
Spis treści

Słowo wstępne	9
Przedmowa	11
Podziękowania	13
O autorach	15
1 DLACZEGO CUDA? DLACZEGO TERAZ?	17
1.1. Streszczenie rozdziału	17
1.2. Era przetwarzania równoległego	17
1.2.1. Procesory CPU	18
1.3. Era procesorów GPU	19
1.3.1. Historia procesorów GPU	19
1.3.2. Początki programowania GPU	20
1.4. CUDA	21
1.4.1. Co to jest architektura CUDA	21
1.4.2. Używanie architektury CUDA	22
1.5. Zastosowania technologii CUDA	22
1.5.1. Obrazowanie medyczne	22
1.5.2. Symulacja dynamiki płynów	23
1.5.3. Ochrona środowiska	24
1.6. Podsumowanie	25
2 KONFIGURACJA KOMPUTERA	27
2.1. Streszczenie rozdziału	27
2.2. Środowisko programistyczne	27
2.2.1. Procesor graficzny z obsługą technologii CUDA	28
2.2.2. Sterownik urządzeń NVIDIA	29
2.2.3. Narzędzia programistyczne CUDA	30
2.2.4. Standardowy kompilator języka C	31
2.3. Podsumowanie	32

3	PODSTAWY JĘZYKA CUDA C	33
3.1.	Streszczenie rozdziału	33
3.2.	Pierwszy program	33
3.2.1.	Witaj, świecie!	34
3.2.2.	Wywoływanie funkcji jądra	34
3.2.3.	Przekazywanie parametrów	35
3.3.	Sprawdzanie właściwości urządzeń	38
3.4.	Korzystanie z wiedzy o właściwościach urządzeń	42
3.5.	Podsumowanie	43
4	PROGRAMOWANIE RÓWNOLEGŁE W JĘZYKU CUDA C	45
4.1.	Streszczenie rozdziału	45
4.2.	Programowanie równoległe w technologii CUDA	45
4.2.1.	Sumowanie wektorów	46
4.2.2.	Zabawny przykład	52
4.3.	Podsumowanie	60
5	WĄTKI	61
5.1.	Streszczenie rozdziału	61
5.2.	Dzielenie równoległych bloków	61
5.2.1.	Sumowanie wektorów — nowe spojrzenie	62
5.2.2.	Generowanie rozchodzących się fal za pomocą wątków	68
5.3.	Pamięć wspólna i synchronizacja	72
5.3.1.	Iloczyn skalarny	74
5.3.2.	Optymalizacja (niepoprawna) programu obliczającego iloczyn skalarny	82
5.3.3.	Generowanie mapy bitowej za pomocą pamięci wspólnej	84
5.4.	Podsumowanie	87
6	PAMIĘĆ STAŁA I ZDARZENIA	89
6.1.	Streszczenie rozdziału	89
6.2.	Pamięć stała	89
6.2.1.	Podstawy techniki śledzenia promieni	90
6.2.2.	Śledzenie promieni na GPU	91
6.2.3.	Śledzenie promieni za pomocą pamięci stałej	96
6.2.4.	Wydajność programu a pamięć stała	97
6.3.	Mierzenie wydajności programów za pomocą zdarzeń	99
6.3.1.	Pomiar wydajności algorytmu śledzenia promieni	100
6.4.	Podsumowanie	103

7	PAMIĘĆ TEKSTUR	105
7.1.	Streszczenie rozdziału	105
7.2.	Pamięć tekstur w zarysie	105
7.3.	Symulacja procesu rozchodzenia się ciepła	106
7.3.1.	Prosty model ogrzewania	106
7.3.2.	Obliczanie zmian temperatury	108
7.3.3.	Animacja symulacji	110
7.3.4.	Użycie pamięci tekstur	114
7.3.5.	Użycie dwuwymiarowej pamięci tekstur	117
7.4.	Podsumowanie	121
8	WSPÓŁPRACA Z BIBLIOTEKAMI GRAFICZNYMI	123
8.1.	Streszczenie rozdziału	124
8.2.	Współpraca z bibliotekami graficznymi	124
8.3.	Generowanie rozchodzących się fal za pomocą GPU i biblioteki graficznej	130
8.3.1.	Struktura GPUAnimBitmap	130
8.3.2.	Algorytm generujący fale na GPU	133
8.4.	Symulacja rozchodzenia się ciepła za pomocą biblioteki graficznej	135
8.5.	Współpraca z DirectX	139
8.6.	Podsumowanie	139
9	OPERACJE ATOMOWE	141
9.1.	Streszczenie rozdziału	141
9.2.	Potencjał obliczeniowy	141
9.2.1.	Potencjał obliczeniowy procesorów GPU NVIDIA	142
9.2.2.	Kompilacja dla minimalnego potencjału obliczeniowego	144
9.3.	Operacje atomowe w zarysie	144
9.4.	Obliczanie histogramów	146
9.4.1.	Obliczanie histogramu za pomocą CPU	146
9.4.2.	Obliczanie histogramu przy użyciu GPU	148
9.5.	Podsumowanie	156
10	STRUMIENIE	157
10.1.	Streszczenie rozdziału	157
10.2.	Pamięć hosta z zablokowanym stronicowaniem	158
10.3.	Strumienie CUDA	162
10.4.	Używanie jednego strumienia CUDA	162
10.5.	Użycie wielu strumieni CUDA	166
10.6.	Planowanie pracy GPU	171
10.7.	Efektywne wykorzystanie wielu strumieni CUDA jednocześnie	173
10.8.	Podsumowanie	175

11	WYKONYWANIE KODU CUDA C JEDNOCZEŚNIE NA WIELU GPU	177
11.1.	Streszczenie rozdziału	177
11.2.	Pamięć hosta niewymagająca kopiowania	178
11.2.1.	Obliczanie iloczynu skalarnego za pomocą pamięci niekopiowanej	178
11.2.2.	Wydajność pamięci niekopiowanej	183
11.3.	Użycie kilku procesorów GPU jednocześnie	184
11.4.	Przenośna pamięć zablokowana	188
11.5.	Podsumowanie	192
12	EPILOG	193
12.1.	Streszczenie rozdziału	194
12.2.	Narzędzia programistyczne	194
12.2.1.	CUDA Toolkit	194
12.2.2.	Biblioteka CUFFT	194
12.2.3.	Biblioteka CUBLAS	195
12.2.4.	Pakiet GPU Computing SDK	195
12.2.5.	Biblioteka NVIDIA Performance Primitives	196
12.2.6.	Usuwanie błędów z kodu CUDA C	196
12.2.7.	CUDA Visual Profiler	198
12.3.	Literatura	199
12.3.1.	Książka Programming Massively Parallel Processors: A Hands-on Approach	199
12.3.2.	CUDA U	199
12.3.3.	Fora NVIDII	200
12.4.	Zasoby kodu źródłowego	201
12.4.1.	Biblioteka CUDA Parallel Primitives Library	201
12.4.2.	CULATools	201
12.4.3.	Biblioteki osłonowe	202
12.5.	Podsumowanie	202
A	OPERACJE ATOMOWE DLA ZAAWANSOWANYCH	203
A.1.	Iloczyn skalarny po raz kolejny	203
A.1.1.	Blokady atomowe	205
A.1.2.	Iloczyn skalarny: blokady atomowe	207
A.2.	Implementacja tablicy skrótów	210
A.2.1.	Tablice skrótów — wprowadzenie	210
A.2.2.	Tablica skrótów dla CPU	212
A.2.3.	Wielowątkowa tablica skrótów	216
A.2.4.	Tablica skrótów dla GPU	217
A.2.5.	Wydajność tablicy skrótów	223
A.3.	Podsumowanie	224
	Skorowidz	225