

# Paralelismo com a GPU

De Rina Wilk para Chorume

15/08/2023

## Sumário

### Começo

#### 1- Apresentação e introdução.

- Sobre a Rina.
- O que é low-level e paralelismo.
- O que é a GPU e o paralelismo com GPU.
- Porque é uma área extremamente importante para o avanço da humanidade.
- Breve explicação sobre as principais tecnologias conhecidas (Vulkan, Opengl, Metal e DirectX).

#### 2- Qual a base de conhecimento e a preparação necessária para iniciar nessa área.

- Uma área muito voltada para a academia.
- Base de conhecimento que você encontra desde o ensino médio até a faculdade.
- É preciso aprender a ser auto-didata.
- Matemática e física é importantíssimo (álgebra, geometria analítica, trigonometria, álgebra linear etc).
- Linguagem de sombreamento/shading languages.
- Paixão necessária.

#### 3- Quais os diferentes campos e sua situação no mercado exterior.

- Biotecnologia, simulação, campos científicos, AI, desenvolvimento de jogos, desenvolvimento de engines, kernel/driver, compiladores/interpretadores, computação gráfica, graphics programming.
- Todas essas áreas compartilham do mesmo fundamento, embora tenham diferenças de abstrações na maioria dos casos.
- Graphics programming, desenvolvimento de game engines, desenvolvimento de jogos e a diferença para a GPGPU.
- O mercado no uso de GPU para proposito geral (GPGPU): IA, automatização, embedded, linguagens de programação, compilador/interpretadores, shaders.
- Freelancer, desenvolvimento de jogos/engines A e AA.
- Desenvolvimento de jogos AAA.

#### 4- Situação do Brasil e o campo graphics programming.

- Criticar o baixo índice de vagas pelo país.
- Trazer os motivos.

#### 5- Conclusões.

- Diferentes áreas que necessitam de uma boa base de conhecimento específico.
- Como a GPU está impactando a humanidade e a falta de mercado no Brasil.
- Futuro.

## **Programação e a Realidade**

### 6- Introdução ao funcionamento da GPU.

- Breve explicação sobre o conceito de pipelines e como a GPU é diferente em comparação com a CPU.
- Instruções, operadores matemáticos e processadores de trabalho específicos.
- Driver - comunicação com as APIs -, GPGPU e consoles.
- Papel do OpenGL e Vulkan.

### 7- Renderização.

- Diferença de fundamentos entre OpenGL e Vulkan.
- Conceito de buffers.
- Abstrações na hora de renderizar: estágios de shader e processo de masterização.
- Shaders.
- Álgebra linear.
- Matrizes de projeção, efeitos 2D (motion blur, hdr, bloom etc).
- Algoritmos de renderização.

### 8- Principais fornecedores/vendedor: Nvidia, AMD e Intel.

- Porque diferentes fornecedores/vendedor (marcas) de GPUs conseguem renderizar as mesmas coisas.
- Diferença de OpenCL e CUDA para APIs de renderização.
- Quais as dificuldades de programar engines e a incompatibilidade de arquiteturas em diferentes fornecedores/vendedor de GPUs.

### 9- Otimização via GPU, solução de falta de poder computacional e AI.

- Como ocorre a otimização via GPU - Compute shaders -.
- Dificuldades pela falta de flexibilidade da GPU.
- Uso de energia e processo de sincronização externa.

### 10- Conclusões.

- Como a GPU é diferente da CPU.
- Como o processo de renderização é complexo.
- Problemas enfrentados pelas diferentes arquiteturas.
- GPGPU e seus contrapontos.

## **Gatinhos**

### 11- Agradecimentos.