

# knihovna programátora

- Je psána jako učebnice i referenční příručka
- Probírá všechny konstrukce jazyka včetně těch běžně přeskakovaných
- Všechny probírané konstrukce demonstruje na příkladech
- Vysvětluje principy, na nichž je jazyk postaven, a jejichž znalost umožňuje lépe využívat jeho možnosti
- Pro výklad složitějších syntaktických konstrukcí používá syntaktické diagramy



RUDOLF PECINOVSKÝ

# Python

Kompletní příručka jazyka  
**PRO VERZI 3.9**



knihovna programátora

---

RUDOLF PECINOVSKÝ

# Python

**Kompletní příručka  
jazyka pro verzi 3.9**

GRADA  
Publishing

### **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**.

**Rudolf Pecinovský**

# **Python**

## **Kompletní příručka jazyka pro verzi 3.9**

Vydala Grada Publishing, a.s.

U Průhonu 22, Praha 7

obchod@grada.cz, www.grada.cz

tel.: +420 234 264 401

jako svou 7738. publikaci

Odpovědný redaktor Petr Somogyi

Fotografie na obálce Depositphotos/mario7

Grafická úprava a sazba Rudolf Pecinovský

Počet stran 480

První vydání, Praha 2020

Vytisklo Tisk Centrum s.r.o., Moravany u Brna

© Grada Publishing, a.s., 2020

Cover Design © Grada Publishing, a. s., 2020

Cover Photo © Depositphotos/mario7

*Názvy produktů, firem apod. použité v knize mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.*

ISBN 978-80-271-1852-6 (ePub)

ISBN 978-80-271-1851-9 (pdf)

ISBN 978-80-271-1269-2 (print)

*Všem, kteří se chtějí něco naučit*

# Stručný obsah

Úvod .....	22
<b>Část A Základy</b> .....	<b>29</b>
1 Startujeme .....	30
2 Zadávání jednoduchých hodnot .....	41
3 Jednoduché výrazy .....	53
4 <b>Proměnné</b> .....	77
5 Logické hodnoty a operace .....	91
6 <b>Jednoduché příkazy</b> .....	99
7 Moduly .....	108
8 Definice funkcí .....	132
9 <b>Pokročilé rysy funkcí</b> .....	153
10 Rozhodování a cykly .....	166
11 Cyklus for a generátory .....	181
12 <b>Ošetřování chyb</b> .....	191
<b>Část B Kontejnery</b> .....	<b>205</b>
13 Seznamy .....	206
14 N-tice .....	221
15 Množiny .....	232
16 Slovníky .....	243
17 <b>Rozšíření definic funkcí</b> .....	255
18 Operace se stringy a formátování .....	267
19 Operace s kontejnery .....	287
20 Práce se soubory .....	298

---

**Část C Objektově orientované programování** 313

---

21	Základy OOP .....	314
22	Třídy a jejich instance .....	326
23	Jednoduché dědění .....	346
24	Násobné dědění .....	357
25	Přetěžování operátorů .....	370
26	Dekorátory .....	389
27	Deskriptory, vlastnosti a sloty .....	402
28	Balíčky .....	415
29	Generátory a iterátory .....	427
30	Ovlivnění tvorby tříd, metatřídy .....	437
31	Korutiny, vlákna, procesy .....	454

---

**Část D Přílohy** 463

---

A	Konfigurace ve Windows .....	464
B	Syntaktické diagramy .....	467
C	Konvence pro psaní programů v Pythonu .....	469
	Literatura .....	472
	Rejstřík .....	473

# Podrobný obsah

Úvod .....	22
Komu je kniha určena .....	22
<b>Struktura příručky</b> .....	23
Koncepte výkladu .....	24
Jazyk identifikátorů .....	25
<b>Potřebné vybavení</b> .....	25
Doprovodné programy .....	25
Použité typografické konvence .....	26
Odbočka – podšeděný blok .....	28
Zpětná vazba .....	28
<b>Část A Základy</b> .....	<b>29</b>
<b>1 Startujeme</b> .....	<b>30</b>
<b>1.1 Hlavní součást instalace</b> .....	<b>30</b>
Platforma .....	30
Dokumentace .....	31
Pracovní režimy .....	32
Vývojové prostředí .....	32
<b>1.2 IDLE – seznámte se</b> .....	<b>33</b>
<b>Spuštění</b> .....	<b>33</b>
Základní popis .....	34
<b>Příkazové okno</b> .....	<b>35</b>
Restart interaktivního systému .....	35
Uložení záznamu seance .....	35
<b>Editační okno</b> .....	<b>36</b>
<b>Umístění editovaných souborů</b> .....	<b>36</b>
<b>Barevné zvýraznění textu</b> .....	<b>36</b>
Použité písmo .....	37
<b>1.3 Objekty a objektové programování</b> .....	<b>37</b>
<b>Objekt, třída, instance</b> .....	<b>38</b>
Objekt .....	38
Třída .....	38
Instance .....	38
<b>1.4 Nejdůležitější zvláštnosti Pythonu</b> .....	<b>38</b>
Přísné a benevolentní programovací jazyky .....	39
<b>2 Zadávání jednoduchých hodnot</b> .....	<b>41</b>
<b>2.1 Zápis celých a desetinných (reálných) čísel</b> .....	<b>41</b>
Zpřehlednění dlouhých čísel pomocí znaku podtržení .....	42
<b>2.2 Komplexní čísla</b> .....	<b>43</b>
<b>2.3 Počáteční nula</b> .....	<b>44</b>
<b>2.4 Zadávání čísel v jiných číselných soustavách</b> .....	<b>44</b>
<b>2.5 Platí – neplatí</b> .....	<b>45</b>



2.6	Zadávání textů.....	45
	Escape sekvence .....	47
	Bílé znaky .....	49
2.7	Nic – None .....	49
2.8	Výpustka – Ellipsis, ... ..	49
2.9	Literály .....	50
2.10	Důležitost přehlednosti .....	50
2.11	Komentáře .....	51
2.12	Shrnutí .....	52
3	Jednoduché výrazy .....	53
3.1	Trocha teorie .....	53
	Operace .....	53
	Operand .....	53
	Operátor .....	54
	Arita operátorů .....	54
	Priorita operátorů .....	56
	Asociativita binárních operátorů .....	56
3.2	Volání funkčního objektu .....	57
	Parametr versus argument .....	58
	Syntaxe volání a návratová hodnota .....	58
	Pořadí vyhodnocování a předávání argumentů .....	59
	Datové × funkční objekty .....	59
3.3	Vestavěné funkce s jednoduchými argumenty .....	59
	Operátory jako funkční objekty .....	60
	Zápis syntaxe .....	60
	Přehled vestavěných funkcí s argumenty jednoduchých typů .....	60
	abs(x) .....	60
	ascii(objekt) .....	61
	bin(x) .....	61
	bool(x) .....	61
	complex(/real/, imag//) .....	61
	divmod(a, b) .....	61
	eval(object) .....	61
	exec(object) .....	61
	float(/x/) .....	62
	hash(object) .....	62
	help(/objekt/) .....	62
	hex(i) .....	62
	chr(i) .....	62
	id(object) .....	62
	input(/výzva/) .....	62
	int(/x/) .....	63
	int(s, base) .....	63
	len(s) .....	63
	max(arg1, arg2 ...) min(arg1, arg2 ...) .....	64
	oct(i) .....	64
	ord(c) .....	64
	pow(x, y/, z/) .....	64
	print(argumenty) .....	64
	range(stop) range(start, stop/, step/) .....	65
	repr(object) .....	65
	round(number/, ndigits) .....	65
	str(object) .....	66
	type(object) .....	66

3.4 Získání nápovědy.....	66
Argument zadán .....	66
Nápověda k některým operátorům a konstrukcím jazyka .....	67
Bez argumentu.....	68
3.5 Numerické operace.....	68
Tři druhy dělení.....	68
Umocňování .....	70
Nekonečna a nesmyslná čísla.....	71
3.6 Bitové posuny.....	72
3.7 Operace s texty.....	72
Sčítání textů .....	72
Násobení textů.....	73
Indexace jednotlivých znaků .....	74
3.8 Zadání na více řádcích .....	74
3.9 Shrnutí .....	76
<b>4 Proměnné.....</b>	<b>77</b>
4.1 Co jsou to proměnné.....	77
Statické a dynamické typování .....	78
4.2 Správa paměti .....	79
4.3 Pravidla pro tvorbu identifikátorů .....	79
4.4 Zavedení proměnné.....	80
Zadání skupiny hodnot .....	81
Proměnné inf a nan .....	83
4.5 Vnořená volání funkčních objektů .....	83
4.6 Zjištění typu objektu v proměnné .....	85
4.7 Uložení funkce do proměnné.....	86
4.8 Lambda-výrazy.....	86
Vnořování volání funkčních objektů.....	87
4.9 Datové a funkční proměnné.....	88
4.10 Mezery ve výrazech a příkazech.....	88
4.11 Uložení do proměnné x propojení s názvem.....	89
4.12 Přířazovací výraz .....	89
4.13 Pomocné proměnné .....	90
4.14 Shrnutí .....	90
<b>5 Logické hodnoty a operace.....</b>	<b>91</b>
5.1 Konstanty True a False .....	91
5.2 Převod jiných hodnot na logické .....	92
5.3 Porovnávání hodnot .....	92
Porovnání reálných čísel .....	93
Porovnávání textů .....	93
Zřetěžené porovnávání .....	94
Porovnávání totožnosti objektů .....	94
5.4 Logické operátory a operace.....	95
5.5 Operace s jednotlivými bity .....	96
5.6 Podmíněný výraz .....	97
5.7 Shrnutí .....	98
<b>6 Jednoduché příkazy.....</b>	<b>99</b>
6.1 Příkaz pass.....	99
6.2 Příkaz tvořený výrazem, výrazový příkaz .....	99
6.3 Několik příkazů na řádku .....	100
6.4 Přířazovací příkaz .....	100
Složená přiřazení.....	100
6.5 Příkaz del.....	101

6.6	<b>Příkaz <code>assert</code></b> .....	102
	Návrh podle kontraktu.....	103
6.7	<b>Složené příkazy a odsazování</b> .....	105
	Výhody a nevýhody koncepce Pythonu .....	106
6.8	<b>Shrnutí</b> .....	107
7	<b>Moduly</b> .....	108
7.1	<b>Trocha teorie – základní informace</b> .....	108
	Atributy a metody .....	108
	Práce s objekty – kvalifikace .....	109
	<b>Vše je součástí nějakého modulu</b> .....	109
	<b>Dva názvy objektů</b> .....	110
	Zdrojový soubor.....	110
	Přeložený soubor.....	110
7.2	<b>Příkaz <code>import</code></b> .....	111
	Čistý import jiného modulu .....	111
	Import modulu pod jiným názvem .....	112
	Přímý import vyjmenovaných objektů .....	114
	Import objektů modulu nezahrnuje import modulu.....	115
	Argumentem příkazů <code>import</code> a <code>from ... import</code> nesmí být výraz.....	115
	Import všech objektů.....	115
	Syntaktické diagramy příkazu <code>import</code> .....	116
7.3	<b>Modul jako objekt</b> .....	116
	Modul <code>builtins</code> a zdánlivě neobjektové programování .....	116
7.4	<b>Postup systému při importu modulu</b> .....	118
	Prohledávané složky .....	118
7.5	<b>Vytvoření vlastního modulu</b> .....	119
	Kódová stránka .....	121
	Dokumentační komentář .....	121
	Zadané příkazy .....	122
	Informace o načítání modulu .....	122
	Definice datových atributů .....	122
	Neveřejné atributy .....	122
	Výrazové příkazy .....	123
7.6	<b>Průběh importu vytvořeného modulu</b> .....	123
	Použitelné názvy .....	125
	Neveřejné atributy.....	125
7.7	<b>Reimport již importovaného modulu</b> .....	126
	<b>Pozor na přímo importované proměnné</b> .....	127
	Specifika funkce <code>importlib.reload()</code> .....	128
	Rozbor chybového hlášení .....	129
	Důsledky chybného zavedení modulu.....	129
	Syntaktické chyby.....	130
7.8	<b>Zprostředkovaný import</b> .....	130
7.9	<b>Shrnutí</b> .....	131
8	<b>Definice funkcí</b> .....	132
8.1	<b>Definice funkce</b> .....	133
8.2	<b>Definice vlastní funkce</b> .....	133
8.3	<b>Přesun definice složeného příkazu k levému okraji</b> .....	136
8.4	<b>Kdy se projeví chyby v definici funkce</b> .....	136
	Volná proměnná.....	138
8.5	<b>Parametry, argumenty a lokální proměnné</b> .....	138
	Definice .....	138
	<b>Lokální proměnné</b> .....	139
	Volání funkcí s parametry .....	140
	Povinně pojmenované argumenty .....	141

Povinně poziční argumenty.....	141
Mix povinností .....	143
8.6 Implicitní hodnoty argumentů .....	144
8.7 Konstantnost předdefinovaných hodnot .....	146
8.8 Funkce s vedlejším efektem .....	147
8.9 Funkce vracející hodnotu a příkaz <code>return</code> .....	147
8.10 Přetěžování funkcí .....	148
Něco přetížít jde .....	149
8.11 Použití anotací .....	149
8.12 Definice funkcí v modulu .....	151
8.13 Shrnutí .....	152
<b>9 Pokročilé rysy funkcí .....</b>	<b>153</b>
9.1 Vnitřní funkce .....	153
Odsazování .....	154
9.2 IDLE a nastavení Show Code Context .....	154
9.3 Jmenné prostory .....	155
9.4 Oblast/rozsah platnosti, působnost (scope) .....	156
Zanoření jmenových prostorů .....	156
9.5 Lokality použitých proměnných .....	157
Příkaz <code>global</code> .....	158
Příkaz <code>nonlocal</code> .....	159
9.6 Vnoření funkce versus vnoření volání funkcí .....	161
9.7 Vnořená volání funkcí .....	162
9.8 Funkce vyššího řádu .....	162
9.9 Atributy funkcí .....	164
9.10 Nelokální proměnné a uzávěry (closures) .....	164
9.11 Další vlastnosti funkcí .....	165
9.12 Shrnutí .....	165
<b>10 Rozhodování a cykly .....</b>	<b>166</b>
10.1 Podmíněný příkaz .....	166
Jednoduchý podmíněný příkaz .....	166
Větev <code>else</code> .....	167
Rozhodování s více větvemi: rozšířený podmíněný příkaz .....	168
10.2 Kontrolní tisky .....	170
Alternativní postup .....	170
Dokonalejší postup .....	172
10.3 Rekurze .....	172
Zásobník návratových adres – ZNA .....	174
10.4 Příkaz <code>while</code> – cyklus se vstupní podmínkou .....	174
10.5 Nekonečný cyklus .....	175
10.6 Příkaz <code>break</code> – cyklus s podmínkou uprostřed .....	176
10.7 Cyklus s ukončovací podmínkou .....	177
10.8 Přiřazení v hlavičce cyklu .....	177
10.9 Větev <code>else</code> .....	179
10.10 Příkaz <code>continue</code> .....	179
10.11 Shrnutí .....	180
<b>11 Cyklus <code>for</code> a generátory .....</b>	<b>181</b>
11.1 Účel a syntaxe cyklu <code>for</code> .....	181
11.2 Vyjmenování hodnot parametru cyklu .....	182
11.3 Využití funkce <code>range()</code> .....	183
Použití indexů .....	183
11.4 Použití stringu jako zdroje .....	184

11.5	Vnořování cyklů .....	184
11.6	Postupné použití několika zdrojů .....	185
11.7	Definice generátorové funkce jako zdroje .....	186
11.8	Větev <code>else</code> .....	187
11.9	Problémy s odsazením v interaktivním režimu .....	188
11.10	Modul s několika užitečnými funkcemi – <code>utils</code> .....	190
11.11	Shrnutí .....	190
12	<b>Ošetřování chyb</b> .....	191
12.1	<b>Tři druhy chyb</b> .....	191
	Syntaktické chyby .....	191
	Běhové chyby .....	192
	Logické chyby .....	192
12.2	Chybové zprávy .....	193
	Syntaktické chyby při interpretaci příkazu v interaktivním režimu .....	193
	Syntaktické chyby při zadávání příkazu .....	193
	Syntaktické chyby při překladu importovaného modulu .....	194
	Běhové chyby .....	194
12.3	I chyby jsou objekty – výjimky .....	196
12.4	Rozdělení výjimek .....	197
12.5	Zachycení a ošetření výjimky .....	197
12.6	Více větví <code>except</code> .....	199
12.7	Větev <code>else</code> .....	199
12.8	Větev <code>finally</code> .....	199
12.9	Syntaktický diagram příkazu <code>try</code> .....	199
12.10	Příklad s kompletní verzí příkazu <code>try</code> .....	200
	Převod se nepodařil .....	200
	Převod se podařil .....	202
12.11	Praktický příklad .....	202
12.12	Vyhození vlastní výjimky .....	202
12.13	Příkaz <code>assert</code> .....	204
12.14	Hierarchie výjimek a definice vlastní výjimky .....	204
12.15	Shrnutí .....	204

## Část B Kontejnery

205

13	<b>Seznamy</b> .....	206
13.1	Proměnné a neměnné objekty .....	206
13.2	Tvorba instancí a konstruktory .....	207
13.3	Základní informace o seznamech .....	208
13.4	<b>Vytváření seznamů</b> .....	208
	Použití literálu .....	208
	Využití konstruktoru <code>list(seq=())</code> .....	209
	Sčítání a násobení .....	210
13.5	Generátorová notace seznamů .....	211
13.6	<b>Modifikace seznamů</b> .....	212
	Metody <code>append()</code> a <code>extend()</code> .....	213
	Rizika práce s odkazy na proměnné objekty .....	214
	Postupné budování seznamu .....	214
	Přičítání jiných zdrojů .....	215
	Indexace prvků seznamu .....	216
	Metody pracující s indexy .....	216
	<code>index(value, start=0, stop=9223372036854775807, /) -&gt; int</code> .....	217
	<code>insert(index, object, /)</code> .....	217

pop(index=-1, /) -> ? .....	217
remove(self, value, /) .....	217
Metody pracující s celým seznamem .....	218
reverse(self, /) .....	218
sort(*, key=None, reverse=False) .....	218
13.7 Vícerozměrné seznamy .....	219
13.8 Souhrnný příklad .....	220
13.9 Shrnutí .....	220
14 N-tice .....	221
14.1 Základní informace o n-ticích .....	221
14.2 Vytváření n-tic .....	221
Vytváření n-tic pomocí literálů .....	222
Využití konstruktora tuple(seq=()) .....	223
Sčítání a násobení .....	224
Přičítání n-tic .....	225
Balení a rozbalování n-tic .....	226
Prohazování proměnných .....	226
Hvězdičkové pravidlo .....	227
14.3 Generátorová notace n-tic .....	228
14.4 Problematika neměnnosti n-tic .....	228
Hešovatelné objekty .....	229
14.5 Přístup k prvkům n-tic .....	229
14.6 Proměnné a neměnné prvky n-tice .....	230
14.7 Sčítání seznamů a n-tic .....	231
14.8 Shrnutí .....	231
15 Množiny .....	232
15.1 Základní informace o množinách .....	232
15.2 Vytváření množin .....	232
Vytváření množin pomocí literálů .....	233
Vytváření množin pomocí konstrukturu set() .....	233
Hešová tabulka .....	233
Další použitelné zdroje .....	234
Vytváření množin prostřednictvím množinových operací .....	234
union(*zdroj) a   b   ... .....	235
intersection(*zdroj) a & b & ... .....	235
difference(*zdroj) a - b - ... .....	235
symmetric_difference(zdroj) a ^ b .....	235
15.3 Generátorová notace množin .....	236
15.4 Zmrazené množiny .....	237
15.5 Modifikace množin .....	238
Modifikace pracující s jedním prvkem .....	238
add(element) .....	239
discard(element) .....	239
remove(element) .....	239
pop() .....	239
Množinové operátory a sdružené operace .....	239
update(*zdroj) a  = b   ... .....	239
intersection_update(*zdroj) a &= b & ... .....	239
difference_update(*zdroj) a -= b   ... .....	240
symmetric_difference_update(zdroj) a ^= b .....	240
Porovnávání množin .....	241
isdisjoint(množina) .....	241
issubset(množina) a <= b .....	241
a < b .....	241

issuperset (množina) a >= b .....	241
a > b .....	242
15.6 Shrnutí .....	242
<b>16 Slovníky .....</b>	<b>243</b>
16.1 Mapovací objekty a slovníky .....	243
16.2 Vytváření slovníků .....	244
Vytváření slovníků pomocí literálů .....	244
Vytváření slovníků pomocí konstruktoru dict() .....	245
Ekvivalence slovníků .....	246
Vytváření slovníků pomocí metody fromkeys() .....	246
16.3 Generátorová notace slovníků .....	247
16.4 Operace se slovníkem .....	248
Práce s hodnotami pomocí „indexace“ klíčem .....	248
Další metody pro práci s jednotlivými položkami .....	249
get(key, default=None, \) .....	249
pop(key, default=#, \) .....	250
popitem() .....	250
setdefault(key, default=None, \) .....	250
Modifikace slovníku daty ze zadaného zdroje .....	251
update(zdroj) .....	251
Operátor   .....	251
Sdružené přiřazení  = .....	251
Slovník jako generátor .....	251
16.5 Pohledy .....	251
items() .....	251
keys() .....	252
values() .....	252
Pohledy jako generátory .....	252
Operace s pohledy .....	253
16.6 Shrnutí .....	254
<b>17 Rozšíření definic funkcí .....</b>	<b>255</b>
17.1 Předávání argumentů odkazem a hodnotou .....	255
Předání argumentu odkazem .....	256
17.2 Proměnný počet pozičních argumentů .....	257
Hvězdičkový parametr .....	257
Hvězdičkový argument .....	259
17.3 Proměnný počet pojmenovaných argumentů .....	260
Dvuhvězdičkový parametr .....	260
Dvuhvězdičkový argument .....	260
17.4 Stručný souhrn .....	262
17.5 Vestavěné funkce pracující s kontejnery .....	262
all(iterable) .....	262
any(iterable) .....	262
dir(object) .....	262
enumerate(iterable, start=0) .....	263
eval(expression, globals=None, locals=None) .....	263
filter(function, iterable) .....	263
globals() .....	264
len(c) .....	264
locals() .....	264
max(iterable, */, key, default/) .....	264
max(arg1, arg2, *args/, key/) .....	264
min(iterable, */, key, default/) .....	265
min(arg1, arg2, *args/, key/) .....	265

reversed(seq) .....	265
slice(stop) slice(start, stop[, step]) .....	265
sorted(iterable, *, key=None, reverse=False) .....	265
sum(iterable, /, start=0) .....	266
vars(object) .....	266
zip(*iterables) .....	266
17.6 Shrnutí .....	266
18 Operace se stringy a formátování .....	267
18.1 Prefixy stringových literálů .....	267
18.2 Formátovací operátor % .....	268
18.3 Metoda format() .....	270
Formátovací string .....	270
Konverze .....	272
Specifikace formátu .....	272
Počet zabraných pozic .....	272
Přesnost .....	273
Typ .....	274
Skupiny .....	276
Alternativní formát a vedoucí nuly .....	276
Znaménko .....	277
Zarovnání a plnění .....	278
Vnořená nahrazovací pole .....	279
18.4 Stringová interpolace – f-stringy .....	280
Samodokumentující se výrazy .....	281
18.5 Příklad: Pascalův trojúhelník .....	282
18.6 Příklad: Trasovací funkce prSE() .....	284
18.7 Shrnutí .....	286
19 Operace s kontejnery .....	287
19.1 Proměnné objekty jako implicitní hodnoty parametrů .....	287
19.2 Kopírování .....	287
Mělké a hluboké kopie objektů .....	288
Alternativní způsob tvorby mělkých kopií .....	289
Nebezpečí hlubokých kopií .....	289
19.3 Rozdělení doposud probraných kontejnerů .....	290
19.4 Přítomnost prvku v kontejneru .....	291
19.5 Řazení prvků posloupnosti .....	291
reversed(seq) .....	291
sorted(iterable, *, key=None, reverse=False) .....	292
19.6 Vykrajování (slicing) .....	293
19.7 Indexování a vykrajování u rozsahů .....	294
19.8 Nahrazování hodnot .....	295
19.9 „Úprava“ neměnných objektů .....	296
19.10 Shrnutí .....	297
20 Práce se soubory .....	298
20.1 Soubory: bleskové opakování .....	298
Soubor, souborový systém, cesta .....	299
Absolutní a relativní cesta .....	300
Substituované disky ve Windows .....	300
20.2 Dva způsoby práce se souborovým systémem .....	300
20.3 Moduly os a os.path .....	301
20.4 Pracovní složka .....	303
20.5 Skládání a rozkládání cest .....	304
20.6 Získání informací o souborech .....	305



20.7 Vytváření složek.....	306
20.8 Zápis a čtení dat.....	307
Otevírání souborů.....	307
Zápis dat, splachování a zavírání souborů.....	309
20.9 Čtení ze souborů.....	310
20.10 Shrnutí.....	312

## **Část C Objektově orientované programování** **313**

<b>21 Základy OOP.....</b>	<b>314</b>
21.1 Základní princip OOP.....	315
21.2 Objekty a jejich atributy.....	316
Terminologická vsuvka.....	317
21.3 Třídy a jejich instance.....	317
Třída versus datový typ.....	318
Instance.....	318
21.4 Objekt třídy versus instance třídy.....	319
21.5 Atributy třídy versus atributy instancí.....	319
21.6 Zprávy.....	320
21.7 Metody.....	320
21.8 Dědění.....	321
Terminologie.....	321
LSP – substituční princip Liskové.....	321
Virtuální metody a jejich přebíjení.....	322
Zakrývání versus přebíjení.....	322
Polymorfismus.....	323
Rodičovský podobjekt.....	323
Násobné dědění a diamantový problém.....	323
Zobecňování.....	325
21.9 Shrnutí.....	325
<b>22 Třídy a jejich instance.....</b>	<b>326</b>
22.1 Definice třídy a jejích atributů.....	326
22.2 Práce s atributy objektu.....	328
Získání a modifikace hodnoty atributu.....	328
Přidání a odebrání atributu.....	329
Třída jako parametr.....	330
22.3 Instance a práce s nimi.....	330
Tři druhy metod.....	331
Metody instancí a parametr <code>self</code> .....	332
Vytváření instancí – konstruktor, alokátor, initor.....	333
Instanční metody třídy C2.....	335
Použití instančních a třídnicích atributů.....	336
Změny instančních a třídnicích datových atributů.....	337
Zavádění nových třídnicích funkčních atributů.....	339
Zavádění nových instančních funkčních atributů.....	340
22.4 Vestavěné třídy jsou nemodifikovatelné.....	341
22.5 Nutnost kvalifikace atributů.....	341
22.6 Speciální atributy.....	342
<code>__bases__</code> .....	343
<code>__class__</code> .....	343
<code>__dict__</code> .....	343
<code>__doc__</code> .....	343
<code>__mro__</code> .....	343
<code>__name__</code> .....	343

__qualname__ .....	343
__subclasses__() .....	343
Ukázky použití .....	343
22.7 Alternativní definice třídy .....	344
22.8 Shrnutí .....	345
<b>23 Jednoduché dědění .....</b>	<b>346</b>
23.1 Vytváření potomka a jeho vlastnosti .....	346
23.2 Příklad .....	347
Třída LA .....	348
Třídy LB a LC .....	349
23.3 Jmenné prostory .....	349
23.4 Používání třídních atributů .....	350
23.5 Volání metod .....	351
23.6 Initory se dědí .....	352
23.7 Hierarchie výjimek .....	353
23.8 Definice vlastní výjimky .....	354
23.9 Vestavěné funkce pracující s objekty .....	354
callable(object) .....	355
classmethod(method, *args, **kwargs) .....	355
delattr(object, name) .....	355
getattr(object, name[, default]) .....	355
hasattr(object, name) .....	355
isinstance(object, classinfo) .....	355
issubclass(class, classinfo) .....	355
setattr(object, name, value) .....	355
staticmethod(method, *args, **kwargs) .....	355
super([type/, object-or-type//]) .....	356
23.10 Shrnutí .....	356
<b>24 Násobné dědění .....</b>	<b>357</b>
24.1 Python a násobné dědění .....	357
24.2 Příklad: jednoduchý diamant .....	358
24.3 Analýza chování .....	359
Přímé zadání volaného initoru .....	360
Problémy s parametry a argumenty – hubnoucí parametr .....	362
24.4 Virtuální metody .....	364
24.5 Složitější hierarchie dědění a MRO .....	364
Nerealizovatelná a následně opravená hierarchie .....	364
Zásady specifikace MRO .....	365
24.6 Komolení jmen a pseudosoukromé atributy .....	366
24.7 Šablonová metoda .....	367
24.8 Shrnutí .....	369
<b>25 Přetěžování operátorů .....</b>	<b>370</b>
25.1 Základy .....	370
Neimplementované operace – <code>NotImplemented</code> .....	371
25.2 Základní sada .....	371
__bool__(self) .....	371
__del__() .....	371
__lt__(self, other) __le__(self, other) __eq__(self, other)	
__ne__(self, other) __gt__(self, other) __ge__(self, other) .....	371
__hash__(self) .....	372
__format__(self, format_spec) .....	372
__repr__(self), __str__(self) .....	372

25.3	Operátory + - * @ / // % ** << >> & ^   a emulace numerických typů .....	372
	Binární operátory .....	373
	Unární operátory .....	373
	Zbylé emulační funkce .....	374
	Konverzní funkce .....	374
	Souhrnný příklad .....	375
25.4	Operátor [] a emulace kontejneru .....	377
	__contains__(self, item) .....	377
	__delitem__(self, key) .....	377
	__getitem__(self, key) .....	377
	__iter__(self) .....	377
	__len__(self) .....	377
	__missing__(self, key) .....	377
	__reversed__(self) .....	378
	__setitem__(self, key, value) .....	378
	Příklad .....	378
25.5	Operátor () – volatelné objekty .....	378
	Příklad .....	379
25.6	Operátor . (tečka) – přístup k atributům .....	380
	__getattr__(self, name) .....	380
	__getattr__(self, name) .....	380
	__setattr__(self, name, value) .....	381
	__delattr__(self, name) .....	381
	__dir__(self) .....	381
	Příklad – definice .....	382
	Příklad – prověrka .....	382
	Aktivita IDLE .....	382
	Další příkazy .....	383
	Lepší řešení .....	385
25.7	Správce kontextu a příkaz with .....	386
	Příklad .....	388
25.8	Shrnutí .....	388
26	Dekorátory .....	389
26.1	Co jsou dekorátory .....	389
	„Operátor“ @ .....	390
26.2	Použití dekorátorů staticmethod a classmethod .....	390
	Instanční versus třídní versus statické metody .....	390
	Příklad použití .....	392
	Příklad s třídní metodou .....	392
	Jednoduchá tovární metoda .....	393
26.3	Tvorba vlastního dekorátoru .....	394
26.4	Dekorátory s parametry .....	397
26.5	Současné použití více dekorátorů .....	397
26.6	Dekorátory třídy .....	398
	Jedináček – singleton .....	398
26.7	Složitější dekorátor – datové třídy .....	400
26.8	Shrnutí .....	401
27	Deskriptory, vlastnosti a sloty .....	402
27.1	Co jsou deskriptory .....	402
	__get__(self, instance, owner=None) .....	402
	__set__(self, instance, value) .....	402
	__delete__(self, instance) .....	403
	__set_name__(self, owner, name) .....	403
	Dělení deskriptorů .....	403

27.2	Příklad	403
27.3	Deskriptor vytvářený vnitřní třídou	405
27.4	Příklad: deskriptor pro odložené vyhodnocení	407
27.5	Co jsou to vlastnosti	408
	Přímá definice vlastnosti	408
	Definice ekvivalentu třídy <code>property</code>	409
27.6	Zadání vlastnosti pomocí dekorátoru	410
27.7	Jaký způsob správy přístupu k atributu zvolit	412
27.8	Sloty	412
	Slabé odkazy (weak references)	413
27.9	Shrnutí	414
<b>28</b>	<b>Balíčky</b>	<b>415</b>
28.1	Balíčky (packages)	415
	Initor balíčku	415
	Implementace	416
	Úpravy modulu v balíčku a jeho opětovné načtení	417
	Nedostupnost neimportovaných balíčků	418
28.2	Relativní import	418
28.3	Vytvoření aplikace či knihovny	419
28.4	Přímé spuštění zadaného skriptu	420
	Rozpoznání režimu, v němž byl modul spuštěn	420
	Demonstrace	420
28.5	Vytvoření spustitelné aplikace	421
	Soubor typu <code>pyz</code>	422
	Vytváříme aplikaci „ručně“	423
28.6	Argumenty příkazového řádku	424
	Zpracování dvousložkového argumentu	425
	Alternativy	426
28.7	Shrnutí	426
<b>29</b>	<b>Generátory a iterátory</b>	<b>427</b>
29.1	Iterovatelné objekty	427
29.2	Iterátory	428
	<code>__iter__()</code>	428
	<code>__next__()</code>	428
29.3	Generátorová funkce a generátorový iterátor	428
29.4	Generátorový výraz	429
29.5	Vestavěné funkce související s iteracemi	429
	<code>iter(object[, sentinel])</code>	429
	<code>map(function, iterable, ...)</code>	429
	<code>next(iterator[, default])</code>	430
29.6	Jak to celé funguje	430
29.7	Operátor <code>yield</code> jako výraz	430
	Demonstrační příklad chování výrazu <code>yield</code>	431
	Demonstrační příklad použití výrazu <code>yield</code>	433
	Funkce <code>cyklus()</code>	433
	Generátorová funkce <code>gen()</code>	434
29.8	Výraz/příkaz <code>yield from</code>	435
29.9	Shrnutí	436
<b>30</b>	<b>Ovlivnění tvorby tříd, metatříd</b>	<b>437</b>
30.1	Ovlivnění tvorby dceřiných tříd	437
	Definice třídy <code>C30a</code>	437
	Zbytek výpisu 30.1	439
	Zákaz definice potomků dané třídy	439