Obsah

Předmluva **3**

Obsah **6**

1 Spojový seznam **11**
1.1 Návrh 11
1.1.1 Návrhový vzor Most 11
1.1.2 Rozhraní šablonové třídy seznam<> 12
1.1.3 Rozhraní třídy seznam_impl<> 14
1.2 Implementace základní verze 15
1.2.1 Šablony tříd seznam<> a seznam_impl<> 15
1.2.2 Metody třídy seznam_impl<> 18
1.2.3 Metody třídy seznam<> 22
1.2.4 Testujeme základní verzi seznamu 23
1.3 Seznam ukazatelů 24
1.3.1 Použití příkazu if constexpr 25
1.3.2 Využití částečné specializace šablon 26
1.3.3 Použití šablony enable_if_t<> 27
1.4 Kopírování a stěhování instancí 28
1.4.1 Kopírování instancí 29
1.4.2 Stěhování obsahu instancí 30
1.5 Iterátor seznamu a metody, které s ním pracují 32
1.5.1 Požadavky kladené na iterátor 32
1.5.2 Třída iterátoru 34
1.5.3 Použití seznamu v příkazu for pro rozsahy 36
1.5.4 Nalezení prvku se zadanou hodnotou 38
1.5.5 Vložení nového prvku za prvek určený iterátorem 39
1.6 Spojování seznamů 42
1.6.1 Operátor + ve třídě seznam_impl<> 42
1.6.2 Implementace operátorů + ve třídě seznam<> 44
1.6.3 Operátor += 46

2 Úvod do metaprogramování **47**
2.1 Základní nástroje 47
2.1.1 Metafunkce a metavýpočet 48
2.1.2 Zobrazení hodnoty nebo typu na typ 48
2.1.3 Náhrada cyklu 51

2.2 Další příklady . . . 52
2.2.1 Výpočet faktoriálu . . . 52
2.2.2 Výpočet Ackermannovy funkce v době překladu . . . 54
2.2.3 Výpočet mocniny v době překladu . . . 55
2.2.4 Náhrada cyklu lineárním kódem . . . 57
2.2.5 Zjišťování vlastností datových typů . . . 59
2.3 Trocha historie . . . 61

3 Šablony s proměnným počtem parametrů **63**
3.1 Balík parametrů . . . 63
3.1.1 Specifikace balíku parametrů . . . 63
3.1.2 Použití balíku parametrů . . . 65
3.1.3 Redukce balíku parametrů . . . 70
3.1.4 Obecné zpracování balíku parametrů . . . 72
3.2 Příklady . . . 72
3.2.1 Jednoduché příklady . . . 73
3.2.2 Šablona uspořádané n-tice . . . 77
3.2.3 Další příklady . . . 83
3.2.4 Posloupnost hodnot . . . 84

4 Neúspěšné dosazení není chybou **93**
4.1 Co vlastně je SFINAE . . . 93
4.1.1 Podrobný popis . . . 93
4.1.2 Chyby SFINAE . . . 94
4.2 Příklady obvyklého uplatnění pravidla SFINAE . . . 95
4.3 Příklady použití SFINAE v metaprogramování . . . 97
4.3.1 Úvodní příklady . . . 98
4.3.2 SFINAE v návratovém typu . . . 100
4.4 Šablona std::enable_if<> . . . 104
4.4.1 Když překladač nedokáže funkce rozlišit . . . 105

5 Podivná rekurze šablon **107**
5.1 Statický polymorfismus . . . 107
5.2 Jiná použití podivné rekurze šablon . . . 112
5.2.1 Podobný nástroj pro různé třídy . . . 112
5.2.2 Implementace návrhového vzoru Jedináček . . . 114
5.2.3 Příklad využití CRTP ve standardní knihovně C++ . . . 116

6 Další pokročilé možnosti **119**
6.1 Generické lambda-výrazy . . . 119
6.1.1 Deklarace lambda-výrazu . . . 119
6.1.2 Jak je lambda-výraz překládán . . . 121
6.1.3 Generické lambda-výrazy . . . 123
6.2 Specifikátor auto jako parametr šablony . . . 125
6.2.1 Základní použití . . . 125
6.2.2 Použití klíčového slova auto v balíku parametrů šablony . . . 127
6.3 Deklarace strukturované vazby . . . 128

6.3.1 Syntaktická pravidla . . . 129
6.3.2 Implementujeme podporu strukturované vazby pro vlastní typ . . . 131
6.4 Užitečné drobnosti . . . 133
6.4.1 Funkce s jediným dovoleným typem parametru . . . 133
6.4.2 Automatické odvození parametrů šablony třídy (C++17) . . . 134
6.4.3 Vlastní alokátor . . . 136
6.5 Specifikátor auto: přehled použití . . . 138
6.5.1 Použití v dnešním C++ . . . 138
6.5.2 Použití v C++98 a v jazyce C . . . 143

7 Omezení šablonových parametrů **144**
7.1 Co je koncept . . . 144
7.1.1 Úvodní příklady . . . 145
7.2 Koncept . . . 151
7.2.1 Omezení . . . 152
7.2.2 Konjunkce omezení . . . 153
7.2.3 Disjunkce omezení . . . 154
7.2.4 Atomická omezení . . . 154
7.2.5 Normalizace omezení . . . 156
7.2.6 Klauzule requires . . . 156
7.2.7 Výraz requires . . . 157
7.2.8 Částečné uspořádání omezení . . . 162
7.3 Koncepty ve standardní knihovně C++20 . . . 164
7.3.1 Přehled konceptů z hlavičkového souboru <concepts> . . . 164

8 Rozsahy a pohledy **167**
8.1 Úvodní příklad . . . 167
8.2 Pojmy . . . 169
8.3 Nástroje . . . 170
8.3.1 Přístup k rozsahu . . . 170
8.3.2 Datové typy a nástroje pro ně . . . 171
8.3.3 Počítaný iterátor . . . 171
8.3.4 Pohledy . . . 172
8.3.5 Továrny na rozsahy . . . 175
8.3.6 Adaptéry . . . 178
8.3.7 Algoritmy . . . 188
8.4 Příklady . . . 191
8.4.1 Řazení obsahu rozsahů . . . 191
8.4.2 Pythagorejské trojice . . . 192
8.4.3 Spojový seznam jako rozsah . . . 197
8.4.4 Závěrečné zpracování rozsahu . . . 200

9 Moduly **207**
9.1 Úvodní příklady . . . 207
9.2 Deklarace modulu . . . 209
9.2.1 Pojmy . . . 210
9.2.2 Syntaxe deklarace modulu . . . 210

9.2.3 Globální modul 213
9.3 Deklarace exportu 214
9.4 Deklarace importu modulu 216
9.5 Fragmenty modulu 219
9.5.1 Fragment globálního modulu 219
9.5.2 Privátní fragment modulu 219

10 Korutiny **222**
10.1 První přiblížení 222
10.2 Trocha teorie 223
10.2.1 Aktivační rámec funkce 223
10.2.2 Aktivační rámec korutiny 224
10.3 Korutina v C++ 225
10.3.1 Omezení kladená na korutiny 225
10.3.2 Výpočet (provádění korutiny) 225
10.3.3 Příslib (promise) 227
10.3.4 Alokace paměti 228
10.3.5 Tělo korutiny 228
10.3.6 Vyhodnocování výrazu await 231
10.4 Nástroje pro implementaci korutin 232
10.4.1 Struktura std::coroutine_traits<> 233
10.4.2 Struktura std::coroutine_handle<> 233
10.4.3 Nástroje pro řízení čekání 234
10.4.4 Třída promise_type 235
10.4.5 Návrh korutiny 236
10.5 Příklady 236

11 Další novinky C++20 **251**
11.1 Netypové parametry šablon 251
11.2 Relační operátory 253
11.2.1 Rozdělení relačních operátorů a jejich nové vlastnosti 253
11.2.2 Operátor trojcestného porovnání 255
11.2.3 Explicitně vytvořené implicitní relační operátory 261
11.2.4 Příklad: Relační operátory pro spojový seznam 263
11.3 Atributy 267
11.3.1 Základní možnosti 267
11.3.2 Návrh podle kontraktu (C++2x) 268
11.3.3 Úroveň překladu programu a porušení podmínek 272
11.4 Modifikátor constexpr 273
11.4.1 Proměnné s modifikátorem constexpr 273
11.4.2 Funkce a metody s modifikátorem constexpr 273
11.4.3 Konstruktor s modifikátorem constexpr 275
11.5 Modifikátor consteval 275
11.5.1 Bezprostřední funkce 275
11.5.2 Příkaz if consteval 277
11.6 Konstantní inicializace 277
11.7 Podproces, který není třeba „připojit“ 279

11.7.1 Třída std::jthread 279
11.7.2 Zastavení podprocesu 281

12 Nástroje **286**
12.1 Funkce ze standardní knihovny 286
12.2 Metafunkce pro práci s datovými typy 288
12.2.1 Je to číselný typ? A jaký? 289
12.2.2 Převod celočíselného typu na znaménkový nebo bezznaménkový 290
12.2.3 Výběr nebo povolení typu 291
12.2.4 Zjišťování vztahů předek – potomek 291
12.2.5 Další operace s datovými typy 292
12.3 Další nástroje 295
12.3.1 Celočíselná posloupnost 296
12.3.2 Obal pro referenční typy 298

Literatura **300**

Rejstřík **301**