

برنامه‌نویسی مقدماتی GPU

با استفاده از معماری CUDA

www.ketab.ir

نویسنده : علی رفیعی

سرشناسه	:	رفیعی، علی، ۱۳۵۸ -
عنوان و نام پدیدآور	:	برنامه‌نویسی مقدماتی GPU (با استفاده از معماری CUDA)/نویسنده علی رفیعی.
مشخصات نشر	:	تهران: آرنا، ۱۳۹۸.
مشخصات ظاهری	:	۱۵۴ص: مصور، جدول.
شابک	:	978-600-356863-1:
وضعیت فهرست نویسی	:	فیبا
موضوع	:	کودا (ساختار کامپیوتر)
موضوع	:	CUDA (Computer architecture)
موضوع	:	برنامه‌نویسی موازی
موضوع	:	Parallel programming (Computer science)
موضوع	:	واحدهای پردازش گرافیکی -- برنامه‌نویسی
موضوع	:	Graphics processing units -- Programming
رده بندی کنگ	:	۶۴۲/۷۶QA
رده یابی دیسی	:	۳۷۵/۰۰۵
شماره کتابشناسی ملی	:	۶۰۳۰۳۸۲



عنوان	:	برنامه نویسی مقدماتی GPU (با استفاده از معماری CUDA)
نویسنده	:	علی رفیعی
ناشر	:	آرنا
چاپ	:	اول ۱۳۹۸
شمارگان	:	۱۰۰۰ نسخه
قیمت:	:	۳۶۰۰۰ تومان
شابک	:	۹۷۸-۶۰۰-۳۵۶-۸۶۳-۱

www.arnapub.com

فهرست مطالب

پیش گفتار	۱۰
مقدمه	۱۲
فصل ۱ - چرا CUDA ؟ و چرا اکنون ؟	۱۴
۱-۱- اهداف فصل	۱۴
۲-۱- صرف پردازش موازی	۱۵
۱-۱- واحدهای پردازش مرکزی	۱۵
۳-۱- ظهور محاسبات GPU	۱۷
۱-۳-۱- تاریخچه محاسبات از GPU	۱۷
۲-۳-۱- اولین پردازش ها توسط GPU	۱۹
۴-۱- CUDA	۲۱
۱-۴-۱- معماری CUDA چیست ؟	۲۲
۲-۴-۱- استفاده از معماری CUDA	۲۲
۵-۱- کاربردهای CUDA	۲۳
۱-۵-۱- تصویر برداری پزشکی	۲۴
۲-۵-۱- محاسبات دینامیک سیالات	۲۵
۳-۵-۱- علوم زیست محیطی	۲۶
۶-۱- مروری بر این فصل	۲۸

فصل ۲ - شروع کار	۲۹
۲-۱- اهداف فصل	۲۹
۲-۲- محیط توسعه نرم افزار	۲۹
۲-۲-۱- پردازنده های گرافیک با توانایی CUDA	۳۰
۲-۲-۲- درایور NVIDIA	۳۲
۲-۱-۳- جعبه ابزار توسعه CUDA	۳۲
۲-۲-۲- کمپایلرهای استاندارد	۳۵
۲-۳- مروری بر این فصل	۳۷
فصل ۳ - مقدمه ای بر C CUDA	۳۸
۳-۱- اهداف فصل	۳۸
۳-۲- اولین برنامه	۳۹
۳-۲-۱- برنامه Hello World	۳۹
۳-۲-۲- فراخوانی هسته یا Kernel	۴۰
۳-۲-۳- انتقال پارامترها	۴۱
۳-۳- جستجو در مشخصات دستگاه ها	۴۶
۳-۴- استفاده از ویژگی های دستگاه	۵۳
۳-۵- مروری بر فصل	۵۶

فصل ۴ - برنامه نویسی موازی به زبان CUDA C	۵۷
۴-۱- اهداف فصل	۵۷
۴-۲- برنامه نویسی موازی CUDA	۵۷
۴-۲-۱- جمع بردارها	۵۸
۴-۲-۲- یک مثال سرگرم کننده	۶۹
۴-۳- مروری بر فصل	۸۳
فصل ۵ - همکاری رشته ها	۸۴
۵-۱- اهداف فصل	۸۴
۵-۲- تقسیم بندی لوک های ازی	۸۴
۵-۲-۱- جمع بردارها	۸۵
۵-۲-۲- تولید موج در GPU با استفاده از رشته ها	۹۷
۵-۳- حافظه مشترک و همگام سازی رشته ها	۱۰۵
۵-۳-۱- ضرب نقطه ای	۱۰۶
۵-۳-۲- بهینه سازی نادرست ضرب نقطه ای	۱۲۰
۵-۳-۳- تصاویر بیت مپ در حافظه مشترک	۱۲۳
۵-۴- مروری بر فصل	۱۲۸

فصل ۶ - حافظه ثابت و رویدادها	۱۲۹
۶-۱- اهداف فصل	۱۲۹
۶-۲- حافظه ثابت	۱۳۰
۶-۲-۱- معرفی برنامه ردیابی پرتو	۱۳۱
۶-۲-۲- ردیابی پرتو در GPU	۱۳۳
۶-۳- ردیابی پرتو با استفاده از حافظه ثابت	۱۴۰
۶-۴-۲- افزایش عملکرد با حافظه ثابت	۱۴۳
۶-۳- سنجش عملکرد با استفاده از رویدادها	۱۴۶
۶-۳-۱- اندازه گیری عملکرد برنامه دیاب پرتو	۱۴۸
۶-۴- مروری بر فصل	۱۵۴

پیش گفتار

فعالیت های اخیر سازنده های بزرگ تراشه مانند NVIDIA این موضوع را بیش از پیش روشن می کند. که طراحی های آینده ریز پردازنده ها و پردازنده های بزرگ محاسباتی ماهیتی ترکیبی و ناهمگون دارند. این سیستم های ناهمگون متکی بر ترکیب دو نوع مهم از پردازنده ها به نسبت های مختلف می باشند :

- پردازنده های چند هسته ای : تمایل زیاد به افزایش قابلیت های سخت افزاری بیشتر در پردازنده ها و در عین حال جلوگیری از کاهش بازدهی، افزایش قابلیت موازی سازی برنامه ها و استفاده بهینه از حافظه باعث شده که تعداد هسته های پردازنده ها روز به روز بیشتر شود.
- پردازنده های خاص منگوره : شتاب دهنده های گرافیکی موازی : برای مثال اخیراً GPU های شرکت NVIDIA در عملکرد ممیز شناور، از CPU های معمولی پیشی گرفته اند. همچنین برنامه نویسی GPU ها به آسانی و حتی آسان تر از برنامه نویسی CPU های چند هسته ای است.

تبادل نسبی استفاده بین این انواع پردازنده در طراحی های آینده روشن نیست و احتمالاً با گذشت زمان تغییر می کند. تردیدی وجود ندارد که نسل های آینده سیستم های کامپیوتری ، از لپ تاپ ها گرفته تا ابر کامپیوترها از ترکیب پردازنده های ناهمگون تشکیل می شوند و در آینده توانایی عملکرد پتافلاپ (10^{15} عمل ممیز شناور در ثانیه) توسط این سیستم ها نیز دور از ذهن نیست.

البته مشکلات و چالش های پیش روی تولید کننده ها در طراحی های جدید پردازنده های ترکیبی هنوز نا امید کننده هستند. پیشرفت و تغییرات نرم افزارهای پایه مانند سیستم های عامل و نرم افزارهای بزرگ کاربردی در حال حاضر نمی توانند با سرعت تغییرات

سخت‌افزارها برابری کنند. در برخی موارد، با افزایش تعداد هسته‌ها در پردازنده نمی‌توان بازدهی بهتری را بدست آورد زیرا زمان بسیار زیادی صرف انتقال داده‌ها می‌شود به جای اینکه صرف محاسبات شود. در موارد دیگر، نرم‌افزارهای منطبق با سخت‌افزارهای جدید سال‌ها پس از رسیدن سخت‌افزار ارائه می‌شود در نتیجه در هنگام تحویل منسوخ شده است. و در برخی موارد مثل GPU های جدید، نرم‌افزار اصلاً اجرا نمی‌شود زیرا محیط‌های برنامه‌نویسی تغییرات زیادی داشته‌اند.

این کتاب به معرفی محیط برنامه‌نویسی برای GPU ها که از قدرتمندترین ابزار محاسبات موازی به‌شمار می‌رود می‌پردازد.

در این کتاب، با استفاده از مثال‌ها و برنامه‌های مختلف و استفاده موثر از امکاناتی که توسط GPU های ساخت شرکت NVIDIA ارائه می‌شود، برنامه‌نویسی به زبان CUDA C به شما معرفی می‌گردد. همچنین، اهمیت اولیه محاسبات موازی از مثال‌های ساده تا اشکال‌زدایی (هم از لحاظ منطقی و هم در لحاظ برنامه‌نویسی) ارائه می‌شود. همچنین در این کتاب به مسائل و موضوعات پیشرفته در ارتباط با استفاده و ساخت بسیاری از برنامه‌های موازی اشاره شده است. در سرتا سر کتاب با استفاده از مثال‌های برنامه‌نویسی مختلف، مفاهیم ارائه شده روشن‌تر می‌شوند.

خواندن این کتاب برای همه افرادی که با سیستم‌های محاسباتی مبتنی بر GPU ها سر و کار دارند پیشنهاد می‌شود. در این کتاب مفاهیم محاسبات موازی به‌وسیله‌ای می‌شوند و رویکردی در قبال بسیاری از مسائلی که ممکن است با آنها مواجه باشیم ارائه می‌شود. این کتاب به خصوص برای تولیدکننده‌های برنامه‌های کاربردی، برنامه‌نویسان برنامه‌های موازی و دانشجویان دروس محاسبات موازی مفید است.