

Obsah

Předmluva	24
Poděkování	24
Kapitola 1	
O knize	27
1.1 Proč tato kniha	28
1.2 Co byste měli vědět, než začnete knihu číst	29
1.3 Styl a struktura knihy	29
1.4 Jak číst tuto knihu	32
1.5 Současný stav vývoje	32
1.6 Kód příkladů a dodatečné informace	32
1.7 Zpětná vazba	32
Kapitola 2	
Úvod do jazyka C++ a standardní knihovny	27
2.1 Historie	36
2.2 Nové vlastnosti jazyka	37
2.2.1 Šablony	38
Netypové parametry šablon	38
Implicitní parametry šablon	39
Klíčové slovo typename	39
Členské šablony	40
Vnořené šablonové třídy	42
2.2.2 Explicitní inicializace základních typů	42
2.2.3 Zpracování výjimek	42
2.2.4 Jmenné prostory	44
2.2.5 Typ bool	45
2.2.6 Klíčové slovo explicit	45
2.2.7 Nové operátory typových konverzí	46
2.2.8 Inicializace konstantních statických členů	47
2.2.9 Definice funkce main()	48
2.3 Složitost a notace „velké O“	48

Kapitola 3

Obecné pojmy	51
3.1 Jmenný prostor std	52
3.2 Hlavičkové soubory	53
3.3 Zpracování chyb a výjimek	54
3.3.1 Standardní třídy výjimek	54
Třídy výjimek pro podporu jazyka	55
Třídy výjimek standardní knihovny	56
Třídy výjimek pro chyby mimo rozsah platnosti programu	57
Výjimky vyvolávané standardní knihovnou	57
Hlavičkové soubory tříd výjimek	57
3.3.2 Členy tříd výjimek	57
3.3.3 Vyvolávání standardních výjimek	58
3.3.4 Odvozování ze standardních tříd výjimek	59
3.4 Alokátory	60

Kapitola 4

Utility	61
4.1 Páry	62
Porovnávání párů	64
4.1.1 Užitečná funkce make_pair()	64
4.1.2 Příklady použití párů	65
4.2 Třída auto_ptr	65
4.2.1 Význam třídy auto_ptr	65
4.2.2 Převod vlastnictví ukazatele auto_ptr	67
Zdroj a jímka (sink)	69
Upozornění	69
4.2.3 Ukazatele auto_ptr jako členy třídy	71
4.2.4 Nesprávné používání ukazatelů auto_ptr	73
4.2.5 Příklady použití ukazatelů auto_ptr	74
4.2.6 Podrobné informace o třídě auto_ptr	77
Definice typů	78
Konstruktory, přiřazovací operátory a destruktory	78
Přístup k hodnotě	79
Manipulace s hodnotou	80
Konverze	80
Příklad implementace třídy auto_ptr	81

10.1.3 Podrobný popis třídy stack<>	427
Definice typů	427
Operace	428
10.1.4 Uživatelsky definovaná třída zásobníku	429
10.2 Fronty	431
10.2.1 Základní rozhraní	433
10.2.2 Příklad použití front	433
10.2.3 Podrobný popis třídy queue<>	434
Definice typů	435
Operace	435
10.2.4 Uživatelsky definovaná třída fronty	437
10.3 Prioritní fronty	439
10.3.1 Základní rozhraní	441
10.3.2 Příklad použití prioritních front	441
10.3.3 Podrobný popis třídy priority_queue<>	442
Definice typů	443
Konstruktory	443
Ostatní operace	444
10.4 Bitové sady	445
10.4.1 Příklady používání bitových sad	446
Bitové sady jako sady příznaků	446
Bitové sady pro operace V/V s binární reprezentací	447
10.4.2 Podrobný popis třídy bitset	448
Operace vytváření, kopírování a rušení	448
Operace neprovádějící změny	449
Operace provádějící změny	450
Přístup pomocí operátoru []	451
Vytváření nových upravených bitových sad	452
Operace typových konverzí	453
Vstupně/výstupní operace	453

Kapitola 11

Řetězce **455**

11.1 Motivace	456
11.1.1 První příklad: Vytváření jmen dočasných souborů	456
11.1.2 Druhý příklad: Získávání slov a jejich obrácený tisk	461

11.2 Popis řetězcových tříd	463
11.2.1 Řetězcové typy	463
Hlavičkový soubor	463
Šablonová třída <code>basic_string<></code>	464
Typy <code>string</code> a <code>wstring</code>	464
11.2.2 Přehled operací	465
Argumenty operací řetězců	466
Neposkytované operace	467
11.2.3 Konstruktory a destruktory	467
11.2.4 Řetězce a řetězce jazyka C	468
11.2.5 Velikost a kapacita	469
11.2.6 Přístup ke znakům	471
11.2.7 Porovnávání	472
11.2.8 Modifikátory	473
Přiřazování	473
Výměna hodnot	474
Vyprazdňování řetězců	474
Vkládání a vyjímání znaků	474
11.2.9 Podřetězce a zřetězení	475
11.2.10 Vstupně/výstupní operátory	476
11.2.11 Hledání a nalézání	477
11.2.12 Hodnota <code>npos</code>	478
11.2.13 Podpora iterátorů pro řetězce	480
Iterátorové funkce řetězců	481
Příklad použití iterátorů řetězců	482
11.2.14 Vícejazyčné aplikace	486
11.2.15 Výkon	488
11.2.16 Řetězce a vektory	488
11.3 Podrobné informace o třídě řetězců	489
11.3.1 Definice typů a statických hodnot	489
11.3.2 Operace vytváření, kopírování a rušení	491
11.3.3 Operace velikosti a kapacity	492
Operace velikosti	492
Operace kapacity	492
11.3.4 Porovnávání	493
11.3.5 Přístup ke znakům	495
11.3.6 Generování řetězců jazyka C a znakových polí	495
11.3.7 Operace provádějící změny	496
Přiřazování	496
Připojování znaků	497

Vkládání znaků	499
Mazání znaků	501
Změna velikosti	501
Nahrazování znaků	501
11.3.8 Hledání a nalézání	503
Hledání znaku	503
Hledání podřetězce	504
Hledání prvního z určitých znaků	505
Hledání posledního z určitých znaků	506
11.3.9 Podřetězce a spojování řetězců	507
11.3.10 Vstupně/výstupní funkce	507
11.3.11 Generování iterátorů	508
11.3.12 Podpora alokátorů	509

Kapitola 12

Čísla

511

12.1 Komplexní čísla	512
12.1.1 Příklady použití třídy <code>complex<></code>	512
12.1.2 Operace komplexních čísel	515
Operace vytváření, kopírování a přiřazování	515
Implicitní typové konverze	516
Přístup k hodnotám	517
Porovnávací operace	518
Aritmetické operace	519
Vstupně/výstupní operace	520
Transcendentální funkce	521
12.1.3 Podrobné informace o třídě <code>complex<></code>	522
Definice typů	522
Operace vytváření, kopírování a přiřazování	522
Přístup k prvkům	524
Vstupně/výstupní operace	525
Operátory	525
Transcendentální funkce	526
12.2 Hodnotová pole	527
12.2.1 Seznámení s hodnotovými poli	527
Hlavičkový soubor	528
Vytváření hodnotových polí	529
Operace hodnotových polí	529
Transcendentální funkce	532

12.2.2 Podmnožiny hodnotových polí	533
Problémy s podmnožinami hodnotových polí	534
Řezy	535
Obecné řezy	539
Maskované podmnožiny	542
Nepřímé podmnožiny	544
12.2.3 Podrobné informace o třídě <code>valarray<></code>	546
Operace vytváření, kopírování a rušení	547
Operace přiřazování	548
Členské funkce	548
Přístup k prvkům	549
Operátory hodnotových polí	550
Transcendentální funkce	552
12.2.4 Podrobné informace o třídách podmnožin hodnotových polí	552
Třídy řezů <code>slice</code> a <code>slice_array</code>	552
Třídy obecných řezů <code>gslice</code> a <code>gslice_array</code>	554
Třída maskovaných podmnožin <code>mask_array</code>	555
Třída nepřímých podmnožin <code>indirect_array</code>	556
12.3 Globální číselné funkce	557
Poslední změny v knihovně proudů V/V	560

Kapitola 13

Vstup/výstup pomocí tříd proudů **559**

13.1 Obecné základy proudů V/V	561
13.1.1 Objekty proudů	561
13.1.2 Třídy proudů	561
13.1.3 Objekty globálních proudů	561
13.1.4 Operátory proudů	562
13.1.5 Manipulátory	563
13.1.6 Jednoduchý příklad	563
13.2 Základní třídy a objekty proudů	564
13.2.1 Třídy a jejich hierarchie	564
Význam tříd vyrovnávacích pamětí proudů	566
Podrobné informace o definicích tříd	566
13.2.2 Globální objekty proudů	568
13.2.3 Hlavičkové soubory	568

13.3 Operátory standardních proudů << a >>	569
13.3 Operátory standardních proudů << a >>	569
13.3.1 Operátor výstupu <<	570
13.3.2 Operátor vstupu >>	571
13.3.3 Vstup/výstup speciálních typů	571
Typ bool	571
13.3 Operátory standardních proudů << a >>	571
Typy char a wchar_t	572
Typ char*	572
Typ void*	572
Vyrovnávací paměti proudů	573
Uživatelsky definované typy	573
13.4 Stav proudů	573
13.4.1 Konstanty stavu proudů	573
13.4.2 Členské funkce pro přístup ke stavu proudů	574
13.4.3 Stav proudu a booleovské podmínky	576
13.4.4 Stav proudu a výjimky	578
13.5 Funkce standardního vstupu/výstupu	582
13.5.1 Členské funkce vstupu	583
13.5.2 Členské funkce výstupu	586
13.5.3 Příklady použití	587
13.6 Manipulátory	588
13.6.1 Jak manipulátory pracují	588
13.6.2 Uživatelsky definované manipulátory	590
13.7 Formátování	590
13.7.1 Formátovací příznaky	590
13.7.2 Vstupně/výstupní formát booleovských hodnot	592
13.7.3 Šířka pole, doplňovaný znak a zarovnání	593
Využití šířky pole, doplňovaného znaku a zarovnání při výstupu	593
Využití šířky pole při vstupu	595
13.7.4 Kladné znaménko a velká písmena	596
13.7.5 Číselný základ	596
13.7.6 Zápis čísel s plovoucí desetinnou čárkou	598
13.7.7 Obecné formátovací definice	601

13.8 Vícejazyčné aplikace	601
13.9 Přístup k souborům	602
13.9.1 Příznaky souborů	606
13.9.2 Náhodný přístup	609
13.9.3 Použití souborových deskriptorů	612
13.10 Spojování vstupních a výstupních proudů	612
13.10.1 Volné propojení pomocí funkce tie()	612
13.10.2 Pevné propojení prostřednictvím vyrovnávacích pamětí proudů	614
13.10.3 Přesměrování standardních proudů	616
13.10.4 Proudů pro čtení a zápis	617
13.11 Třídy proudů pro řetězce	619
13.11.1 Třídy řetězcových proudů	619
13.11.2 Třídy proudů s typem char*	623
13.12 Vstupně/výstupní operátory uživatelsky definovaných typů	625
13.12.1 Implementace operátorů výstupu	625
13.12.2 Implementace operátorů vstupu	627
13.12.3 Pomocné funkce vstupu/výstupu	629
13.12.4 Uživatelsky definované operátory nepoužívající formátovací funkce	630
13.12.5 Uživatelsky definované formátovací příznaky	632
13.12.6 Konvence uživatelsky definovaných operátorů vstupu/výstupu	634
13.13 Třídy vyrovnávacích pamětí proudů	635
13.13.1 Uživatelský pohled na vyrovnávací paměti proudů	635
13.13.2 Iterátory vyrovnávacích pamětí proudů	637
Iterátory vyrovnávacích pamětí výstupních proudů	638
Iterátory vyrovnávacích pamětí vstupních proudů	639
Příklad použití iterátorů vyrovnávacích pamětí proudů	640
13.13.3 Uživatelsky definované vyrovnávací paměti proudů	640
Uživatelsky definované výstupní vyrovnávací paměti	641
Uživatelsky definované vstupní vyrovnávací paměti	647
13.14 Výkonnost	652
13.14.1 Synchronizace se standardními proudy jazyka C	652
13.14.2 Ukládání dat ve vyrovnávacích pamětech proudů	653
13.14.3 Přímé používání vyrovnávacích pamětí proudů	654

Kapitola 14

Vícejazyčné aplikace 657

14.1 Různé kódování znaků	659
14.1.1 Text širokých a vícebajtových znaků	659
14.1.2 Traity znaků	660
14.1.3 Speciální znaky ve vícejazyčných aplikacích	663
14.2 Lokální objekty	664
14.2.1 Použití lokálních objektů	666
14.2.2 Facety lokálních objektů	670
14.3 Podrobné informace o lokálních objektech	672
14.4 Podrobné informace o facetech	676
14.4.1 Formátování čísel	677
Interpunkce čísel	677
Formátování čísel	678
Rozbor čísel	679
14.4.2 Formátování data a času	680
Rozbor data a času	681
Formátování data a času	683
14.4.3 Formátování peněžních hodnot	684
Interpunkce peněžních hodnot	684
Formátování peněžních hodnot	686
Rozbor peněžních hodnot	687
14.4.4 Klasifikace a převod znaků	688
Klasifikace znaků	688
Specializace šablonové třídy <code>cctype</code> pro typ <code>char</code>	690
Jednoduše použitelné globální funkce klasifikace znaků	691
Převod kódování znaků	692
14.4.5 Kolace řetězců	696
14.4.6 Mezinárodní zprávy	697

Kapitola 15

Alokátory 699

15.1 Využití alokátorů aplikačními programátory	700
15.2 Využití alokátorů programátory knihoven	701
Iterátory neinicializovaných úložišť	703
Dočasná paměť	703

15.3 Implicitní alokátor	704
15.4 Uživatelsky definovaný alokátor	706
15.5 Podrobné informace o alokátorech	708
15.5.1 Definice typů	708
15.5.2 Operace	710
15.6 Podrobné informace o utilitách neinicializované paměti	711
Kde získáte standard	716
Diskusní skupiny (newsgroups)	716
Internetové adresy	716
Seznam použité literatury	719
Stručný slovník pojmů	723
Rejstřík	729

4.3 Číselná omezení	83
Třída <code>numeric_limits<></code>	84
Příklad použití šablony <code>numeric_limits<></code>	88
4.4 Pomocné funkce	90
4.4.1 Určení minima a maxima	90
4.4.2 Výměna dvou hodnot	91
4.5 Dodatečné porovnávací operátory	92
4.6 Hlavičkové soubory <code><cstdint></code> a <code><cstdlib></code>	94
4.6.1 Definice hlavičkového souboru <code><cstdint></code>	94
4.6.2 Definice hlavičkového souboru <code><cstdlib></code>	95
Kapitola 5	
Standardní knihovna šablon	97
5.1 Komponenty knihovny STL	98
5.2 Kontejnery	100
5.2.1 Sekvenční kontejnery	101
Vektory	101
Obousměrné fronty	102
Seznamy	104
Řetězce	105
Normální pole	105
5.2.2 Asociativní kontejnery	105
5.2.3 Kontejnerové adaptéry	106
5.3 Iterátory	107
5.3.1 Příklady použití asociativních kontejnerů	110
Příklady použití sad a multisad	111
Příklady použití map a multimap	113
Mapy jako asociativní pole	114
5.3.2 Kategorie iterátorů	116
5.4 Algoritmy	117
5.4.1 Rozsahy	119
5.4.2 Zpracování více rozsahů	124
5.5 Iterační adaptéry	126
5.5.1 Vkládací iterátory	126
5.5.2 Proudové iterátory	128
5.5.3 Zpětné iterátory	130

5.6 Manipulační algoritmy	132
5.6.1 „Vyjímání“ prvků	132
5.6.2 Manipulační algoritmy a asociativní kontejnery	135
5.6.3 Algoritmy versus členské funkce	136
5.7 Uživatelsky definované obecné funkce	137
5.8 Funkce jako argumenty algoritmů	138
5.8.1 Příklady funkcí jako argumentů algoritmů	138
5.8.2 Predikáty	140
Unární predikáty	140
Binární predikáty	142
5.9 Funkční objekty	143
5.9.1 Co jsou funkční objekty	143
5.9.2 Předdefinované funkční objekty	148
5.10 Prvky kontejnerů	151
5.10.1 Požadavky na prvky kontejnerů	151
5.10.2 Hodnotová nebo odkazová sémantika	152
5.11 Chyby a výjimky uvnitř knihovny STL	154
5.11.1 Zpracování chyb	154
5.11.2 Zpracování výjimek	156
5.12 Rozšiřování knihovny STL	158

Kapitola 6

Kontejnery knihovny STL **159**

6.1 Společné vlastnosti a operace kontejnerů	160
6.1.1 Společné vlastnosti kontejnerů	160
6.1.2 Společné operace kontejnerů	161
Inicializace	161
Operace zjišťování velikosti	163
Porovnávání	163
Přirazování a funkce swap()	163
6.2 Vektory	164
6.2.1 Vlastnosti vektorů	164
Velikost a kapacita	165
6.2.2 Operace vektorů	166
Operace vytváření, kopírování a rušení	166
Operace neprovádějící změny	166

Přiřazování	167
Přístup k prvkům	168
Funkce pro získávání iterátorů	169
Vkládání a vyjímání prvků	169
6.2.3 Využití vektorů jako normálních polí	171
6.2.4 Zpracování výjimek	171
6.2.5 Příklady použití vektorů	172
6.2.6 Třída vector<bool>	173
6.3 Obousměrné fronty	175
6.3.1 Vlastnosti obousměrných front	176
6.3.2 Operace obousměrných front	177
6.3.3 Zpracování výjimek	179
6.3.4 Příklady použití obousměrných front	180
6.4 Seznamy	181
6.4.1 Vlastnosti seznamů	181
6.4.2 Operace seznamů	182
Operace vytváření, kopírování a rušení	182
Operace neprovádějící změny	183
Přiřazování	183
Přístup k prvkům	183
Funkce pro získávání iterátorů	184
Vkládání a vyjímání prvků	184
Spojovací funkce	186
6.4.3 Zpracování výjimek	187
6.4.4 Příklady použití seznamů	188
6.5 Sady a multisady	189
6.5.1 Vlastnosti sad a multisad	191
6.5.2 Operace sad a multisad	192
Operace vytváření, kopírování a rušení	192
Operace neprovádějící změny	193
Speciální vyhledávací operace	194
Přiřazování	196
Funkce pro získávání iterátorů	196
Vkládání a vyjímání prvků	197
6.5.3 Zpracování výjimek	200
6.5.4 Příklady použití sad a multisad	200
6.5.5 Příklad specifikace pravidla řazení za běhu programu	204

6.6 Mapy a multimap	206
6.6.1 Vlastnosti map a multimap	207
6.6.2 Operace map a multimap	208
Operace vytváření, kopírování a rušení	208
Operace neprovádějící změny	209
Speciální vyhledávací operace	210
Přiřazování	211
Funkce pro získávání iterátorů a přístup k prvkům	211
Vkládání a vyjímání prvků	214
6.6.3 Mapy jako asociativní pole	217
6.6.4 Zpracování výjimek	218
6.6.5 Příklady použití map a multimap	218
Mapa jako asociativní pole	218
Multimapa jako slovník	220
Nalezení prvku s určitou hodnotou	221
6.6.6 Příklad map, řetězců a zadávání pravidla řazení za běhu programu	223
6.7 Další kontejnery knihovny STL	226
6.7.1 Řetězce jako kontejnery knihovny STL	226
6.7.2 Normální pole jako kontejnery knihovny STL	227
Přímé použití normálních polí	227
Obálka pole	228
6.7.3 Tabulky miniatur (hash tables)	230
6.8 Implementace odkazové sémantiky	230
6.9 Kdy použít který kontejner	233
6.10 Podrobné informace o typech a členech kontejnerů	237
6.10.1 Definice typů	237
6.10.2 Operace vytváření, kopírování a rušení	239
6.10.3 Operace neprovádějící změny	241
Operace zjišťování velikosti	241
Kapacitní operace	241
Operace porovnávání	242
Speciální operace neprovádějící změny asociativních kontejnerů	242
6.10.4 Přiřazování	245
6.10.5 Přímý přístup k prvkům	246
6.10.6 Operace získávání iterátorů	247
6.10.7 Vkládání a vyjímání prvků	248
6.10.8 Speciální členské funkce seznamů	253

6.10.9 Podpora alokátorů	256
Základní členy alokátorů	256
Konstruktory s volitelným parametrem alokátoru	256
6.10.10 Přehled zpracovávání výjimek v kontejnerech knihovny STL	257

Kapitola 7

Iterátory knihovny STL 261

7.1 Hlavičkové soubory iterátorů	262
7.2 Kategorie iterátorů	262
7.2.1 Vstupní iterátory	263
7.2.2 Výstupní iterátory	264
7.2.3 Dopředné iterátory	264
7.2.4 Obousměrné iterátory	266
7.2.5 Iterátory s náhodným přístupem	266
7.2.6 Problém inkrementace a dekrementace vektorových iterátorů	268
7.3 Pomocné funkce iterátorů	269
7.3.1 Přemístění iterátorů pomocí funkce advance()	269
7.3.2 Výpočet vzdálenosti mezi iterátory pomocí funkce distance()	271
7.3.3 Výměna hodnot iterátorů pomocí funkce iter_swap()	273
7.4 Iterační adaptéry	274
7.4.1 Zpětné iterátory	274
Iterátory a zpětné iterátory	275
Převod zpětných iterátorů na normální pomocí funkce base()	278
7.4.2 Vkládací iterátory	279
Funkčnost vkládacích iterátorů	279
Druhy vkládacích iterátorů	280
Dozadu vkládací iterátory	281
Dopředu vkládací iterátory	282
Obecné vkládací iterátory	283
Uživatelsky definovaný vkládací iterátor pro asociativní kontejnery	285
7.4.3 Proudové iterátory	285
Iterátory výstupního proudu	285
Iterátory vstupního proudu	287
Další příklad proudových iterátorů	289
7.5 Traity iterátorů	290
7.5.1 Psaní obecných funkcí pro iterátory	291
Použití typů iterátorů	292

Použití kategorií iterátorů	292
Implementace funkce distance()	293
7.5.2 Uživatelsky definované iterátory	294

Kapitola 8

Funkční objekty knihovny STL **297**

8.1 Návrh funkčních objektů	298
8.1.1 Funkční objekty jako pravidla řazení	299
8.1.2 Funkční objekty s vnitřním stavem	300
8.1.3 Návrhová hodnota algoritmu for_each()	304
8.1.4 Predikáty versus funkční objekty	305
8.2 Předdefinované funkční objekty	307
8.2.1 Funkční adaptéry	308
8.2.2 Funkční adaptéry členských funkcí	309
8.2.3 Funkční adaptéry normálních funkcí	312
8.2.4 Uživatelsky definované funkční objekty pro funkční adaptéry	313
8.3 Dodatečné kompoziční funkční objekty	314
8.3.1 Unární adaptéry kompozičních funkčních objektů	315
Vnořený výpočet pomocí compose_f_gx	316
Spojení dvou pravidel pomocí adaptéru compose_f_gx_hx	317
8.3.2 Binární adaptéry kompozičních funkčních objektů	319

Kapitola 9

Algoritmy knihovny STL **321**

9.1 Hlavičkové soubory algoritmů	322
9.2 Přehled algoritmů	322
9.2.1 Stručný úvod	322
9.2.2 Dělení algoritmů	323
Algoritmy neprovádějící změny	324
Algoritmy provádějící změny	326
Vyjímací algoritmy	327
Mutační algoritmy	328
Řadící algoritmy	328
Algoritmy seřazených rozsahů	331
Číselné algoritmy	332
9.3 Pomocné funkce	333
9.4 Algoritmus for_each()	334

9.4 Algoritmus for_each()	335
9.5 Algoritmy neprovádějící změny	337
9.5.1 Počítání prvků	337
9.5.2 Minimum a maximum	339
9.5.3 Vyhledávání prvků	340
Hledání prvního odpovídajícího prvku	340
Hledání prvních n po sobě jdoucích odpovídajících prvků	343
Hledání prvního podrozsahu	345
Hledání posledního podrozsahu	348
Hledání prvního z několika možných prvků	350
Hledání dvou po sobě jdoucích prvků se stejnou hodnotou	351
9.5.4 Porovnávání rozsahů	353
Testování rovnosti	353
Hledání první odlišnosti	355
Testování na „menší než“	356
9.6 Algoritmy provádějící úpravy	359
9.6.1 Kopírování prvků	359
9.6.2 Převádění a spojování prvků	362
Převádění prvků	362
Spojování prvků dvou řad	364
9.6.3 Výměna prvků	365
9.6.4 přiřazování nových hodnot	367
Přiřazování stejné hodnoty	367
Přiřazování generovaných hodnot	368
9.6.5 Nahrazování prvků	369
Nahrazování hodnot uvnitř řady	369
Kopírování a nahrazování prvků	370
9.7 Vyjímací algoritmy	372
9.7.1 Vyjímání určitých hodnot	372
Vyjímání prvků v řadě	372
Vyjímání prvků během kopírování	374
9.7.2 Vyjímání duplicit	375
Vyjímání po sobě jdoucích duplicit	375
Vyjímání duplicit během kopírování	377
9.8 Mutační algoritmy	379
9.8.1 Obracení pořadí prvků	379
9.8.2 Rotování prvků	381
Rotování prvků v řadě	381

Rotování prvků během kopírování	382
9.8.3 Permutace prvků	383
9.8.4 Míchání prvků	385
9.8.5 Přesouvání prvků dopředu	387
9.9 Řadicí algoritmy	388
9.9.1 Řazení všech prvků	389
9.9.2 Částečné řazení	392
9.9.3 Řazení podle n-tého prvků	395
9.9.4 Algoritmy hromady	396
Podrobnější informace o algoritmech hromady	397
Příklad použití hromad	398
9.10 Algoritmy seřazených rozsahů	400
9.10.1 Hledání prvků	400
Zjišťování přítomnosti jednoho prvků	400
Zjišťování přítomnosti několika prvků	402
Hledání první nebo poslední možné pozice	403
Hledání první a poslední možné pozice	405
9.10.2 Slučování prvků	406
Sčítání dvou seřazených sad	406
Sjednocení dvou seřazených sad	408
Průnik dvou seřazených sad	409
Rozdíl dvou seřazených sad	410
Příklad všech slučujících algoritmů	412
Slučování po sobě jdoucích seřazených rozsahů	413
9.11 Číselné algoritmy	414
9.11.1 Získávání výsledků	415
Výpočet výsledku jedné řady	415
Výpočet vnitřního součinu dvou řad	416
9.11.2 Převádění relativních a absolutních hodnot	418
Převádění relativních hodnot na absolutní	418
Převádění absolutních hodnot na relativní	419
Příklad převádění relativních hodnot na absolutní	421

Kapitola 10

Speciální kontejnery **423**

10.1 Zásobníky	424
10.1.1 Základní rozhraní	425
10.1.2 Příklad použití zásobníků	426