

# Программа моделирования детской игрушки “Лизун”

Студент: Царев Антон Андреевич ИУ7-53Б

Научный руководитель: Новик Наталья Владимировна

# Цели и задачи

**Целью** данной курсовой работы является разработка программного обеспечения, моделирующего детскую игрушку “Лизун”

**Задачи** работы:

- выбрать модель тела и алгоритмы, необходимые для реализации программы моделирования детской игрушки “Лизун”
- разработать соответствующее программное обеспечение
- сравнить среднее время работы параллельной и последовательной реализаций алгоритма удаления невидимых линий и поверхностей
- закрепить знания и навыки, приобретенные в ходе изучения курса компьютерной графики

# Объекты сцены

- Камера
- Точечный источник света
- Пол
- Слайм

# Выбор модели представления объекта

- Каркасная модель
- Поверхностная модель
- Объемная модель

# Выбор модели представления поверхности объекта

- Полигональная сетка
- Аналитическое описание объекта

# Выбор метода физического моделирования объекта

- Метод конечных элементов
- Метод граничных элементов
- Метод с использованием модели масс с пружинами

# Выбор алгоритма удаления невидимых ребер и поверхностей

- Алгоритм Робертса
- Алгоритм Варнока
- Алгоритм, использующий z-буфер
- Алгоритм обратной трассировки лучей

# Выбор модели освещения

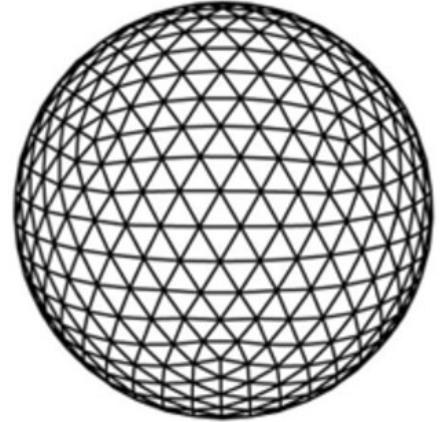
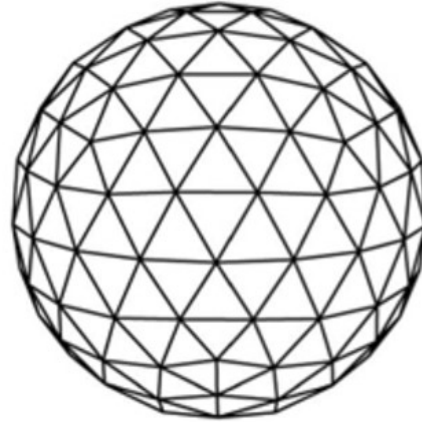
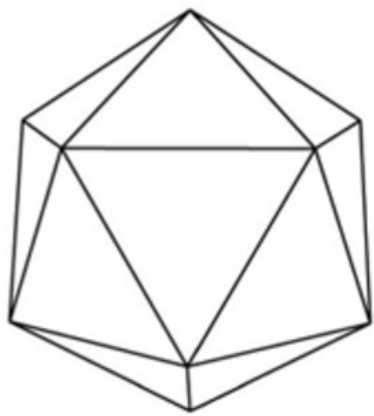
- Модель Ламберта
- Модель Фонга
- Глобальная модель Уиттеда



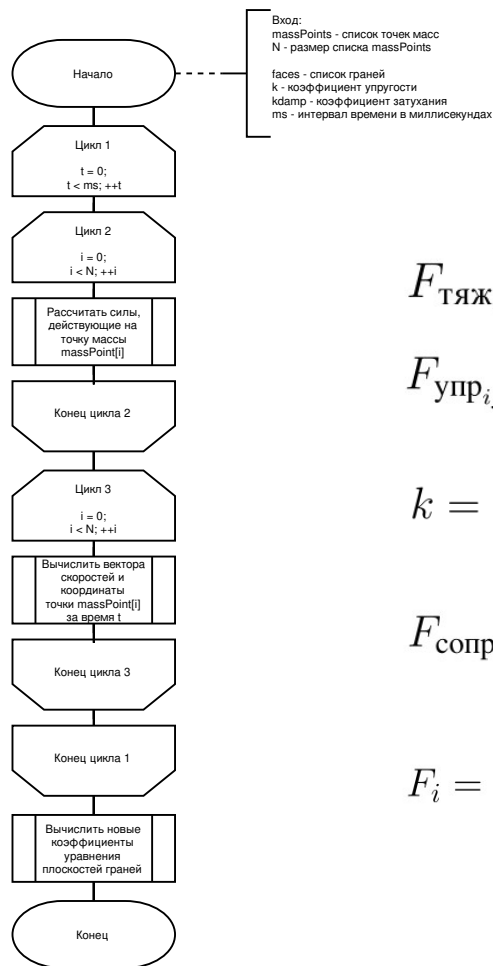
# Структура слайма



# Генерация слайма



# Расчет физических параметров слайма



$$F_{\text{тяж}_i} = m_i g$$

$$F_{\text{упр}_{ij}} = -k(|x_{ij}| - l_{ij}) \frac{x_{ij}}{|x_{ij}|}$$

$$k = \frac{k_e}{n} = \frac{k_e d_e}{d}$$

$$F_{\text{сопр}_{ij}} = -k_d \frac{(v_{ij}; x_{ij})}{(x_{ij}; x_{ij})} \frac{x_{ij}}{|x_{ij}|}$$

$$F_i = F_{\text{тяж}_i} + \sum_j^{K_i} (F_{\text{упр}_{ij}} + F_{\text{сопр}_{ij}})$$

$$a_i = \frac{F_i}{m_i}$$

$$v_i = v_{0i} + a_i t$$

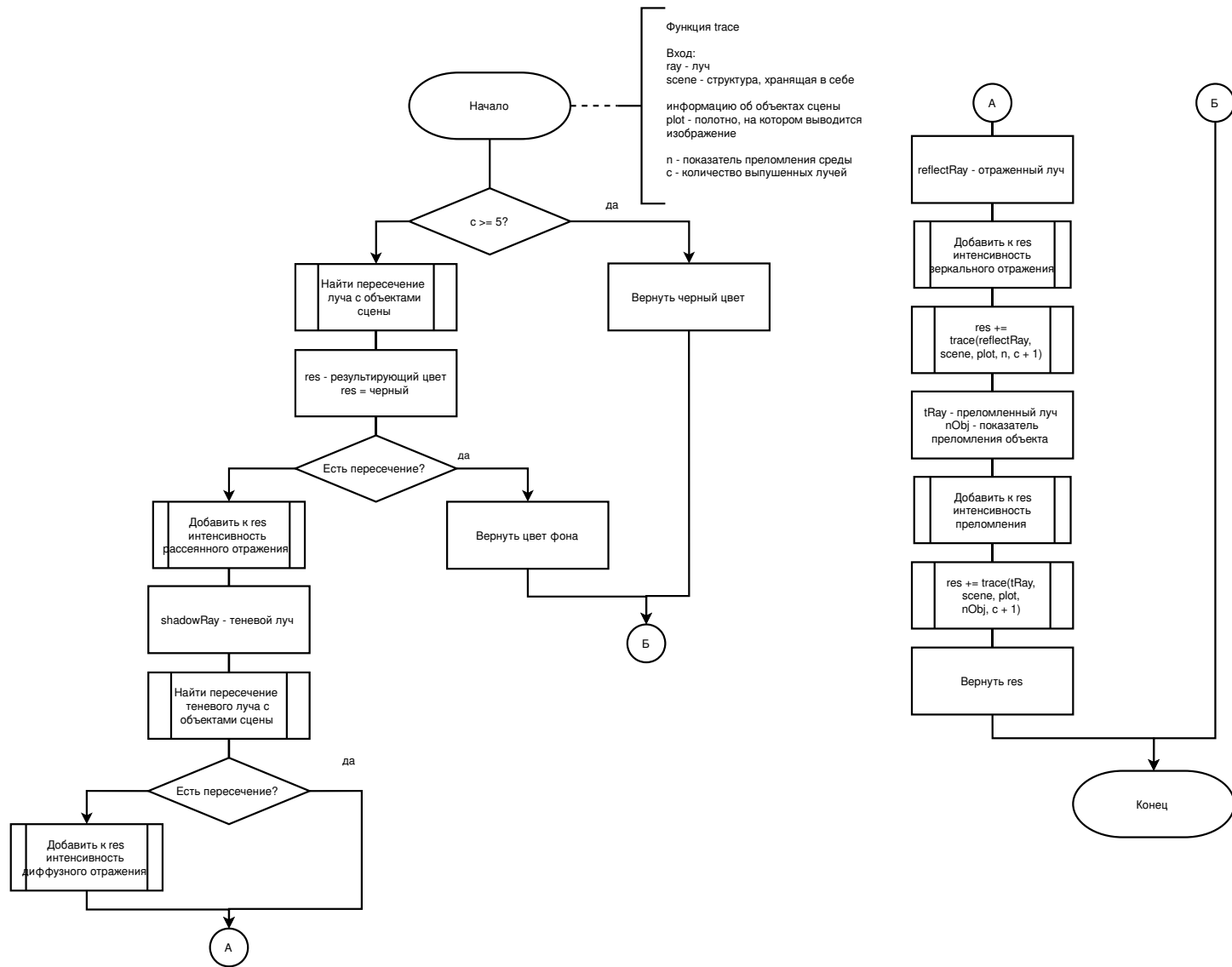
$$P_i(x, y, z) = P_{0i}(x_0, y_0, z_0) + v_{0i} t + a_i t^2$$

# Разработка рекурсивного алгоритма обратной трассировки лучей

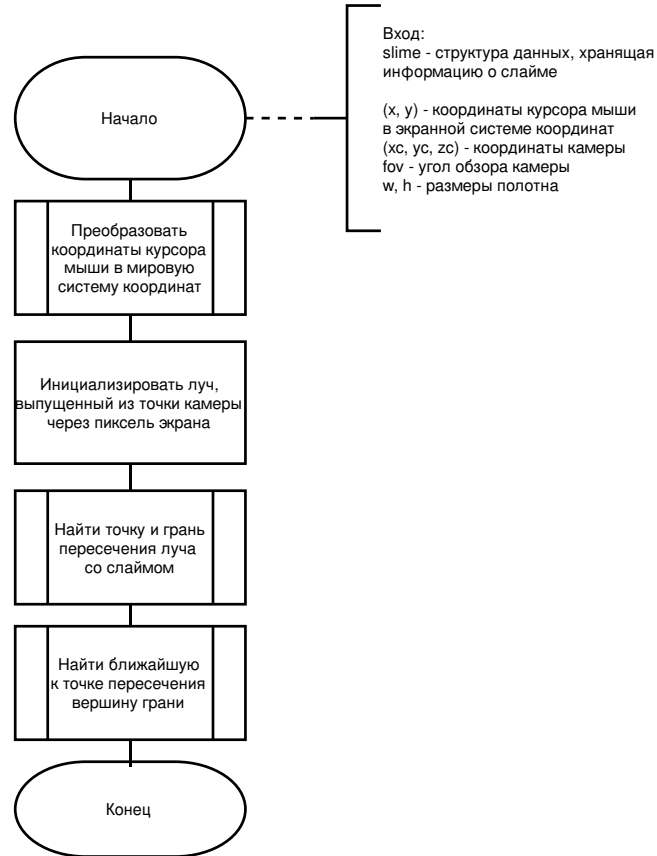
$$v_r = v - 2(n; v)n$$

$$v_t = \frac{n_0}{n}v - \left(\frac{n_0}{n}(N; v) + \cos\theta_t\right)N$$

$$\theta_t = \frac{n_0 \sin(\theta)}{n}$$



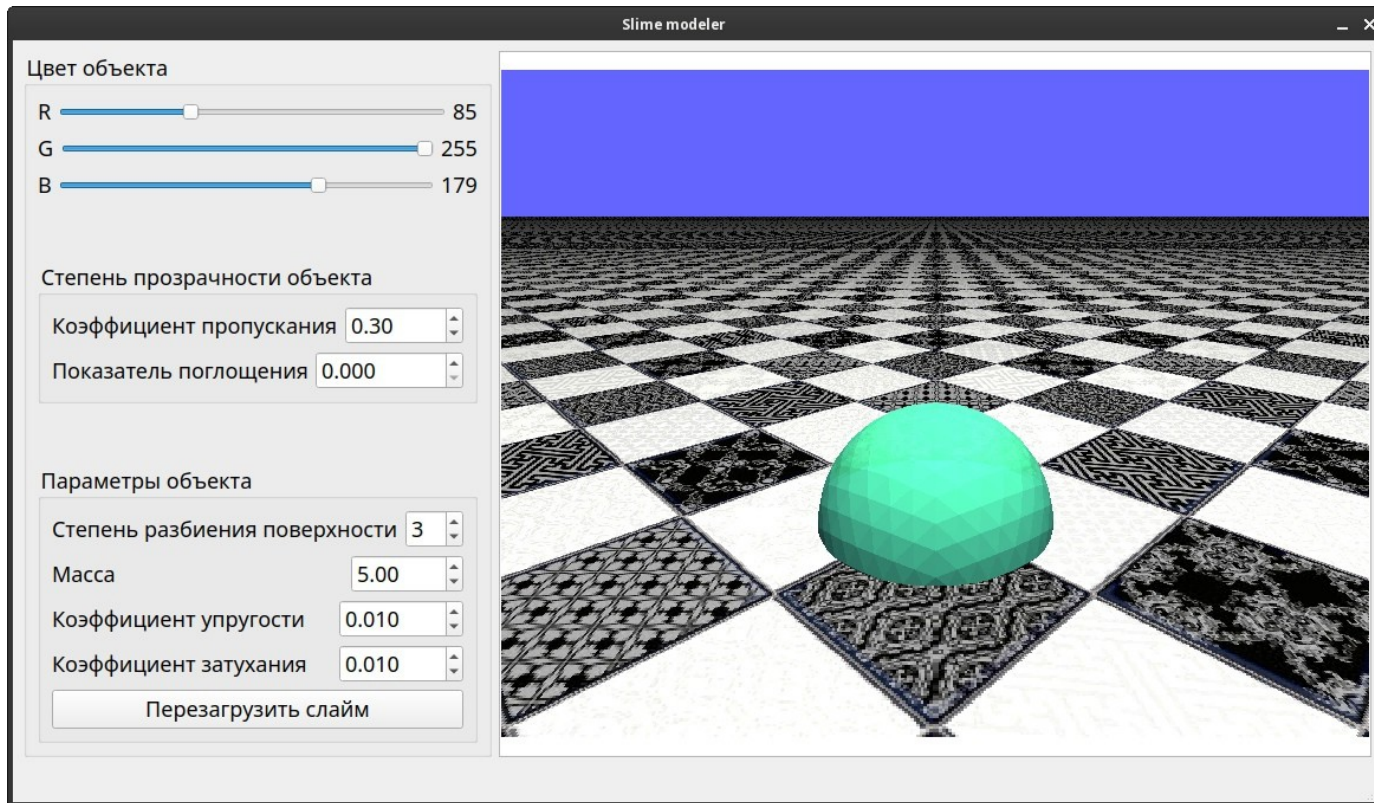
# Разработка алгоритма захвата точки слайма пользователем



# Инструменты разработки

- Язык программирования: C++
- Среда разработки: Visual Studio Code
- Фреймворк QT
- Прочие инструменты: qmake, pthread

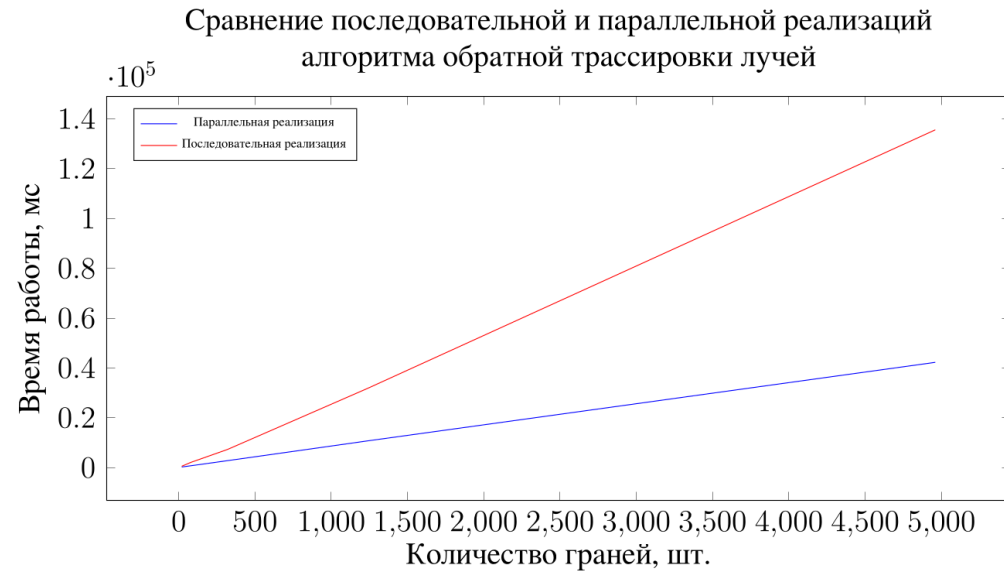
# Интерфейс программы





# Сравнение временной эффективности последовательной и параллельной реализаций обратной трассировки лучей

Количество граней объекта, шт.	Среднее время работы параллельной реализации, мс	Среднее время работы последовательной реализации, мс
20	233	536
80	739	2099
320	2788	7283
1240	10738	31847
4960	42253	135596



# Заключение

В ходе работы были выполнены следующие задачи:

- выбраны модель тела и алгоритмы, необходимые для реализации программы моделирования детской игрушки “Лизун”
- разработано соответствующее программное обеспечение
- сравнено среднее время работы параллельной и последовательной реализаций алгоритма удаления невидимых линий и поверхностей
- закреплены знания и навыки, приобретенные в ходе изучения курса компьютерной графики