

knihovna programátora

- Učebnice jazyka i referenční příručka v jednom
- Probírá konstrukce nevyžadující hlubší znalost objektového programování
- Výklad většiny konstrukcí doplňují syntaktické diagramy
- Použití konstrukcí je předvedeno na příkladech
- Rozbor příčin častých chyb a jejich řešení
- Na knihu navazuje další titul věnovaný objektovým konstrukcím



Rudolf Pecinovský

Python 3.14

ALGORITMICKÉ KONSTRUKCE



knihovna programátora

RUDOLF PECINOVSKÝ

Python 3.14

ALGORITMICKÉ KONSTRUKCE

GRADA
Publishing

Rudolf Pecinovský

Python 3.14

Algoritmické konstrukce

Vydala Grada Publishing, a.s.
U Průhonu 22, Praha 7
obchod@grada.cz, www.grada.cz
tel.: +420 234 264 401
jako svou 10344. publikaci

Odpovědný redaktor Petr Somogyi
Fotografie na obálce Depositphotos/mario7
Grafická úprava a sazba Rudolf Pecinovský
Počet stran 496
První vydání, Praha 2025
Vytisky Tiskárny Havlíčkův Brod a.s.

© Grada Publishing, a.s., 2025
Cover Design © Grada Publishing, a. s., 2025
Cover Photo © Depositphotos/mario7

*Upozornění pro čtenáře a uživatele této knihy:
Všechna práva vyhrazena. Žádná část této tištěné či elektronické knihy
nesmí být reprodukována a šířena v papírové, elektronické či jiné podobě
bez předchozího písemného souhlasu nakladatele.
Neoprávněné užití této knihy bude trestně stíháno.
Názvy produktů, firem apod. použité v knize mohou být ochrannými známkami
nebo registrovanými ochrannými známkami příslušných vlastníků.
Automatizovaná analýza textů nebo dat ve smyslu čl. 4 směrnice 2019/790/EU
a použití této knihy k trénování AI jsou bez souhlasu nositele práv zakázány.*

ISBN 978–80–271–8342–5 (ePub)
ISBN 978–80–271–8341–8 (pdf)
ISBN 978–80–271– 6061–7 (print)

*Mé ženě Jarušce a dětem
Štěpánce, Pavlínce, Ivance a Michalovi*

Stručný obsah

Úvod	21
------------	----

Část A Superzáklady	31
----------------------------	-----------

Kapitola 1 Startujeme	32
Kapitola 2 Zadávání jednoduchých hodnot	57
Kapitola 3 Zadávání textů – stringů	73
Kapitola 4 Volání funkcí	90
Kapitola 5 Jednoduché výrazy	109
Kapitola 6 Proměnné, výrazy, příkazy	123
Kapitola 7 Logické hodnoty a operace	145
Kapitola 8 Další jednoduché příkazy	157

Část B Složené příkazy	167
-------------------------------	------------

Kapitola 9 Moduly	168
Kapitola 10 Vytvoření vlastního modulu	185
Kapitola 11 Balíčky	206
Kapitola 12 Definice funkcí	221
Kapitola 13 Parametry, argumenty a lokální proměnné funkcí	234
Kapitola 14 Pokročilé rysy funkcí	255
Kapitola 15 Rozhodování a příkaz with	276
Kapitola 16 Opakování kódu – rekurze a cykly	289
Kapitola 17 Tvorba aplikací	309
Kapitola 18 Ošetřování chyb	323

Část C Kontejnery	347
Kapitola 19 Seznamy.....	348
Kapitola 20 N-tice	367
Kapitola 21 Množiny.....	382
Kapitola 22 Slovníky	395
Kapitola 23 Rozšíření definic funkcí	409
Kapitola 24 Formátování stringů	428
Kapitola 25 Operace s kontejnery.....	453
Kapitola 26 Práce se soubory	464
Literatura.....	487
Rejstřík	489

Podrobný obsah

Úvod	21
Komu kniha není určena	21
Komu je kniha určena	22
Struktura příručky	22
Koncepte výkladu	24
Jazyk identifikátorů	25
Potřebné vybavení	25
Operační systém	25
Doprovodné programy a jejich organizace	25
Použité typografické konvence	27
Odbočka – podšeděný blok	29
Zpětná vazba	29
Část A Superzáklady	31
Kapitola 1 Startujeme	32
1.1 Hlavní součást instalace	32
1.1.1 Platforma	32
1.1.2 Dokumentace	33
1.1.3 PEP	34
1.1.4 Pracovní režimy	34
1.1.5 Překladač versus interpret	35
1.2 Vývojová prostředí	35
1.2.1 PyCharm a IntelliJ IDEA	35
1.2.2 Visual Studio Code	36
1.2.3 Jupyter Notebook a JupyterLab	36
1.2.4 Řádkový editor a IDLE	37
1.3 Spuštění Pythonu	37
1.3.1 Spouštění ve Windows	37
1.3.2 Spouštění v macOS a Linuxu	39
1.3.3 Co je nejdůležitější	39
1.3.4 Spuštění Pythonu v demonstračních programech v knize	39
1.4 Řádkový editor – REPL	39
1.4.1 Komentáře	40
1.4.2 Odsazování	41
1.4.3 Víceřádkové příkazy	41
Fyzické a logické řádky	42
1.4.4 Dvě verze řádkového editoru	42
1.5 Spuštění skriptu napsaného v Pythonu	48
1.5.1 Windows	49
1.6 Prostředí IDLE	49
1.6.1 Spuštění IDLE	49
1.6.2 Základní popis	50
1.6.3 Příkazové okno	51

1.6.4	Editační okno	53
1.6.5	Umístění editovaných souborů	54
1.6.6	Problémy při práci se schránkou	54
1.6.7	Barevné zvýraznění textu	55
1.6.8	Pokročilá nastavení	55
1.7	Shrnutí kapitoly a doprovodné soubory	56
Kapitola 2	Zadávání jednoduchých hodnot	57
2.1	Objekty a objektové programování	58
2.1.1	Koncepce Pythonu	58
2.1.2	Explicitně	59
2.1.3	Objekt, třída, instance, odkaz	59
2.1.4	Kontejnery	61
2.2	Datový typ	62
2.3	Nejdůležitější zvláštnosti Pythonu	63
	Přísné a benevolentní programovací jazyky	63
2.4	Zápis celých a desetinných (reálných) čísel	64
2.4.1	Zpřehlednění dlouhých čísel pomocí znaku podtržení	65
2.4.2	Rozlišení celých a reálných čísel	66
2.5	Komplexní čísla	67
2.6	Počáteční nula u celých čísel	68
2.7	Zadávání čísel v jiných číselných soustavách	68
2.8	Logické hodnoty – platí × neplatí	69
2.9	Nic – None	69
2.10	Výpustka – Ellipsis,	70
2.11	Objekt NotImplemented	70
2.12	Literály	70
2.13	Důležitost přehlednosti	71
2.14	Shrnutí kapitoly a doprovodné soubory	72
Kapitola 3	Zadávání textů – stringů	73
3.1	Stringové literály	73
3.1.1	Víceřádkové stringy	74
3.1.2	Použití funkce <code>print()</code>	76
3.2	Komentáře	76
3.3	Escape sekvence	77
3.3.1	Zobrazení escape-sekvencí funkcí <code>print()</code>	81
3.3.2	Zjištění názvu znaku	82
3.4	Slučování sousedních textových literálů	82
3.5	Prefixy stringových literálů	83
3.6	Stringová interpolace – f-stringy	85
3.6.1	Samodokumentující se výrazy	85
3.6.2	Shrnutí zásad pro práci s f-stringy	85
3.7	Bajtové objekty	87
3.7.1	Bajtové stringy a bajtové literály	87
3.7.2	Třída <code>bytearray</code> – zadávání bajtových polí	88
3.8	Shrnutí kapitoly a doprovodné soubory	89
Kapitola 4	Volání funkcí	90
4.1	Volání volatelného objektu	90
4.1.1	Parametr versus argument	91
4.1.2	Syntaxe volání a návratová hodnota	92
4.1.3	Pořadí vyhodnocování a předávání argumentů	92
4.1.4	Datové × funkční (volatelné) objekty	92
4.2	Vytváření objektů	93
4.3	Vestavěné funkce s jednoduchými argumenty	94

4.3.1	Zápis syntaxe.....	94
4.3.2	Anotace	95
4.3.3	Signatura funkce	96
4.3.4	Přehled vestavěných funkcí s argumenty jednoduchých typů	97
4.4	Získání nápovědy	104
4.4.1	Argument zadán	104
4.4.2	Bez argumentu.....	106
4.5	Rozdělení volání na více řádků	107
4.6	Shrnutí kapitoly a doprovodné soubory	107
	Kapitola 5 Jednoduché výrazy	109
5.1	Trocha teorie	109
5.1.1	Výraz	109
5.1.2	Operace	110
5.1.3	Operand	110
5.1.4	Operátor	110
5.1.5	Arita operátorů.....	110
	Operátory jako volatelné objekty – funkce	111
5.1.6	Priorita operátorů	111
5.1.7	Asociativita binárních operátorů	113
5.2	Závorky, čárka jako operátor vytvoření n-tice	114
5.2.1	Pozor na záměnu čárky s desetinnou tečkou	115
5.3	Numerické operace	116
5.3.1	Tři druhy dělení.....	116
5.3.2	Umocňování.....	117
5.3.3	Nekonečna a nesmyslná čísla.....	118
5.4	Operace se stringy	119
5.4.1	Sečítání textů	120
5.4.2	Násobení stringu celým číslem.....	120
5.4.3	Indexace jednotlivých znaků.....	121
5.5	Shrnutí kapitoly a doprovodné soubory	122
	Kapitola 6 Proměnné, výrazy, příkazy	123
6.1	Co jsou to proměnné	123
6.1.1	Proměnná versus atribut – kvalifikace.....	124
6.2	Správa paměti	124
	Statické a dynamické typování	125
6.3	Pravidla pro tvorbu identifikátorů	126
6.3.1	Klíčová slova tvrdá a měkká	127
6.3.2	Systémové identifikátory – dunder.....	127
6.4	Zavedení proměnné – přiřazovací příkaz	128
6.5	Výrazy × příkazy, výrazové příkazy	130
6.5.1	Výrazové příkazy	130
6.5.2	Zadání více příkazů na řádku – oddělovací středník	130
6.5.3	Zadání skupiny hodnot.....	131
6.5.4	Proměnné inf a nan	133
6.5.5	Proměnná _	133
6.6	Vnořená volání volatelných objektů	134
6.7	Zjištění typu objektu v proměnné	135
6.8	Uložení funkce do proměnné	136
6.9	Lambda-výrazy	137
6.9.1	Využití při snižování počtu argumentů	138
6.10	Datové a funkční proměnné	139
6.11	Mezery ve výrazech a příkazech	139
6.12	Uložení do proměnné × propojení s názvem	139
6.13	Přiřazovací výraz	140
6.14	Pomocné proměnné	141

6.15 F-stringy – rozšiřující informace	141
6.15.1 Další pravidla.....	142
6.15.2 Formátování f-stringů.....	143
6.16 Shrnutí kapitoly a doprovodné soubory.....	144
Kapitola 7 Logické hodnoty a operace	145
7.1 Konstanty True a False	145
7.2 Převod jiných hodnot na logické	146
7.3 Logické operátory a operace	147
7.3.1 Zkrácené vyhodnocení	147
7.3.2 Pozor na priority	148
7.4 Porovnávání hodnot.....	149
7.4.1 Porovnání reálných čísel	149
7.4.2 Porovnávání a řazení textů – stringů	150
7.4.3 Zřetěžené porovnávání.....	150
7.4.4 Porovnávání totožnosti objektů – operátor is	151
7.5 Operace s jednotlivými bity	152
7.6 Bitové posuny	153
7.6.1 Aritmetický × logický posun	154
7.7 Podmíněný výraz	154
7.8 Shrnutí kapitoly a doprovodné soubory.....	155
Kapitola 8 Další jednoduché příkazy	157
8.1 Přiřazovací příkaz	157
8.2 Rozšířený přiřazovací příkaz	157
8.3 Příkaz del	158
8.4 Příkaz assert.....	159
Návrh podle kontraktu.....	160
8.5 Příkaz pass	162
8.6 Složené příkazy a odsazování	163
8.6.1 Výhody a nevýhody koncepce <i>Pythonu</i>	164
8.7 Restart systému	165
8.8 Shrnutí kapitoly a doprovodné soubory.....	165

Část B Složené příkazy

167

Kapitola 9 Moduly.....	168
9.1 Základní principy modulů	168
9.1.1 Vše je součástí nějakého modulu	168
9.1.2 Dva názvy objektů.....	168
9.1.3 Zdrojový soubor.....	169
9.1.4 Přeložený soubor	169
9.2 Další trocha teorie OOP	170
9.2.1 Atributy	170
9.2.2 Práce s objekty – kvalifikace	170
9.2.3 Další podrobnosti o tvorbě objektů	171
Terminologická vsuvka: co je konstruktor.....	172
9.3 Příkaz import.....	173
9.3.1 Čistý import jiného modulu	173
9.3.2 Import modulu pod jiným názvem	175
9.3.3 Přímý import vyjmenovaných objektů.....	176
9.3.4 Import objektů modulu nezahrnuje import jejich modulu	177
9.3.5 Argumentem příkazů import a from ... import nesmí být výraz.....	178
9.3.6 Hromadný import všech atributů modulu – hvězdičkový import	178
9.3.7 Systémové identifikátory	180
9.3.8 Syntaktické diagramy příkazu import.....	180

9.4 Modul jako objekt	181
9.4.1 Modul <code>builtins</code> a zdánlivě neobjektové programování.....	182
9.5 Postup systému při importu modulu	182
9.6 Shrnutí kapitoly a doprovodné soubory.....	183
Kapitola 10 Vytvoření vlastního modulu	185
10.1 Demonstrační modul <code>m10a_Modul</code>	185
10.1.1 Kódová stránka.....	187
10.1.2 Dokumentační komentář a atribut <code>__doc__</code>	187
10.1.3 Kontrolní tisky načítání modulu.....	188
10.1.4 Definice datových atributů	188
10.1.5 Neveřejné atributy	188
10.1.6 Reakce na zadané příkazy	189
10.1.7 Alternativní zveřejnění atributů.....	189
10.2 Modul <code>dbg</code> a kontrolní tisky	190
10.2.1 Co modul <code>dbg</code> není	191
10.3 Průběh importu vytvořeného modulu.....	192
10.3.1 Použitelné názvy.....	193
10.4 Hromadný (hvězdičkový) import	194
10.5 Zveřejnění atributů pomocí <code>__all__</code>	196
10.6 Import jako přiřazovací příkaz – shrnutí.....	198
10.7 Reimport již importovaného modulu	199
10.7.1 Pozor na přímo importované proměnné	201
10.7.2 Specifika funkce <code>importlib.reload()</code>	203
10.7.3 Rozbor chybového hlášení.....	204
10.7.4 Důsledky chybného zavedení modulu	204
10.7.5 Syntaktické chyby	204
10.8 Shrnutí kapitoly a doprovodné soubory.....	205
Kapitola 11 Balíčky.....	206
11.1 Balíčky (packages)	206
11.1.1 Initor standardního balíčku	207
11.1.2 Zobrazování začátku a konce inicializace balíčků a modulů	207
11.1.3 Uspořádání balíčků pro demonstraci	208
11.1.4 Demonstrace importu podbalíčků	210
11.2 Relativní import.....	212
11.2.1 Relativní import z rodičovských balíčků.....	212
11.2.2 Relativní import z podbalíčků	213
11.2.3 Demonstrace relativního importu	213
11.2.4 Cyklický import.....	215
11.3 Co dělat, když interpret na modul nevidí.....	215
11.3.1 Detekce a změna pracovní složky	216
11.4 Kde hledat moduly.....	217
11.4.1 Prohledávané složky	217
11.5 Přidání složky mezi prohledávané	218
11.6 Shrnutí kapitoly a doprovodné soubory.....	219
Kapitola 12 Definice funkcí.....	221
12.1 Povšechně o funkcích.....	221
12.1.1 Funkce je standardní objekt.....	222
12.2 Definice funkce je jen zvláštní přiřazovací příkaz	222
12.3 Definice vlastní funkce.....	222
12.3.1 Jednořádková definice	223
12.3.2 Víceřádková definice.....	224
12.3.3 Interaktivní režim versus zdrojový kód modulu.....	225
12.3.4 Prázdné funkce	226
12.4 Zadávání stringů zabírajících více řádků.....	227

12.4.1 Použití funkce <code>textwrap.dedent()</code>	229
12.5 Definice funkcí v modulu	230
12.6 Kdy se projeví chyby v definici funkce	231
12.7 Shrnutí kapitoly a doprovodné soubory	233
Kapitola 13 Parametry, argumenty a lokální proměnné funkcí	234
13.1 Parametry, argumenty a lokální proměnné	234
13.1.1 Definice	234
13.1.2 Lokální proměnné	235
13.1.3 Volání funkcí s parametry	236
13.1.4 Povinné pojmenované argumenty	237
13.1.5 Povinné poziční argumenty	238
13.1.6 Mix pozičních a pojmenovaných argumentů	239
13.2 Implicitní hodnoty argumentů	240
13.2.1 Ještě jednou porovnání reálných čísel	242
13.3 Konstantnost předdefinovaných hodnot	243
13.4 Funkce s vedlejším efektem	244
13.5 Funkce vracějící hodnotu a příkaz <code>return</code>	244
13.6 Přetěžování funkcí	245
13.6.1 Něco přetížit jde	246
13.7 Anotace	247
13.8 Dekorátory	248
13.9 Pomocné funkce a dekorátory z modulu <code>dbg</code>	249
13.9.1 Základní funkce modulu <code>dbg</code>	249
13.9.2 Přehled nabízených dekorátorů	250
13.10 Shrnutí kapitoly a doprovodné soubory	253
Kapitola 14 Pokročilé rysy funkcí	255
14.1 Vnitřní funkce	255
14.1.1 Odsazování	257
14.1.2 Přístup z okolního kódu	257
14.1.3 Import uvnitř funkce	258
14.2 IDLE a nastavení Show Code Context	258
14.3 Jmenné prostory	259
14.4 Oblast/rozsah platnosti, působnost (scope)	259
14.4.1 Zanoření oblastí platnosti	260
14.5 Lokalita použitých proměnných	261
14.5.1 Volná proměnná	261
14.5.2 Příkaz <code>global</code>	263
14.5.3 Příkaz <code>nonlocal</code>	264
14.6 Vnoření funkce versus vnoření volání funkcí	267
14.7 Zobrazování kontextu	267
14.8 Vnořená volání funkcí	268
14.9 Funkce vyššího řádu	269
14.10 Atributy funkcí	270
14.11 Nelokální proměnné a uzávěry (closures)	271
14.12 Zákaz hvězdičkového importu uvnitř funkcí	272
14.13 Možné řešení cyklického importu	272
14.14 Shrnutí kapitoly a doprovodné soubory	274
Kapitola 15 Rozhodování a příkaz <code>with</code>	276
15.1 Rozhodovací příkazy	276
15.2 Jednoduchý podmíněný příkaz	277
15.3 Úplný podmíněný příkaz	277
15.4 Rozšířený podmíněný příkaz	279
15.5 Přepínač <code>match</code>	279

15.5.1	Trocha terminologie	280
15.5.2	Postup vyhodnocení	281
15.5.3	Sdružování hodnot ve vzorech	281
15.5.4	Klíčové slovo <code>_</code> je jen symbol	282
15.5.5	Další možnosti	282
15.6	Představení knihovny <code>KarelCz7475</code>	283
15.6.1	Práce se světem – dvorkem	283
15.6.2	Zadatelné příkazy	284
15.6.3	Testovatelné podmínky	284
15.6.4	Skrytí aktuálního stavu robota	285
15.7	Příkaz <code>with</code> a správce kontextu	285
15.7.1	Pravidla používání příkazu <code>with</code>	286
15.7.2	Aplikace na příkladu s robotem	286
15.8	Shrnutí kapitoly a doprovodné soubory	288
Kapitola 16 Opakování kódu – rekurze a cykly		289
16.1	Rekurze	289
16.1.1	Nekonečná rekurze	289
16.1.2	Rádně ukončená rekurze	290
	Zásobník návratových adres – ZNA	291
16.1.3	Rekurzivní definice faktoriálu a dodatečná dekorace funkce	291
16.1.4	Nevhodnost použití rekurze pro hluboké volání	293
16.2	Příkaz <code>while</code> – cyklus se vstupní podmínkou	293
16.3	Nekonečný cyklus	294
16.4	Příkaz <code>break</code> – cyklus s podmínkou uprostřed	295
16.5	Cyklus s ukončovací podmínkou	296
16.6	Přiřazení v hlavičce cyklu	296
16.7	Větev <code>else</code>	298
16.8	Příkaz <code>continue</code>	299
16.9	Účel a syntaxe cyklu <code>for</code>	300
16.10	Vyjmenování hodnot parametru cyklu	301
16.11	Využití objektů typu <code>range</code>	302
16.11.1	Důvod použití větve <code>else</code>	302
16.12	Větev <code>else</code>	303
16.13	Použití indexů	304
16.14	Použití stringu jako zdroje	305
16.15	Využití objektů typu <code>enumerate</code>	305
16.16	Vnořování cyklů	306
16.17	Postupné použití několika zdrojů	306
16.18	Shrnutí kapitoly a doprovodné soubory	307
Kapitola 17 Tvorba aplikací		309
17.1	Vytvoření aplikace či knihovny	309
17.2	Přímé spuštění versus <code>import</code> modulu	309
17.2.1	Rozpoznání režimu, v němž byl modul spuštěn	310
17.2.2	Demonstrace rozpoznání způsobu spuštění	310
17.2.3	Spuštění modulu umístěného v balíčku	311
17.3	Argumenty příkazového řádku	312
17.4	Rozsáhlejší aplikace	315
17.4.1	Umístění kódu aplikace	316
17.4.2	Initor balíčku a změny záhlaví modulů balíčku	316
17.4.3	Modul <code>io</code> pro ošetření vstupu	316
17.4.4	Modul <code>game</code> s jádrem hry	318
17.4.5	Řídící modul <code>main</code>	319
17.5	Vytvoření spustitelné aplikace	320

17.5.1 Soubor typu <code>pyz</code>	321
17.5.2 Vytváříme aplikaci „ručně“	321
17.5.3 Alternativy	322
17.6 Shrnutí kapitoly a doprovodné soubory	322
Kapitola 18 Ošetřování chyb	323
18.1 Tři druhy chyb	323
18.1.1 Syntaktické chyby	323
18.1.2 Běhové chyby	324
18.1.3 Logické chyby	325
18.1.4 Varování	325
18.2 Chybové zprávy	325
18.2.1 Syntaktické chyby při práci v interaktivním režimu	326
18.2.2 Syntaktické chyby při překladu importovaného modulu	326
18.2.3 Běhové chyby	327
Odchylky v hlášeném prvotním zdroji	328
18.3 I chyby jsou objekty – výjimky	329
18.4 Rozdělení výjimek	330
18.4.1 Výjimky versus varování	331
18.5 Zachycení a ošetření výjimky	331
18.5.1 Více větví <code>except</code>	332
18.5.2 Větev <code>else</code>	333
18.5.3 Větev <code>finally</code>	333
18.5.4 Syntaktický diagram příkazu <code>try</code>	333
18.5.5 Příklad s kompletní verzí příkazu <code>try</code>	334
18.6 Praktický příklad	336
18.7 Zdánlivé záludnosti větve <code>finally</code>	336
18.8 Vyvolání výjimky	339
18.9 Problematika oddělených tisků	341
18.10 Příkaz <code>assert</code>	342
18.11 Jak vyvolat a zachytit varování	343
18.11.1 Vyvolání varovného hlášení	343
18.11.2 Zachycení varovného hlášení	344
18.11.3 Režimy zachytávání	345
18.12 Shrnutí kapitoly a doprovodné soubory	346

Část C Kontejnery

347

Kapitola 19 Seznamy	348
19.1 Proměnné a neměnné objekty	348
19.2 Základní informace o seznamech	349
19.3 Vytváření seznamů	350
19.3.1 Použití literálu	350
19.3.2 Využití konstruktoru <code>list()</code>	351
19.3.3 Sčítání a násobení	352
19.3.4 Generátorová notace seznamů	353
19.3.5 Tovární funkce a funkce vracející seznam	355
19.4 Modifikace seznamů	356
19.4.1 Metody <code>append()</code> a <code>extend()</code>	356
19.4.2 Rizika práce s odkazy na proměnné objekty	357
19.4.3 Postupné budování seznamu	358
19.4.4 Přičítání jiných zdrojů	359
19.4.5 Indexace prvků seznamu	360
19.4.6 Metody pracující s indexy	361
19.4.7 Metody pracující s celým seznamem	362

19.5	Vícerozměrné seznamy	363
19.6	Souhrnný příklad	364
19.7	Anotace odkazující na seznamy	365
19.8	Shrnutí kapitoly a doprovodné soubory	366
Kapitola 20	N-tice	367
20.1	Základní informace o n-ticích	367
20.2	Vytváření n-tic	368
20.2.1	Vytváření n-tic pomocí literálů	368
20.2.2	Využití konstruktoru <code>tuple()</code>	369
20.2.3	Sčítání a násobení	370
20.2.4	Přičítání n-tic	371
20.2.5	Balení a rozbíjení n-tic	372
20.2.6	Prohazování proměnných	372
20.2.7	Hvězdičkové pravidlo	373
20.3	Generátorová notace n-tic	373
20.4	Problematika neměnnosti n-tic	374
	Hešovatelné objekty	375
20.5	Přístup k prvkům n-tic	375
20.6	Sčítání seznamů a n-tic	376
20.7	Proměnné a neměnné prvky n-tice	377
20.8	Pojmenované n-tice	378
20.9	Anotace odkazující na n-tice	380
20.10	Shrnutí kapitoly a doprovodné soubory	381
Kapitola 21	Množiny	382
21.1	Základní informace o množinách	382
21.2	Vytváření množin	382
21.2.1	Vytváření množin pomocí literálů	383
21.2.2	Vytváření množin pomocí konstruktoru <code>set()</code>	383
21.2.3	Vytváření množin prostřednictvím množinových operací	385
21.3	Generátorová notace množin	388
21.4	Zmrazené množiny	388
21.5	Modifikace množin	389
21.5.1	Modifikace pracující s jedním prvkem	390
21.5.2	Množinové operátory a rozšířené operace	391
21.5.3	Porovnávání množin	393
21.6	Anotace odkazující na množiny	394
21.7	Shrnutí kapitoly a doprovodné soubory	394
Kapitola 22	Slovníky	395
22.1	Mapovací objekty a slovníky	395
22.2	Vytváření slovníků	396
22.2.1	Vytváření slovníků pomocí literálů	396
22.2.2	Vytváření slovníků pomocí konstruktoru <code>dict()</code>	397
22.2.3	Ekvivalence slovníků	398
22.2.4	Vytváření slovníků pomocí metody <code>fromkeys()</code>	398
22.3	Generátorová notace slovníků	399
22.4	Operace se slovníkem	401
22.4.1	Práce s hodnotami pomocí „indexace“ klíčem	401
22.4.2	Další metody pro práci s jednotlivými položkami	402
22.4.3	Modifikace slovníku daty ze zadaného zdroje	403
22.4.4	Slovník jako generátor	404
22.5	Pohledy	404
22.5.1	Pohledy jako zdroje dat	405
22.5.2	Operace s pohledy	406
22.6	Anotace odkazující na slovníky	407

22.7 Shrnutí kapitoly a doprovodné soubory.....	407
Kapitola 23 Rozšíření definic funkcí	409
23.1 Předávání argumentů odkazem a hodnotou	409
23.1.1 Dvě hodnoty odkazu.....	410
23.1.2 Proč jde o předávání hodnotou.....	410
23.2 Pomocná funkce <code>gr()</code>	412
23.3 Proměnný počet pozičních argumentů	412
23.3.1 Hvězdičkový parametr.....	413
23.3.2 Hvězdičkový argument.....	414
23.4 Proměnný počet pojmenovaných argumentů.....	416
23.4.1 Dvuhvězdičkový parametr.....	416
23.4.2 Dvuhvězdičkový argument.....	416
23.5 Stručný souhrn	417
23.5.1 Podivné chování	418
23.5.2 Použití v definicích literálů	418
23.6 Vestavěné funkce pracující s kontejnery	419
23.7 Funkce modulu <code>dbg</code> pracující s kontejnery.....	425
23.8 Shrnutí kapitoly a doprovodné soubory.....	426
Kapitola 24 Formátování stringů.....	428
24.1 Formátovací operátor %.....	428
24.2 Pokročilejší metody formátování	430
24.2.1 Metody <code>format()</code> a <code>format_map()</code> versus f-stringy	430
24.2.2 Formátovací string.....	431
24.2.3 Syntaxe nahrazovacího pole	432
24.2.4 Vyhodnocení nahrazovaného textu	432
24.2.5 Formátování nahrazujícího textu	434
24.2.6 Konverze	435
24.3 Specifikace formátu.....	436
24.3.1 Počet zabraných pozic	436
24.3.2 Přesnost.....	438
24.3.3 Typ hodnoty	438
24.3.4 Skupiny číslic	441
24.3.5 Alternativní formát a vedoucí nuly	442
24.3.6 Znaménko	443
24.3.7 Zarovnání a plnění	443
24.3.8 Vnořená nahrazovací pole ve specifikaci formátu	445
24.3.9 Formátování samodokumentujících se nahrazovacích polí.....	446
24.4 Příklad: Pascalův trojúhelník.....	446
24.5 Formátování instancí dalších typů.....	447
24.6 Šablony stringů.....	448
24.6.1 Instanční metody třídy <code>string.Template</code>	449
24.6.2 Ukázka použití stringových šablon.....	451
24.7 T-stringy jako šablony stringů	451
24.8 Shrnutí kapitoly a doprovodné soubory.....	452
Kapitola 25 Operace s kontejnery.....	453
25.1 Přítomnost prvku v kontejneru.....	453
25.2 Proměnné objekty jako implicitní hodnoty parametrů	454
25.3 Kopírování	454
Mělké a hluboké kopie objektů	454
25.3.1 Zdánlivé kopie neměnných kontejnerů	455
25.3.2 Alternativní způsob tvorby mělkých kopií.....	456
25.3.3 Nebezpečí hlubokých kopií.....	456
25.4 Rozdělení doposud probraných kontejnerů	456
25.5 Řazení prvků posloupnosti.....	457

25.5.1	reversed(seq)	458
25.5.2	sorted(iterable, *, key=None, reverse=False)	458
25.6	Vykrajování (slicing)	459
25.7	Indexování a vykrajování u rozsahů	460
25.8	Nahrazování hodnot	461
25.9	„Úprava“ neměnných objektů	463
25.10	Shrnutí kapitoly a doprovodné soubory	463
Kapitola 26	Práce se soubory	464
26.1	Soubory: bleskové opakování	464
26.1.1	Soubor, souborový systém, cesta	465
26.1.2	Absolutní a relativní cesta	466
26.1.3	Substituované disky ve Windows	466
26.2	Koncepce vstupu a výstupu dat v <i>Pythonu</i>	466
26.2.1	Starší koncepce souborů v jazycích C nebo Pascal	467
26.2.2	Novější koncepce datových proudů	467
26.2.3	Koncepce Pythonu	467
26.2.4	Shrnutí používané terminologie	468
26.3	Dva způsoby práce se souborovým systémem	469
26.4	Moduly os a os.path	471
26.5	Pracovní složka	472
26.6	Skládání a rozkládání cest	473
26.7	Vytváření a mazání složek	474
26.7.1	Mazání	475
26.8	Získání informací o souborech	476
26.9	Zápis a čtení dat	477
26.9.1	Problematika kódování ve Windows	477
26.9.2	Otevírání souborů a datových proudů	478
26.9.3	Zápis dat, splachování a zavírání proudů a přidružených souborů	480
26.10	Automatické zavírání proudů použitím with	482
26.11	Čtení ze souborů	483
26.12	Shrnutí kapitoly a doprovodné soubory	485
Literatura		487
Rejstřík		489

0 autorovi

Ing. Rudolf Pecinovský, CSc. studoval teoretickou kybernetiku na *Fakultě jaderné a fyzikálně inženýrské ČVUT* a poté technickou kybernetiku na *Fakultě Elektrotechnické ČVUT*, kterou ukončil v roce 1979. Titul CSc. získal v Ústavu teorie informace a automatizace ČSAV v roce 1983. Od počátku 80. let učí a publikuje, přičemž svůj výzkum soustředí především na oblast vstupních kurzů moderního programování a výuku objektivě orientovaného paradigmatu. V současné době učí na *Fakultě jaderné a fyzikálně inženýrské ČVUT* a na *Fakultě informatiky a statistiky Vysoké školy ekonomické v Praze*. Vedle toho učí ještě v řadě kurzů pro začátečníky i profesionální programátory. Doposud mu vyšlo přes 70 knih, které byly přeloženy do pěti jazyků. Většina jeho knih je zaměřena na výuku moderního programování a návrh objektivě orientované architektury.



Poděkování

Vím, že se v českých knížkách většinou neděkuje, ale tvorba knih je spojena s takovými oběťmi řady lidí z mého blízkého i vzdálenějšího okolí, že bych měl velkou újmu na duši, kdybych tak neučinil.

Chtěl bych především nesmírně poděkovat své ženě Jarušce, která byla po celou dobu mojí největší oporou a jejíž nekonečná trpělivost a vstřícnost mi pomohla dokončit knihu v termínu, který se příliš nelišil od toho, jež jsme původně s nakladatelem dohodli, a ne až někdy za rok po něm. Stále marně přemýšlím, kde má schovanou tu svatozář.

Původně jsem se domníval, že s dalším vydáním nebude moc práce. Šeredně jsem se zmýlil, protože veškeré úpravy, jež jsem do knihy zanesl, jsou vykoupeny hodinami studia a experimentování, které rodina s neuvěřitelnou trpělivostí snášela.

Na vylepšování textu nového vydání se podílela řada dalších lidí. Mezi nimi musím poděkovat především těm, kteří si dali práci s odhalováním případných chyb ve vznikajícím rukopisu. Mezi nimi pak především Ludkovi Šťastnému, který po celou přípravu rukopis pročetl a odhaloval v něm pasáže, jež by si zasloužily vylepšit. Neméně velkou zásluhu na současné podobě má i Jirka Kofránek, který mne průběžně upozorňoval na některé problémy s výukou podle běžně používaných postupů. Řadu podnětných myšlenek přinesl i Michal Palas, jenž pracuje v oblasti analýzy a zpracování dat a přispěl tak řadou poznatků z praxe.

V neposlední řadě patří můj dík redakci, především redaktoru Petru Somogyimu, který trpělivě snášel mé neustálé modifikace již zkorigovaného textu, a Radku Matulíkovi, který mne k napsání jednotlivých knih z posledních let vyhecoval a byl pak ochoten týden či dva počkat, když se mi nepodařilo přesně dodržet původně dohodnutý termín.