

?

Hogyan írv extrém gyors programot? - Bevezetés a CUDA programozásba (Fehér Krisztián)



104 oldal, B/5, 2019., ISBN 978-615-5477-78-2

Értékelés: Még nincs értékelve

Ár

Fogyasztói ár1600,00 Ft

Kedvezmény-390,00 Ft

[Tegye fel kérdését a termékrl!](#)

Leírás

Izgalmas technológiai áttörésekről olvashatunk nap mint nap, melyek mind nagy számítási igényű megoldásokról szólnak. Az úttörő megoldások élvonalában az NVIDIA cég CUDA platformja áll, mely egyedülálló hardveres és szoftveres megoldásaival forradalmi megoldásokat kínál a jelen és a jövő kutatóinak, a videojátékfejlesztőknek és a technológia hétköznapi felhasználóinak egyaránt. Ahogy a szakma figyelme egyre inkább a videokártyák párhuzamos számítási képességeinek általános célokra történő felhasználására terelődik, úgy válik egyre fontosabbá ennek a technológiának az ismerete.

Ez a könyv a CUDA C programozás alapjairól szól Windows operációs rendszeren, kompakt és közérthető formában. Könyvünk az első magyar nyelvű, kereskedelmi forgalomban kapható szakkönyv a GPGPU (General-Purpose Computing on Graphics Processing Units) témakörében. CUDA alapú eszközökkel általános feladatokat gyorsíthatunk fel, CPU-alapú megközelítésekhez képest sokszoros mértékben. Amennyiben az olvasó rendelkezik egy NVIDIA videokártyával, akár azonnal elkezdhet CUDA programokat fejleszteni rá. Még a legolcsóbb videokártyák is gigászi számítási teljesítményt rejtnek magukban, könyvünk segítségével pedig megtanulható ennek kihasználása, tetszőleges célokra. A könyv központi példája is egy olyan grafikus motor, mely akár több száz millió háromszöget is képes megjeleníteni másodpercenként a képernyőn.

A könyvet er?s ismeretterjeszt? jellegéb?l adódóan nemcsak programozókedv? olvasók forgathatják haszonnal, hanem bárki, aki érdekli?dik a modern, adatfeldolgozást gyorsító technológiák iránt.

Tartalomjegyzék:

1. El?szó 7
 - 1.1. A téma aktualitása 7
 - 1.2. A könyv felépítése 8
 - 1.3. Kinek szól a könyv? 8
 - 1.4. A szerz?r?l 9
2. A CUDA madártávlattól 10
 - 2.1. Mi az a CUDA? 10
 - 2.2. CUDA generációk 10
 - 2.3. Mire használják a CUDA-t? 11
 - 2.3.1. Mesterséges intelligencia 11
 - 2.3.2. Adatközpontok 11
 - 2.3.3. GPU felh? 11
 - 2.3.4. Dizájn és vizualizáció, fizikai szimulációk 11
 - 2.3.5. Autonóm eszközök, önvezet? autók 12
 - 2.3.6. Játékipar 12
 - 2.3.7. Bioinformatika, vegyipar 12
 - 2.3.8. Adatkutatás 12
 - 2.3.9. Hadiipar 12
 - 2.3.10. Automatizált elektronikai tervezés 13
 - 2.3.11. Pénzügyi számítások 13
 - 2.3.12. Orvosi képképzés 13
 - 2.4. Hogyan programozhatunk CUDA-ban? 13
 - 2.4.1. Központi oldal 14
 - 2.4.2. Támogatott eszközök 14
 - 2.4.3. Fejleszt?készlet, dokumentáció 14
 - 2.5. A gyors adatfeldolgozás alapfeltételei 15
 - 2.6. Mennyivel gyorsabbak a CUDA magok? 15
 - 2.7. Összegzés 15
3. CUDA alapú kártyák bemutatása 16
 - 3.1. GeForce, Quadro, Tesla kártyák 16
 - 3.2. GeForce 16

- 3.3. Quadro 17
- 3.4. Tesla 17
- 3.5. Összegzés 17
- 4. Fejlesztőrendszer előkészítése 18
 - 4.1. A Visual Studio telepítése 18
 - 4.2. A CUDA Toolkit telepítése 18
 - 4.3. CUDA verziószámozás értelmezése 19
 - 4.4. Meghajtóprogramok 19
 - 4.5. Összegzés 19
- 5. CUDA lekérdezések 20
 - 5.1. Alapfogalmak 20
 - 5.2. Végrehajtási konfiguráció 20
 - 5.3. Új CUDA projekt létrehozása Visual Studioban 21
 - 5.4. A legegyszerűbb program 22
 - 5.5. CUDA eszközök jelenléte 23
 - 5.6. Eszközadatok lekérdezése 23
 - 5.7. További hasznos segédprogramok 26
 - 5.7.1. bandwidthTest - Memória adatátvitel sebessége 27
 - 5.7.2. topologyQuery - Topológia lekérdezése 27
 - 5.8. Összegzés 28
- 6. CUDA párhuzamos programozási alapok 29
 - 6.1. Memóriaterületek kezelése, elérése 29
 - 6.1.1. Menedzselt memóriaelérés 31
 - 6.2. Szálak, blokkok és társaik 32
 - 6.3. Párhuzamosítási technikák 34
 - 6.4. Egyszerű példa, CPU változat 35
 - 6.5. GPU kód – Aktuális végrehajtási szál meghatározása 36
 - 6.6. A legegyszerűbb program, kernellel 38
 - 6.7. Párhuzamos végrehajtás blokkok és szálak felhasználásával 39
 - 6.7.1. Több szál megadása 39
 - 6.7.2. Több blokk megadása 40
 - 6.7.3. Szinkronizáljunk! 41
 - 6.8. Kétdimenziós feldolgozás konfigurációja 42
 - 6.9. Összegzés 45
- 7. Nagyteljesítményű grafika, CPU kóddal 46

7.1. A Direct2D megoldás bemutatása 47

7.2. Pufferelés megvalósítása 48

7.3. 2D alakzatok kirajzolása 55

7.3.1. Pont kirajzolása 55

7.3.2. Vonal kirajzolása 56

7.3.3. Háromszög kirajzolása 58

7.4. Z-pufferelés 61

7.4.1. Z-pufferelt Pont kirajzolása 62

7.4.2. Z-pufferelt Vonal kirajzolása 63

7.4.3. Z-pufferelt Háromszög kirajzolása 64

7.5. Összegzés 65

8. Extrém grafikus teljesítmény – CUDA-val 68

8.1. Hol gyorsítsunk? 68

8.2. Pufferelés megvalósítása 68

8.3. 2D alakzatok kirajzolása 69

8.3.1. Pont kirajzolása 69

8.3.2. Vonal kirajzolása 69

8.3.3. Háromszög kirajzolása 70

8.4. Z-pufferelés 72

8.4.1. Z-puffer inicializálása 72

8.4.2. Z-pufferelt pont kirajzolása 72

8.4.3. Z-pufferelt vonal kirajzolása 72

8.4.4. Z-pufferelt háromszög kirajzolása 74

8.5. Összegzés 76

9. Komplex példa: extrém gyors 3D OBJ modell megjelenít? 77

9.1. Példák 84

9.2. Láthatóság vizsgálata 88

9.3. Fény használata 89

9.4. Textúrák alkalmazása 92

10. Záró gondolatok 94

11. Függelék 95

11.1. Több videokártya használata 95

11.2. CUDA programok teljesítménymérése 97

11.2.1. Általános teljesítménymérés 97

11.2.2. Teljesítménymérés CPU-val 97

11.2.3. Teljesítménymérés CUDA-ban 98

11.3. CUDA programok debuggolása 99

11.3.1. A „csináld magad” módszer 99

11.3.2. Hibakeresés az NSight beépül? modullal 99

11.3.3. CUDA hibák kiírása 100

11.4. Futtatási profilok elemzése 100

11.5. A könyvben használt CUDA függvények 102

12. Ajánlott irodalom 103