

# Оснoвы вычислений общего назначения на ГПУ

## 1. Tutorial 1 – Matrix Addition

Пример демонстрирует простейшую реализацию сложения двух матриц на графическом процессоре с использованием языка GLSL (OpenGL Shading Language).

### Задание

- Модифицируйте пример таким образом, чтобы выполнялось алгебраическое умножение двух матриц. Вычислите производительность такой реализации (GFLOPS). Какой фактор является лимитирующим для получения высокой производительности?
- Оптимизируйте предыдущую реализацию, используя *четырёхкомпонентные* текстуры вместо *однокомпонентных*. Как изменится производительность?

$a_{00}$	$a_{10}$	$a_{20}$	$a_{30}$	$a_{40}$	$a_{50}$	$a_{60}$	$a_{70}$
$a_{01}$	$a_{11}$	$a_{21}$	$a_{31}$	$a_{41}$	$a_{51}$	$a_{61}$	$a_{71}$
$a_{02}$	$a_{12}$	$a_{22}$	$a_{32}$	$a_{42}$	$a_{52}$	$a_{62}$	$a_{72}$
$a_{03}$	$a_{13}$	$a_{23}$	$a_{33}$	$a_{43}$	$a_{53}$	$a_{63}$	$a_{73}$

 $\times$ 

$b_{00}$	$b_{10}$	$b_{20}$	$b_{30}$
$b_{01}$	$b_{11}$	$b_{21}$	$b_{31}$
$b_{02}$	$b_{12}$	$b_{22}$	$b_{32}$
$b_{03}$	$b_{13}$	$b_{23}$	$b_{33}$
$b_{04}$	$b_{14}$	$b_{24}$	$b_{34}$
$b_{05}$	$b_{15}$	$b_{25}$	$b_{35}$
$b_{06}$	$b_{16}$	$b_{26}$	$b_{36}$
$b_{07}$	$b_{17}$	$b_{27}$	$b_{37}$

## 2. Tutorial 2 - Particle System

Пример демонстрирует обработку большого числа *невзаимодействующих* частиц (до  $10^6$ ). Начальные состояния частиц генерируются случайным образом в соответствии с некоторым правилом. На частицы влияет сила гравитации, под действием которой они будут опускаться вниз. При этом возможны столкновения со стенками ящика, внутри которого находятся частицы, и с вращающимся шаром.

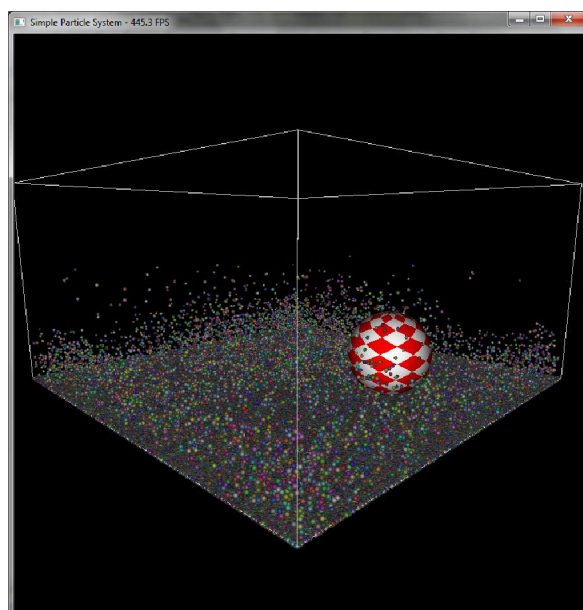
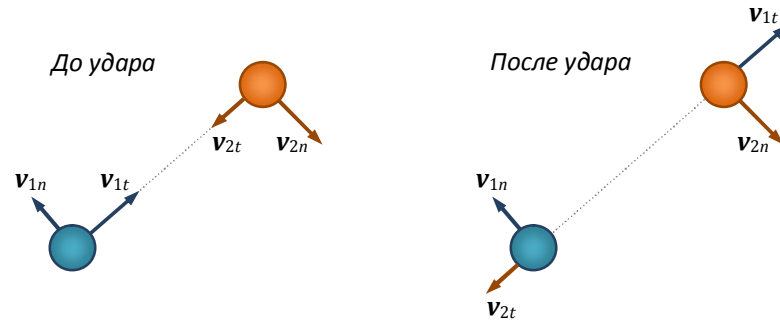


Рис. 1. Пример работы приложения *Tutorial 2 - Particle System*

### Задание

- Изучите данный пример использования системы частиц и подумайте, какие эффекты и объекты могут быть смоделированы таким образом. Измените законы генерации начальных атрибутов частиц: замените прямоугольную площадку на круглую и предложите новое правило получения начальных скоростей.

- Реализуйте взаимодействие между отдельными частицами системы. В простейшем случае удар частиц можно обработать следующим образом. Из скоростей частиц выделяются составляющие, которые лежат на прямой, проходящей через их центры масс. Данные составляющие меняются местами. Составляющие скорости, ортогональные указанной прямой, не изменяются.



- Попробуйте смоделировать какое-либо явление с помощью системы частиц. Например, можно рассмотреть многоступенчатый водопад. При этом начальные положения частиц должны лежать на некотором отрезке или ломаной линии (будем считать, что вода падает с некоторого выступа), начальные скорости частиц имеют лишь горизонтальную составляющую, а частицы падают под действием силы тяжести (используется то же самое уравнение движения). Цвет частиц также следует генерировать случайно, однако не во всем множестве возможных цветов, а в некоторой окрестности синего и голубого цвета (одинаковый цвет использовать не стоит, так как частицы будут неразличимы). На пути следования частиц встречаются препятствия в виде сфер или прямоугольников.

### 3. Tutorial 3 - N-Bodies Simulation

Пример демонстрирует интерактивное моделирование гравитационного взаимодействия  $N$  точечных масс ( $N \sim 16K$ ). В начальный момент времени заданы массы, положения и скорости материальных точек. Требуется найти *положения* точек для всех *последующих* моментов времени. Для моделирования используются распространенные методы прямого интегрирования: Эйлера, Рунге-Кутты 2-ого и 4-ого порядка.

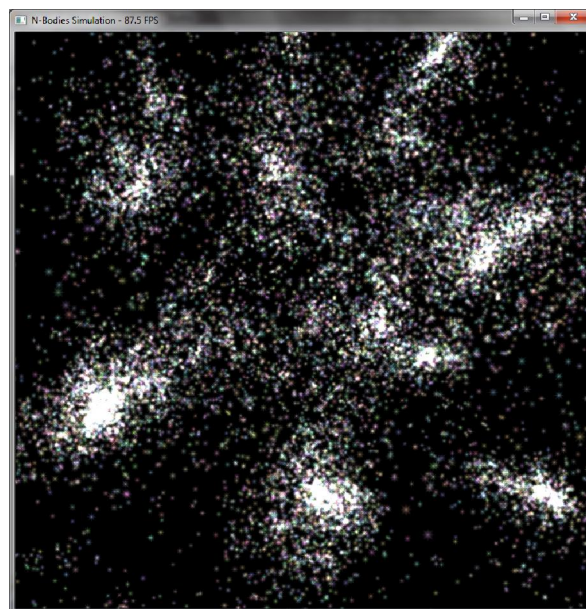


Рис. 2. Пример работы приложения *Tutorial 3 - N-Bodies Simulation*

#### Задание

- Измените законы генерации начальных атрибутов частиц для качественного моделирования таких объектов как двойная звезда или спиральная галактика.

- Введите новые факторы в процесс моделирования. Например, установите в центр системы тело с существенно большей массой.

#### 4. Tutorial 4 - Implicit Surfaces

Программа выполняет визуализацию неявно заданной поверхности, отвечающей уравнению  $F(x, y, z) = 0$ . Для визуализации используется метод классической трассировки лучей.

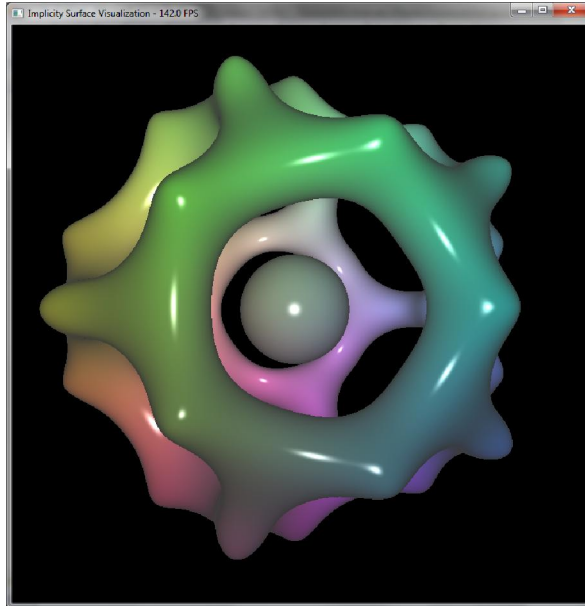


Рис. 3. Пример работы приложения *Tutorial 4 - Implicit Surfaces*

#### Задание

- Расширьте набор доступных объектов для визуализации. Для добавления нового объекта необходимо разработать два основных алгоритма: поиск точки пересечения с лучом и расчет нормали в заданной точке. Например, расположите “под поверхностью” прямоугольную или круглую площадку. Считая, что площадка является гладкой, рассчитайте отраженное изображение поверхности.
- Добавьте на сцену второй точечный источник света (первый источник света совпадает с камерой или наблюдателем). Реализуйте динамическое перемещение данного источника во время визуализации. Рассчитайте тени для данного источника света (выпуская теневой луч).
- Реализуйте какой-либо алгоритм сглаживания с целью устранения зазубренности линий и резких цветовых переходов (например, можно реализовать простейший метод *суперсэмплинга*, трассируя для каждого пикселя несколько первичных лучей и усредняя итоговый цвет). Подумайте, каким образом можно повысить эффективность таких алгоритмов.