

Вопросы к экзамену по курсу “Вычисления на видеокартах”. Осенний семестр 2020 г..

- 1) Чем отличается архитектура CPU и GPU?
- 2) Как в целом выглядит топология компьютера включая IO (диск), CPU, RAM, GPU и VRAM?
- 3) Что такое coalesced memory access? Бывает ли он при работе с локальной/глобальной памятью GPU? Приведите пример с хорошим и плохим случаем.
- 4) Что такое bank conflicts? Бывают ли они при работе с локальной/глобальной памятью GPU? Приведите пример с хорошим и плохим случаем.
- 5) Что такое code divergence? Приведите пример с хорошим и плохим случаем.
- 6) В каком случае можно не выставлять барьер между обращениями к одной и той же локальной памяти из разных workItems? Возможно ли выполнить эту оптимизацию с поддержкой произвольной архитектуры? Если да - то как?
- 7) Какие бывают узкие места в работе ядра на GPU? От чего ядро может тормозить?
- 8) Что и почему является узким местом в реализации на GPU каждой из задач: сложение двух массивов, перемножение матриц, N-body симуляция гравитации?
- 9) Расскажите как найти сумму элементов массива на GPU.
- 10) Расскажите как по массиву чисел построить массив префиксных сумм на GPU.
- 11) Расскажите как адаптировать алгоритм сортировки bitonic sort на GPU.
- 12) Расскажите как адаптировать алгоритм сортировки radix sort на GPU.
- 13) Расскажите как адаптировать алгоритм сортировки merge sort на GPU.
- 14) Как реализовать перемножение плотных матриц на GPU?
- 15) Как можно эффективно хранить разреженную матрицу? В каких случаях какой вариант предпочтительнее?
- 16) Как реализовать software растеризацию множества треугольников на видеокарте?
- 17) Как эффективно реализовать растеризацию полуплоскости в квадрате 8x8?
- 18) Что такое Look Up Table (LUT)? Приведите пример.
- 19) Зачем в реконструкции 3D поверхности методом пуассона и вариационным методом балансировка октодерев? Можно ли обойтись без нее на CPU? А на GPU?
- 20) Как эффективно реализовать N-body симуляцию взаимного гравитационного притяжения на GPU? А на CPU?
- 21) Как можно реализовать трассировку лучей с kd-tree хранящим треугольники (с которыми нам надо проверять наш луч на пересечения) так чтобы был code divergence, и как - чтобы его почти не было?
- 22) Пусть есть произвольная программа с исходным кодом, у вас есть 40-часовая рабочая неделя и ваша задача ее оптимизировать - как вы подступитесь?
- 23) Есть много фотографий, требуется каждую фотографию обработать каким-то алгоритмом. Как реализовать архитектуру приложения которая будет максимально эффективно использовать произвольное количество видеокарт (включая случай когда их нет) и ядер процессора.
- 24) Что такое Z-Order Curve/Morton code?
- 25) Как реализовать Ray Marching на базе SDF? Как посчитать нормаль в точке пространства на базе SDF? Как посчитать насколько затенена точка пространства (точечный источник освещения недалеко, требуется поддержка полутеней)?
- 26) Что такое персистентные потоки? В каких случаях и за счет чего они эффективнее типичной ситуации “каждая рабочая группа обрабатывает свой кусок данных”?