



# Визуализация взаимодействия частиц при столкновении (взрыв)

Студент:

Сиденко Анастасия Генадьевна  
ИУ7-53Б

Руководитель:

Кострицкий Александр Сергеевич



# Введение

На протяжении десятилетий взрывы были самыми динамичными и визуально привлекательными спецэффектами в кино и видеоиграх.

Причины для использования компьютеров при создании взрывных эффектов:

- забота о безопасности актеров
- компьютерные взрывы дешевле и быстрее





# Цели и задачи

**Целью проекта** является создание максимально приближенной модели взрыва большого числа частиц, при столкновении с телом, имеющим больший размер с использованием графического редактора систем частиц. Моделирование основано на физическом явлении взрыва и возникающих побочных эффектов в заданном пространстве за заданное время и взаимодействующих с окружающей средой.

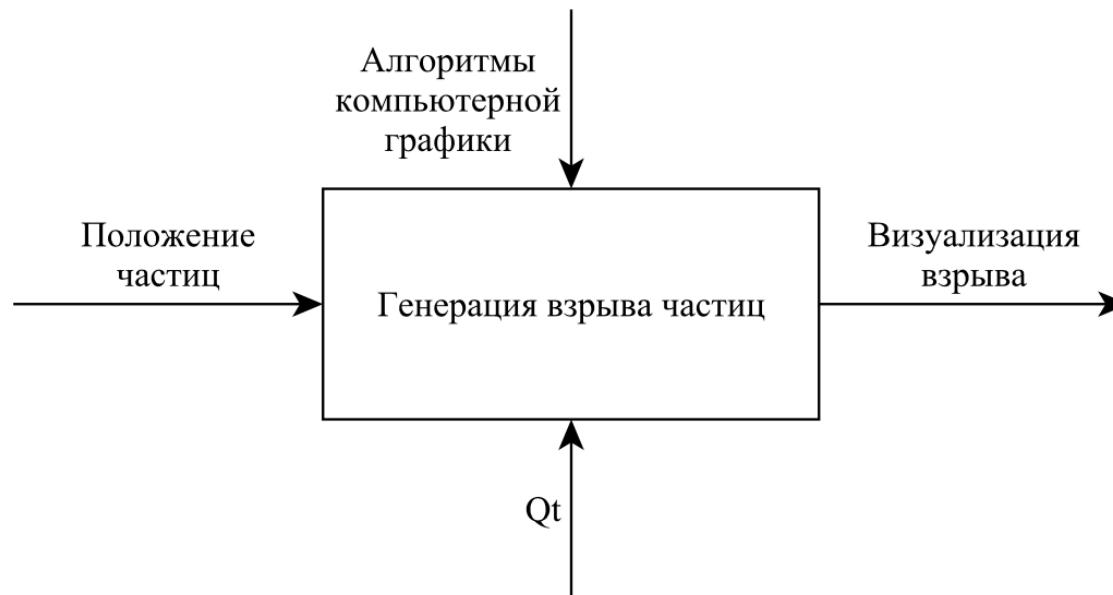
**Для достижения поставленной цели необходимо решить следующие задачи.**

- Определить понятие системы частиц.
- Создать движок для работы с частицами.
- Изучить физическое явление - взрыв.
- Смоделировать взрыва большого числа частиц, при столкновении с телом.



# Аналитический раздел

- **Система частиц** – широко используемый в компьютерной графике метод представления объектов, не имеющих четких геометрических границ.
- Визуализация взрыва частиц может быть описана по разному в зависимости от источника силы, порождающей взрыв. В данной работе, источником является шарообразное тело, врезающееся в систему более мелких частиц.
- **Функциональная модель**, отображающая структуру и функции системы:





# Аналитический раздел

## Обзор и анализ существующих решений

3D моделирование объектов	
Полигонами	Частицами
Полая модель без внутреннего наполнения	Полностью заполняют внутренности модели, где каждая частица представляется как материальная точка с дополнительными атрибутами



# Аналитический раздел

## Обзор и анализ существующих решений

Алгоритмы удаления невидимых поверхностей		
Алгоритм, использующий z-буфер	Алгоритм разбиения области Варнока	Метод трассировки лучей
<ul style="list-style-type: none"><li>+ Наиболее простой.</li><li>+ Время работы алгоритма не зависит от сложности сцены.</li><li>+ Многоугольники, составляющие сцену, могут обрабатываться в произвольном порядке.</li><li>- Затраты памяти.</li><li>- Трудности с реализацией эффектов прозрачности или просвечивания и устранения лестничного эффекта.</li><li>- Сложная работа с поверхностями второго порядка.</li></ul>	<ul style="list-style-type: none"><li>- Более сложный.</li><li>- Алгоритм работает с многоугольниками.</li></ul>	<ul style="list-style-type: none"><li>+ Простой, позволяет совместить определение видимости с расчётом цвета пикселя.</li><li>+ Сцена состоит из геометрических примитивов, пересечение вычисляется аналитически.</li><li>+ Может быть модифицирован для отображения общего освещения сцены.</li><li>- Большое количество вычислений и следовательно большие временные затраты.</li></ul>



# Конструкторский раздел

## Математические основы метода математического моделирования

- Алгоритм обратной трассировки лучей
  - Определение невидимых граней
    - Пересечение луча со сферой
    - Пересечение луча с плоскостью
  - Источники освещения
  - Диффузное рассеяние
  - Тени
- Моделирование взрыва
  - Построение физической и математической модели взаимодействия



# Технологический раздел

Для выполнения проекта был выбран язык программирования C++/QT, объектно-ориентированный подход программирования и паттерны проектирования

## Алгоритм программы.

1. Визуализация начального экрана и интерфейса.
2. Ввод и чтение из пользовательского файла.
3. Если ошибок не возникло, загрузка новой модели.
4. Визуализация модели с учетом матрицы камеры.
  1. Алгоритм трассировки для каждого пикселя экрана.
  2. Определение затененности для каждого пикселя экрана.
  3. Применение диффузного отражения для каждого пикселя экрана.
  4. Отрисовка пикселей.
5. При нажатой кнопке взрыва.
  1. Пересчет скорости каждого объекта по законам физики.
  2. Пересчет положения каждого объекта с учетом скорости.
6. Переход к пункту 4, если не загружен новый файл, иначе пункт 2.

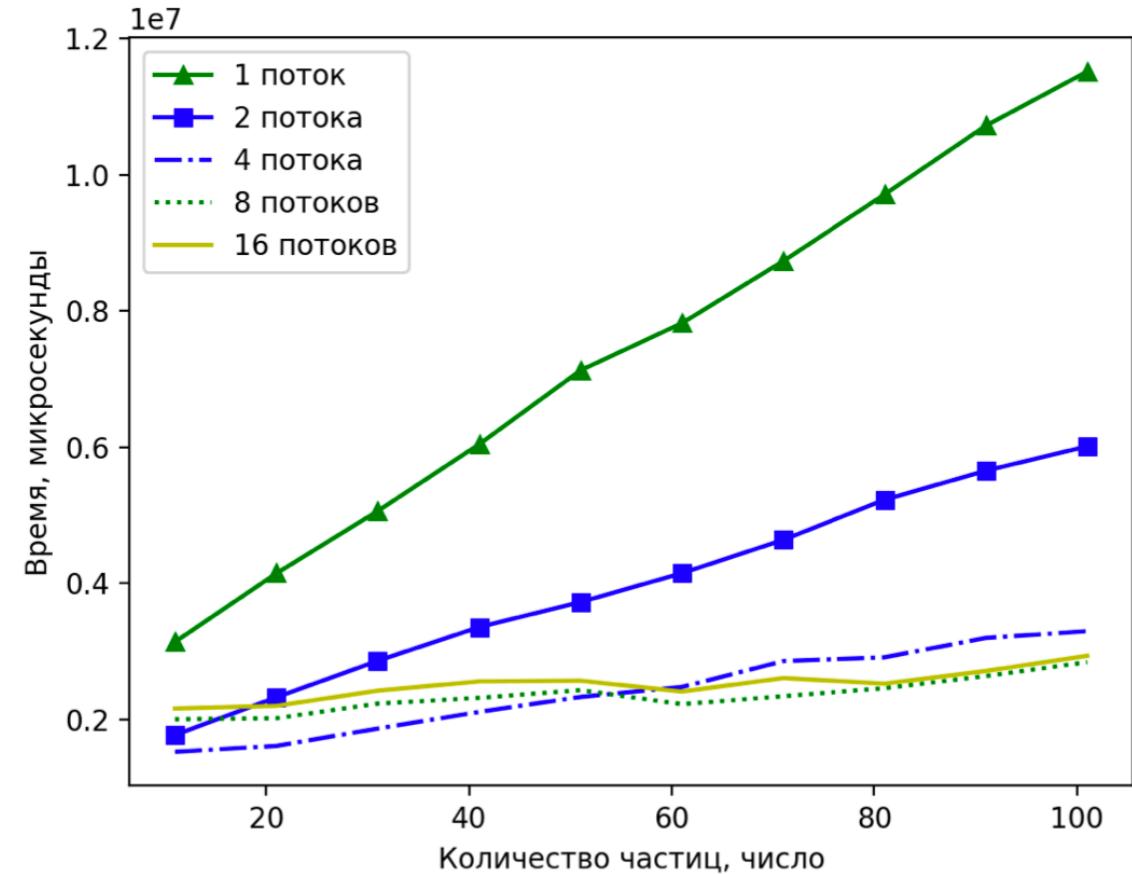


# Экспериментальный раздел

В работе для ускорения вычислений используется **параллельное программирование**. Необходимо исследовать оптимальное число потоков необходимых для наиболее быстрой генерации изображения.

Было проведено исследование для определения оптимального количества потоков.

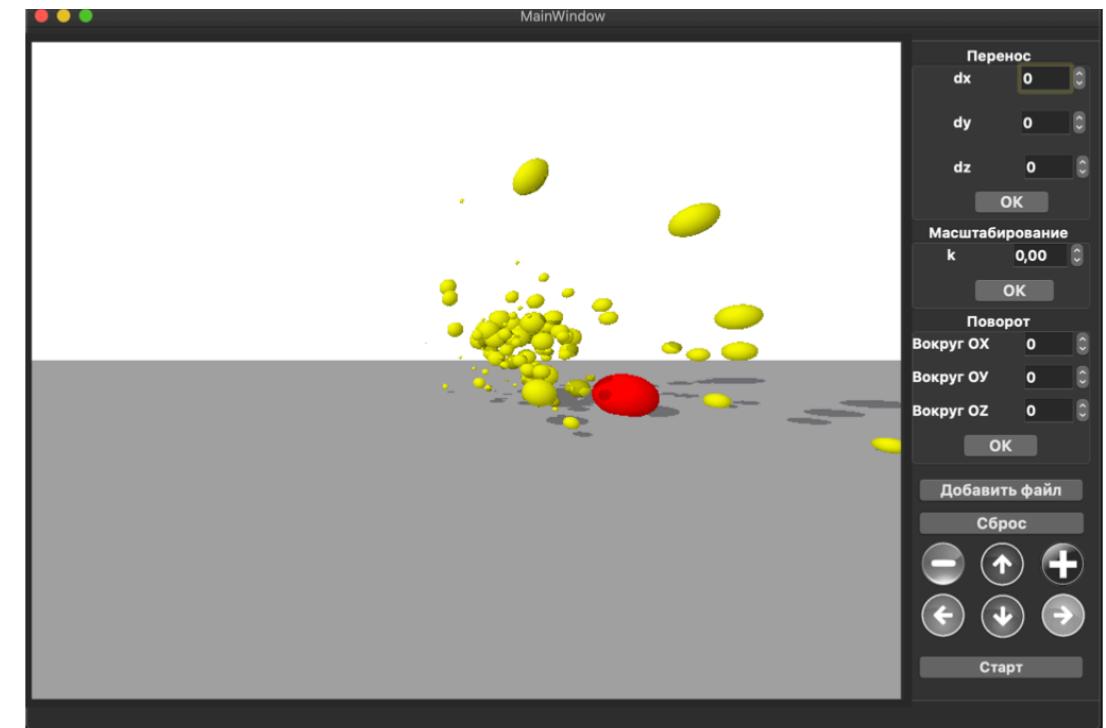
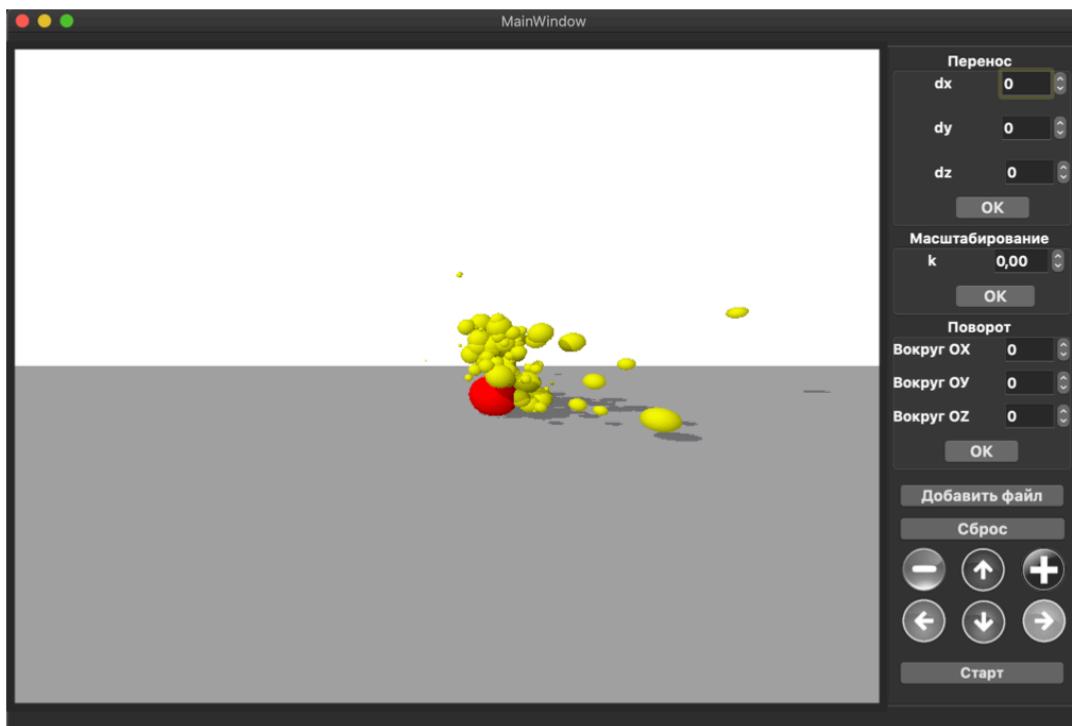
Результаты представлены на графике.





# Экспериментальный раздел

## Примеры работы программы





# Выводы

**В результате проделанной работы выполнены следующие задачи.**

- Определено понятие системы частиц.
- Создан движок для работы с частицами.
- Изучено физическое явление - взрыв.
- Смоделирован взрыв большого числа частиц при столкновении с телом.

**Достигнута цель проекта** – создание максимально приближенной модели взрыва большого числа частиц при столкновении с телом, имеющим больший размер с использованием графического редактора систем частиц.



**Спасибо за внимание!**