Live Lecture Chat Window

May 1, 2024

14:24:31 Would a GPU based on the older more hyper-specialized architecture be more efficient if the hard-wired allocation is the correct one, or is the modern architecture just better?

I suppose it would be more efficient if it was exactly tuned (ratio of pixel processors to vertex processors) for the application, but being able to dynamically adjust would be better in the long run.

14:28:15 How do AMD and other people count cores? I've noticed non-NVIDIA core counts are a lot smaller, so are people like Apple and AMD only counting the Streaming Multiprocessors as 'cores'?

If the number of "cores" is in the double or triple digits, then they are probably counting what Nvidia would call an SM. If the number of "cores" is in the 4-5 digit range, then that probably is the CUDA Core-equivalents.

14:36:18 Will the Nvidia Blackwell be available to the public?

It appears so. The Tom's Hardware article puts the cost of the 50x0 line as starting at $300 and going up to $2,000. They point out that this is guesswork on their part, but their guesswork is usually pretty good.

14:56:20 From Alex Walsh to Everyone:
"Nvidia improves airflow through the cooler by removing its DVI output. You get three DisplayPort connectors and one HDMI 2.0 port, while a bundled DP-to-DVI dongle covers anyone still using that interface."
https://www.tomshardware.com/reviews/nvidia-geforce-gtx-1080-ti,4972.html

That makes sense. Thanks.

14:57:20 What do you mean by streaming?

Blasting rivers of data into the processors, which you can do because each piece of data is undergoing the exact same operations. There are no choices (if-statements) to be made.

14:59:54 From Khaled to Everyone:
https://www.youtube.com/watch?v=-P28LKWTzrI

I love this video! This was done at an Nvidia GTC conference a few years ago. Good explanation of doing something serially versus doing it in parallel. That fact that it was something designed to make a mess makes it even better! 😊

15:04:25 I have a question about project 3. I'm having an issue with my random seeding.

You don't have to do random seeding at all. Just leave the leap-through-the-Cities approach in place from the sample code.

15:17:02 That was my question if we share the whole struct or elements of the structs for pragma?

You apparently cannot declare as shared the elements of the struct, just the entire array of structs (I tried it too).

15:17:02 Would we need collapse for our pragma for this?

Don't think so. The collapse clause is at its very best when you have nested for-loops that are doing roughly the same operation on different directions of the data. A good example is looking through a 2D array. You could collapse the outer Y-for-loop into the inner X-for-loop.

[Thanks to all those who shared pet photos at this point. I think you brightened everyone's day!]