

# Obsah

<b>Úvodem .....</b>	<b>13</b>
Komu je kniha určena .....	14
Cíle knihy .....	14
Struktura knihy .....	14
Poděkování.....	16
O autorovi.....	16
Zpětná vazba od čtenářů.....	17
Dotazy.....	17
Errata .....	17

## Kapitola 1

<b>Hardware, procesory a vlákna.....</b>	<b>19</b>
Prohlídka útrob počítače .....	20
Motivace pro vícejádrové procesory .....	21
Podpora pro více vláken na jediném čipu.....	23
Zvýšení rychlosti zpracování instrukcí pomocí proudových procesorových jader....	27
Použití mezipaměti pro uchovávání právě používaných dat.....	29
Použití virtuální paměti pro uložení dat .....	31
Příklad z virtuálních adres na fyzické adresy.....	32
Charakteristické vlastnosti víceprocesorových systémů.....	34
Vliv prodlevy a šířky pásma na výkon.....	36
Příklad zdrojového kódu do jazyka symbolických adres.....	37
Srovnání výkonu 32bitového a 64bitového kódu.....	38
Zajištění správného pořadí paměťových operací.....	40
Rozdíly mezi procesy a vlákny .....	41
Shrnutí .....	44

## Kapitola 2

<b>Programování s ohledem na výkon .....</b>	<b>45</b>
Definování výkonu .....	46
Algoritmická složitost.....	47
Příklady algoritmické složitosti.....	48

Význam algoritmické složitosti.....	51
S algoritmickou složitostí je třeba zacházet opatrně.....	52
<b>Vliv struktury na výkon.....</b>	<b>53</b>
Výkon a výhodné kompromisy ve zdrojovém kódu a strukturách sestavení.....	53
Strukturování aplikací pomocí knihoven.....	56
Vliv datových struktur na výkon.....	66
<b>Role kompilátoru.....</b>	<b>71</b>
Dva typy optimalizace kompilátoru.....	73
Výběr vhodných možností kompilátoru.....	74
Jak využít optimalizaci mezi soubory pro zlepšení výkonu.....	75
Použití optimalizace pomocí informací z profilu.....	78
Jak může potenciální aliasing ukazatelů bránit optimalizacím kompilátoru.....	80
<b>Identifikování míst, kde se spotřebovává čas, pomocí profilování.....</b>	<b>83</b>
Běžně dostupné profilovací nástroje.....	84
<b>Jak neoptimalizovat.....</b>	<b>89</b>
<b>Výkon podle návrhu.....</b>	<b>90</b>
<b>Shrnutí.....</b>	<b>91</b>

## Návrhová 3

<b>Identifikování příležitostí pro paralelizmus.....</b>	<b>93</b>
Použití více procesů pro zlepšení produktivity systému.....	94
Více uživatelů využívajících jediný systém.....	96
Zlepšování efektivity stroje prostřednictvím konsolidace.....	96
Použití kontejnerů pro izolování aplikací sdílejících jeden systém.....	97
Hostování více operačních systémů pomocí hypervizorů.....	98
Použití paralelizmu pro zlepšení výkonu jediné úlohy.....	100
Jeden přístup k vizualizaci paralelních aplikací.....	100
Jak může paralelní zpracování změnit výběr algoritmů.....	101
Amdahlův zákon.....	102
Stanovení maximálního praktického počtu vláken.....	103
Jak náklady na synchronizaci redukuje škálování.....	105
<b>Vzory paralelizace.....</b>	<b>106</b>
Datový paralelizmus pomocí instrukcí SIMD.....	106
Paralelizace pomocí procesů a vláken.....	108
Více nezávislých úloh.....	108
Více volně propojených úloh.....	109
Více kopií těžé úlohy.....	110
Jediná úloha rozdělená na více vláken.....	110
Práce na jediném prvku pomocí proudu úloh.....	111

Rozdělení práce na klienta a server .....	112
Rozdělení odpovědnosti mezi producentem a konzumentem .....	113
Kombinování strategií paralelizace.....	113
<b>Jak závislosti ovlivňují schopnost kódu běžet paralelně .....</b>	<b>114</b>
Antizávislosti a výstupní závislosti .....	115
Rozbití závislostí pomocí spekulování.....	117
Kritické cesty .....	121
<b>Identifikování příležitostí pro paralelizaci.....</b>	<b>121</b>
<b>Shrnutí .....</b>	<b>122</b>

## Obsah dílu 6

<b>Synchronizace a sdílení dat .....</b>	<b>123</b>
Soupeření o data.....	124
Nástroje pro detekci soupeření o data .....	126
Jak se soupeření o data vyhnout.....	128
Synchronizační primitiva .....	128
Mutexy a kritické oblasti.....	129
Zámky spinlock .....	130
Semaforey.....	130
Zámky čtenáři-zapisovač .....	131
Bariéry .....	132
Atomické operace a kód bez zámků .....	132
Uvážnutí typu deadlock a livelock .....	134
Komunikace mezi vlákny a procesy .....	135
Paměť, sdílená paměť a soubory mapované do paměti.....	135
Podmínková proměnná.....	136
Signály a události.....	139
Fronty zpráv .....	139
Pojmenované roury .....	140
Komunikace prostřednictvím vrstev sítě.....	141
Další přístupy ke sdílení dat mezi vlákny.....	142
Soukromá data vlákna .....	142
Shrnutí .....	143

## Obsah dílu 6

<b>Vlákna v rozhraní POSIX .....</b>	<b>145</b>
Tvorbá vláken .....	146
Ukončení vlákna.....	147
Předávání dat do a z podřízených vláken.....	148

Oddělená vlákna .....	149
Nastavení atributů pro vlákna v rozhraní POSIX .....	150
Kompilování vícevláknového kódu .....	153
Ukončení procesu .....	155
Sdílení dat mezi vlákny .....	156
Ochrana přístupu pomocí zámků mutexu .....	156
Atributy mutexu .....	158
Použití zámků spinlock .....	158
Zámky pro čtení a zápis .....	160
Bariéry .....	163
Semaforey .....	164
Podmínkové proměnné .....	171
Proměnné a paměť .....	175
Víceprocesové programování .....	179
Sdílení paměti mezi procesy .....	180
Sdílení semaforů mezi procesy .....	183
Fronty zpráv .....	184
Roury a pojmenované roury .....	186
Komunikace mezi procesy pomocí signálů .....	188
Sokety .....	193
Reentrantní kód a příznaky kompilátoru .....	196
Shrnutí .....	197

## Práce s vlákny

### **Práce s vlákny v systému Windows ..... 199**

Tvorba nativních vláken systému Windows .....	200
Ukončování vláken .....	205
Tvorba a obnova běhu pozastavených vláken .....	207
Popisovače na prostředky jádra .....	207
Metody synchronizace a sdílení prostředků .....	208
Příklad nutnosti synchronizace mezi vlákny .....	209
Ochrana přístupu ke kódu pomocí kritických sekcí .....	210
Ochrana oblastí kódu pomocí mutexů .....	212
Malé zámky pro čtení a zápis .....	213
Semaforey .....	215
Podmínkové proměnné .....	217
Signalizace dokončení události dalším vláknům či procesům .....	218
Práce s širokými řetězci ve Windows .....	220
Tvorba procesů .....	221

Sdílení paměti mezi procesy .....	224
Dědění popisovačů v podřízených procesech.....	226
Pojmenování mutexů a jejich sdílení mezi procesy.....	228
Komunikace pomocí rour.....	230
Komunikace pomocí soketů.....	233
Atomické aktualizace proměnných.....	237
Alokování lokálního úložiště vlákna .....	238
Nastavení priority vlákna .....	241
Shrnutí .....	242

## Kapitola 3

### **Automatická paralelizace a rozhraní OpenMP ..... 243**

Vytvoření paralelní aplikace pomocí automatické paralelizace.....	244
Identifikace a paralelizace redukcí .....	248
Automatická paralelizace kódů obsahujících volání.....	249
Jak pomoci kompilátoru při automatické paralelizaci kódu .....	251
Vytvoření paralelní aplikace pomocí rozhraní OpenMP.....	254
Použití rozhraní OpenMP pro paralelizaci cyklů .....	255
Chování aplikace využívající rozhraní OpenMP .....	255
Stanovení oboru platnosti proměnných uvnitř paralelních oblastí rozhraní OpenMP .....	256
Paralelizace redukcí pomocí rozhraní OpenMP .....	258
Přístup k soukromým datům mimo paralelní oblast.....	259
Zlepšení rozdělení práce pomocí plánování.....	260
Použití paralelních sekcí k provádění nezávislé práce.....	264
Vnořený paralelizmus.....	265
Použití rozhraní OpenMP pro dynamicky definované paralelní úlohy .....	266
Udržení dat jako soukromých pro vlákna.....	270
Řízení běhového prostředí OpenMP .....	272
Čekání na dokončení práce .....	275
Omezení vláken provádějících určitou oblast kódu.....	277
Provádění kódu v paralelní oblasti v určitém pořadí .....	281
Sbalování cyklů pro lepší vyvážení pracovního zatížení.....	282
Zajištění konzistence paměti.....	283
Příklad paralelizace .....	284
Shrnutí .....	288

**Ručně programovaná synchronizace a sdílení ..... 289**

Atomické operace.....	290
Tvorba složitějších atomických operací pomocí instrukcí porovnání a prohození .....	292
Vynucení pořadí paměťových operací pro zajištění správné operace .....	295
Podpora direktiv pro uspořádání paměťových operací ze strany kompilátorů .....	298
Přeuspořádání operací kompilátorem .....	298
Proměnné deklarované pomocí klíčového slova volatile.....	302
Atomické operace poskytované operačním systémem .....	303
Algoritmy bez zámků .....	306
Dekkerův algoritmus.....	306
Producent-konzument s kruhovou pamětí.....	309
Škálování k více konzumentům či producentům .....	312
Škálování modelu producent-konzument na více vláken.....	313
Model producent-konzument s atomickými operacemi .....	320
Problém ABA .....	322
Shrnutí .....	325

**Škálování s vícejádrovými procesory ..... 327**

Omezení škálování aplikací.....	328
Výkon limitovaný sériovým kódem .....	328
Superlineární škálování .....	331
Nevyvážené pracovní zatížení .....	332
Přetížené zámky.....	333
Škálování knihovního kódu .....	339
Nedostatečná práce.....	340
Algoritmické omezení.....	343
Omezení škálování ze strany hardwaru.....	345
Sdílení šířky pásma mezi jádry .....	346
Falešné sdílení.....	348
Konflikt a kapacita u mezipaměti .....	351
Hladovění po prostředcích proudu .....	356
Omezení škálování ze strany operačního systému .....	361
Přetížení .....	361
Zlepšení umístění v paměti pomocí svázání s procesorem.....	363
Inverze priority .....	371
Vícejádrové procesory a škálování .....	372
Shrnutí .....	373

<b>Další techniky paralelizace .....</b>	<b>375</b>
Výpočty pomocí jednotky GPU .....	376
Rozšíření jazyků .....	379
Stavební bloky pro práci s vlákny .....	379
Jazyk Cilk++ .....	381
Technologie GCD .....	385
Funkce navrhované pro následující standardy jazyků C a C++ .....	386
Jazyk C++/CLI společnosti Microsoft .....	389
Alternativní jazyky .....	391
Technologie clusteringu .....	393
Rozhraní MPI .....	394
Algoritmus MapReduce jako strategie pro škálování .....	397
Výpočetní systémy .....	398
Transakční paměť .....	399
Vektorizace .....	400
Shrnutí .....	401

<b>Několik poznámek na závěr .....</b>	<b>403</b>
Programování paralelních aplikací .....	404
Identifikování úloh .....	404
Odhad nárůstu výkonu .....	405
Zjištění závislostí .....	405
Soupeření o data a omezení škálování u zámků mutexu .....	405
Granularita zamykání .....	406
Paralelní kód na vícejádrových procesorech .....	406
Optimalizace programů pro vícejádrové procesory .....	407
Budoucnost .....	408

<b>Rejstřík .....</b>	<b>409</b>
-----------------------	------------