Obsah

Úvod **11**

Jak tato kniha vznikla ... 11
Co můžete od knížky očekávat ... 12
Jak jsou organizovány programy ... 13
Poděkování ... 14
Kde hledat nejnovější informace ... 14
Typografické a syntaktické konvence ... 15

1. Pole **16**

1.1. Podpora práce s poli – třída `Arrays` ... 16
1.1.1. Základní metody třídy `Arrays` ... 17
1.2. Řazení objektů ... 19
1.2.1. Přirozené řazení ... 19
1.2.2. Absolutní řazení ... 23
1.2.3. Řazení akcentovaných řetězců ... 27
1.2.4. Ideální porovnatelná třída ... 29
1.3. Binární vyhledávání ... 31
1.4. Práce s částí pole ... 33
1.5. Méně časté operace s poli ... 35
1.5.1. Kopírování pole – `System.arraycopy()` ... 35
1.5.2. Metoda `asList()` ... 38
1.5.3. Metoda `equals()` ... 38

2. Kolekce (kontejnery) **43**

2.1. Hlavní výhody používání kolekcí ... 43
2.2. Základní typy kolekcí ... 44

3. Společné metody seznamů a množin 48
3.1. Rozhraní `Collection` ... 48
3.1.1. Metody pro plnění kolekce ... 48
3.1.2. Metody pro ubírání z kolekce ... 48
3.1.3. Dynamické vlastnosti kolekcí ... 49
3.1.4. Získání přístupového objektu ... 49
3.1.5. Převod kolekce na běžné pole ... 49
3.2. Rozhraní `List` ... 49
3.2.1. Změny v kolekci ... 49
3.2.2. Získání obsahu kolekce ... 50
3.3. Rozhraní `Set` ... 50

4. Třída `ArrayList` 51
4.1. Velikost versus kapacita ... 51
4.2. Základní dovednosti s `ArrayList` ... 53
4.2.1. Metody z rozhraní `Collection` ... 53
4.2.2. Metody z rozhraní `List` – využití indexů ... 55
4.3. Vlastní třída jako prvek seznamu ... 56
4.4. Problémy při práci s podseznamy ... 58
4.5. Převod seznamu nebo množiny na běžné pole objektů ... 60
4.6. Převod běžného pole na kolekci ... 62
4.6.1. Seznam pevné velikosti ... 62

5. Třída `Collections` 64
5.1. Nejčastěji používané metody třídy `Collections` ... 64
5.1.1. Plnění seznamu ... 64
5.1.2. Řazení seznamu ... 65
5.1.3. Binární vyhledávání v seřazeném seznamu ... 65
5.1.4. Hledání mezních hodnot ... 65
5.1.5. Otočení pořadí seznamu ... 66
5.1.6. Zamíchání seznamu ... 66
5.2. Méně známé metody z `Collections` ... 68
5.2.1. Hledání podseznamu ... 68
5.2.2. Náhrada prvků seznamu ... 69
5.2.3. Rotace seznamu ... 69
5.2.4. Výměna prvků ... 70

6. Problém typové anonymity prvků v kolekci 71
6.1. Sjednocení typově rozdílných prvků pomocí rozhraní ... 73
6.2. Kolekce pro jeden typ prvků ... 74

7. Speciální druhy seznamů 77
7.1. Spojový seznam – `LinkedList` ... 77
7.1.1. Datová struktura zásobník ... 78
7.1.2. Datová struktura fronta ... 80

8. Iterátory 82
8.1. Metody iterátoru ... 83
8.2. Třída `ListIterator` ... 86

9. Výhodnost jednotlivých seznamů 88
9.1. Srovnání seznamů navzájem ... 88
9.2. Srovnání seznamů a polí ... 96

10. Množiny 97
10.1. Rozhraní `Set` ... 97
10.2. Implementace množin ... 98
10.3. Základní dovednosti s množinou ... 99
10.4. Práce s vlastní třídou v množině ... 100
10.4.1. Pokusy s překrýváním metod ... 100
10.5. Množinové operace ... 103
10.5.1. Triky za použití množinových operací ... 105
10.6. Použití `Collections` v množinách ... 105
10.7. Rozhraní `SortedSet` a třída `TreeSet` ... 108
10.7.1. Použití absolutního řazení u `TreeSet` ... 109
10.7.2. Vztah množiny a podmnožiny ... 110
10.7.3. Triky při použití podmnožin typu `TreeSet` ... 111
10.8. Rychlostní parametry ... 113
10.8.1. Hešovací množina versus seřazená množina ... 113
10.8.2. Množiny versus seznamy ... 114

11. Mapy 116
11.1. Implementace map ... 116
11.2. Rozhraní `Map` ... 117
11.3. Základní operace s mapou ... 119
11.4. Průchod prvky mapy ... 121
11.5. Změna mapy pomocí pohledů ... 123
11.6. Triky s mapami ... 125
11.6.1. Porovnání map ... 126
11.6.2. Praktické použití map ... 127
11.7. Třída `TreeMap` ... 129

12. Speciální dovednosti s kolekcemi 134
12.1. Vestavěná ochrana kolekcí proti nekonzistenci dat ... 134
12.2. Přidaná funkčnost – wrapery ... 136
12.2.1. Synchronizační wraper ... 136
12.2.2. Neměnitelný wrapper ... 138
12.3. Práce se singletony ... 140
12.4. Možnosti zrychlování práce s hešovací tabulkou ... 141

13. Speciální typy kolekcí 143
13.1. Třída `LinkedHashSet` ... 143
13.2. Třída `LinkedHashMap` ... 144
13.3. Třída `WeakHashMap` ... 146
13.3.1. Principy „slabých" referencí ... 146
13.3.2. Ukázka možností `WeakHashMap` ... 147
13.3.3. Praktické použití `WeakHashMap` ... 149
13.4. Třída `IdentityHashMap` ... 152

14. Jak připravit ideální třídu pro uložení do kolekcí 153
14.1. Hešování ... 153
14.1.1. Pravidla pro použití hešovacích tabulek ... 154
14.1.2. Chybné hešování ... 155
14.2. Obecné principy metody `equals()` ... 159
14.2.1. Perfektní `equals()` ... 159

14.3. Způsoby přípravy `hashCode()` ... 162
14.3.1. Primitivní řešení ... 162
14.3.2. Hešovací kód může být stejný ... 163
14.3.3. Hešovací kód se nesmí měnit ... 165
14.3.4. Doporučení pro přípravu perfektní metody `hashCode()` ... 166
14.3.5. Praktické použití perfektní `hashCode()` ... 168
14.4. Co všechno by měl mít prvek kolekce ... 173
14.4.1. Typický prvek kolekce pro české řazení ... 177

15. Extrémně velká celá čísla **183**

16. Speciální požadavky na reálné číslo **185**
16.1. Vytvoření objektu `BigDecimal` ... 186
16.2. Praktické použití `BigDecimal` ... 187
16.2.1. Pro přesné výpočty s desetinnými čísly pro hodnoty „běžných" velikostí ... 187
16.2.2. Pro přesné výpočty s čísly s extrémní přesností ... 189
16.2.3. Zaokrouhlování ... 190

17. Práce s náhodnými čísly **195**
17.1. Třída `Random` – základní použití ... 195
17.2. Generování čísel z daného intervalu ... 197
17.3. Normální rozložení pravděpodobnosti ... 198
17.4. Speciální rozložení pravděpodobnosti ... 201
17.4.1. Obecné normální rozložení ... 201
17.4.2. Exponenciální rozložení ... 203

18. Práce s datem a časem **204**
18.1. Třída `Date` ... 204
18.2. Třída `Calendar` ... 205
18.2.1. Atribut `lenient` ... 206
18.2.2. Hraniční časy ... 206
18.2.3. Konstanty třídy `Calendar` ... 207
18.2.4. Získávání hodnot ... 208

18.2.5. Formátování hodnot pro výpis ... 210
18.2.5.1. Nevhodné formátování hodnot ... 210
18.2.5.2. Doporučené formátování hodnot ... 211
18.2.5.3. Přehled základních formátovacích vzorů ... 212
18.2.5.4. Složené formátování ... 213
18.2.6. Získávání speciálních hodnot ... 214
18.2.7. Nastavení hodnot ... 215
18.2.8. Přidání hodnoty s kumulativní změnou ... 217
18.2.9. Přidání hodnoty bez kumulativní změny ... 218
18.3. Třída `GregorianCalendar` ... 220
18.4. Třída `TimeZone` ... 222
18.5. Třída `SimpleTimeZone` ... 223

19. Zip – komprimace dat **226**
19.1. Třída `ZipFile` ... 226
19.2. Třída `ZipInputStream` ... 227
19.3. Třída `ZipEntry` ... 228
19.3.1. Použití `ZipEntry` pro čtení ... 228
19.3.2. Použití `ZipEntry` pro vytvoření komprimovaných dat ... 232
19.4. Třída `ZipOutputStream` – zápis komprimovaných dat ... 233

Literatura **236**

Rejstřík **237**