JavaScript	224
PHP	224

Když se myslí jazykem **225**

Imperativní jazyky	225
Logické jazyky	225
Funkcionální jazyky	226

Závěrem **227**

Příloha B

Opakování nejdůležitějších pravidel a pojmů **229**

Postup při psaní programů **229**

Pravidla dobrého programátora **229**

Komentáře	230
Pomocné výpisy	230
Přehlednost napsaného kódu	230
Recyklace	230
Složitost algoritmu	230
Využití vhodných datových typů	231

Souhrn operátorů **231**

Logické operátory	231
Porovnávání čísel	232
Číselné operace	232
Znakové operace	233

Příloha C

Slovníček pojmů **235**

Algoritmus	235
Aplikace	235
ASCII tabulka	236

Databáze	236
Datová struktura	236
Datový typ	236
Deklarace	237
Dynamická datová struktura	237
Eratosthenovo síto	237
Faktoriál	238
Fibonacciho posloupnost	238
Funkce	238
Globální proměnná	238
Inicializace	239
Klíčové slovo	239
Knihovna	239
Komentář	240
Kompilátor	240
Logické operátory	240
Lokální proměnná	240
nil	241
Ordinální datový typ	241
Procedura	241
Programovací jazyk	241
Proměnná	241
Prvočíslo	241
Překladač	242
Rekurze	242
Složitost	242
Strojový kód	242
Uživatel	242
Uživatelské rozhraní	243
Volání hodnotou	243
Volání odkazem	243
Vývojář	243
Vývojové prostředí	243
Zdrojový kód	243

Rejstřík

245

Úvodem

Programování se rychle stalo jedním z nejdůležitějších oborů lidské činnosti. Počítačem se dnes řídí téměř vše: od digitálních hodinek, mobilních telefonů a praček, přes požární systémy či ovládání světel až po letadla nebo kosmické rakety.

Stěrače v autě například umějí reagovat na sílu deště a podle toho samy určují frekvenci stírání okna. Jízda je pak pohodlnější, a tím i bezpečnější. Díky ulehčení a zpříjemnění všeho, co děláme, se kvalitně vytvořené počítačové programy dostávají čím dál blíže našemu praktickému životu.

Jak brzo uvidíme, umět programy vytvářet není vůbec těžké. Je pouze třeba osvojit si jistý způsob myšlení, který je nám ovšem už dávno vlastní. Stačí ho jen využít. Pokud navíc zjistíme, jak s počítačem komunikovat, máme ho v hrsti – bude dělat, co mu přikážeme.

Komu je kniha určena

Učebnice je pro všechny, kteří mají nějakou zkušenost s počítači, ale o programování zatím vůbec nic nevědí. Příručka je určena každému, kdo se chce naučit základním dovednostem a technikám programování, ale i těm čtenářům, které programování zajímá jen jako silný fenomén dnešní doby. Pro počítačové programování totiž platí:

- nezáleží na věku, ve kterém začneme programovat,
- nezáleží ani na výši vzdělání, protože v programování jde pouze o způsob přemýšlení a dorozumění se s počítačem,
- nikdy není pozdě s programováním začít.

Pro výuku z této publikace potřebujeme jen dvě věci: počítač a chuť učit se novým věcem.

Forma výkladu

Všechny důležité dovednosti i znalosti si osvojíme doslova krok za krokem a bez zabíhání do zbytečně komplikovaných a nepotřebných detailů. Celý výklad se totiž drží několika základních pravidel:

- začneme zcela od nuly,
- pomalu se budeme učit přemýšlet jako programátor,
- naučíme se mluvit s počítačem tak, aby nás poslouchal,

- vždy upozorníme na užitečné tipy a na možná úskalí,
- text doplňuje mnoho názorných obrázků,
- snadnému pochopení pomáhá i přímé propojení textu s probíraným jevem na obrázku,
- na konci budeme znát všechny nejdůležitější programátorské techniky,
- tyto techniky budeme umět použít v programátorské praxi.

V textu se vyskytuje několik speciálních odstavců, které mají za úkol doplnit základní výklad o další užitečné, podstatné či prostě zajímavé okolnosti:



Poznámka: Poskytuje rozšiřující informace na okraj.



Tip: Doplnuje výklad o zajímavou a prospěšnou informaci.



Důležité: Informace, které byste si měli zapamatovat.



Řešení problému: Pokud hrozí, že se v některém postupu mohou objevit potíže, tento odstavec na ně upozorní a vysvětlí, jak je řešit.

Pro ještě názornější ukázky jsou některé části programových kódů **tučně zvýrazněny**. Tučnost nemá v programu žádný význam – slouží pouze pro zvýraznění právě probíraného pojmu nebo nějaké důležité techniky.

Konkrétní postup výuky

O co při programování jde. Nemůžeme se hned vrhnout ke klávesnici a začít psát programy. Podobně jako plavci si některé základní techniky osvojíme nejprve na břehu. Necháme počítač ještě několik okamžiků vypnutý a v *kapitole 1* si povíme o těch nejdůležitějších věcech: jak se má programátor chovat, jaká dodržovat pravidla a také se naučíme první důležité pojmy. Bez těchto základů bychom se do hlubší vody nemohli pustit.

Potřebné nástroje a první program. S počítačem je potřeba se nějak dorozumět a v *kapitole 2* si řekneme jak. Z Internetu získáme volně dostupný nástroj, který poslouží jako překladač z naší řeči do jazyka počítače a naopak.

Druhá kapitola bude opravdu výjimečná, protože právě v ní se to stane poprvé! Dlouho očekávaný, sice prostinký a jednoduchý, ale krásný a plně funkční prográmeček konečně spatří světlo světa. V té chvíli se každý z nás stane programátorem. Nebude cesty zpět.

Počítačová paměť. Bez paměti bychom to daleko nedotáhli. Plavat už umíme, ale proč to nezkusit lodí? V *kapitole 3* se naučíme pracovat s pamětí počítače: zjistíme, jak do ní ukládat věci na pozdější časy a jak je odtud zase získat zpět.

Podmínky a cykly. Dalším krokem na cestě ke skvělým programům bude obyčejné rozhodování a opakování. Zní to jednoduše a je to jednoduché. Přesto jde o nejdůležitější programátorské techniky, bez kterých nelze vyplout na otevřené moře. Podmínky a cykly probereme v *kapitolách 4 a 5*.

Několik jednoduchých programů. Všechno, co již umíme, si prakticky vyzkoušíme na několika programech, ukážeme si pár pěkných triků, jak si zjednodušit práci, budeme vytvářet náhodná čísla a pracovat se soubory. To vše v *kapitole 6*.

Složitější hry s pamětí. Využívat paměť počítače se sice naučíme v kapitole 4, ale na ostřejší kousky si troufneme později. V *kapitole 7* se mimo jiné naučíme tvořit seznamy lidí i s jejich základními údaji (číslo telefonu, adresa apod.). Vzájemně provázané údaje budeme ukládat do paměti a také si ukážeme, jak je z ní jednoduše dostávat.

Využití dříve napsaných programů. Není třeba hotový program tvořit znovu a znovu. Pokud je funkční a v pořádku, využijme ho. Názorně si vše předvedeme v *kapitole 8*. V té se dozvíme i to, jak využívat programy, které napsal někdo jiný. Programátoři celého světa spolupracují a napsané programy běžně poskytují.

Příklady ke knize

Celý balík všech cvičení z knihy si čtenáři mohou stáhnout z adresy <http://knihy.cpress.cz/k1620> – najdete je na záložce **Soubory ke stažení**.

1. Kódy si stáhněte a uložte do počítače.
2. Následně balík rozbalte pomocí některého archivačního programu (např. WinRAR – viz www.winrar.cz).
3. Po rozbalení se na disku vytvoří složka *Programovani_PUZ*, v níž naleznete soubory příkladů uložené v jednotlivých složkách podle kapitol.

Ke snadnější orientaci slouží přehledné rozhraní, které spustíte poklepáním na soubor *spustit.html* v hlavní složce. Dále se řiďte pokyny rozhraní.