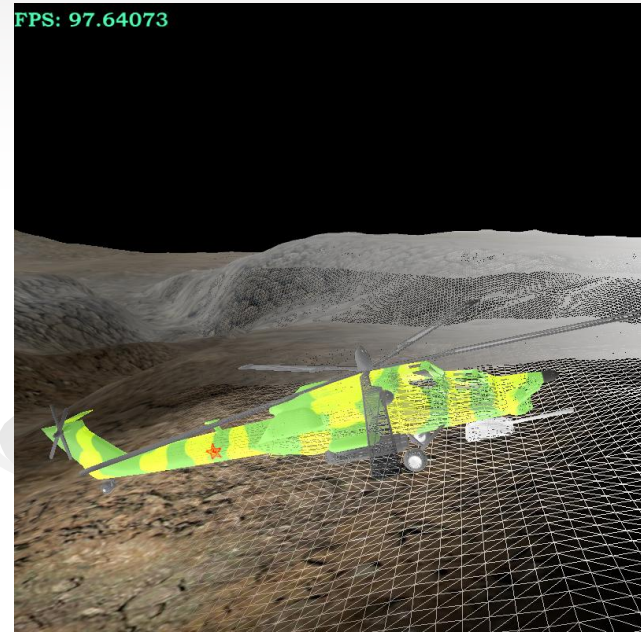


# Разработка системы анимации трехмерных сцен на GPU

Иванов Тимофей (8-1)	Файзуллин Музаффар (10-1)
Амбросовская Дарья (8-2)	Шабанов Никита (10-3)
Фокеев Борис (8-2)	Мосягин Олег (10-5)
Подкопаев Александр (8-4)	Сорин Алексей (10-5)
Уророва Софья (8-4)	Сорин Николай (10-5)
Григорович Вячеслав (9-5)	Филиппов Денис (10-5)
Крейнин Матвей (9-5)	Кожухаров Никита (10-6)
Писарев Евгений (9-5)	Пономаренко Ульяна (10-6)
Синяков Степан (9-5)	Тарасов Денис (10-6)



Научный руководитель: Галинский В.А.,

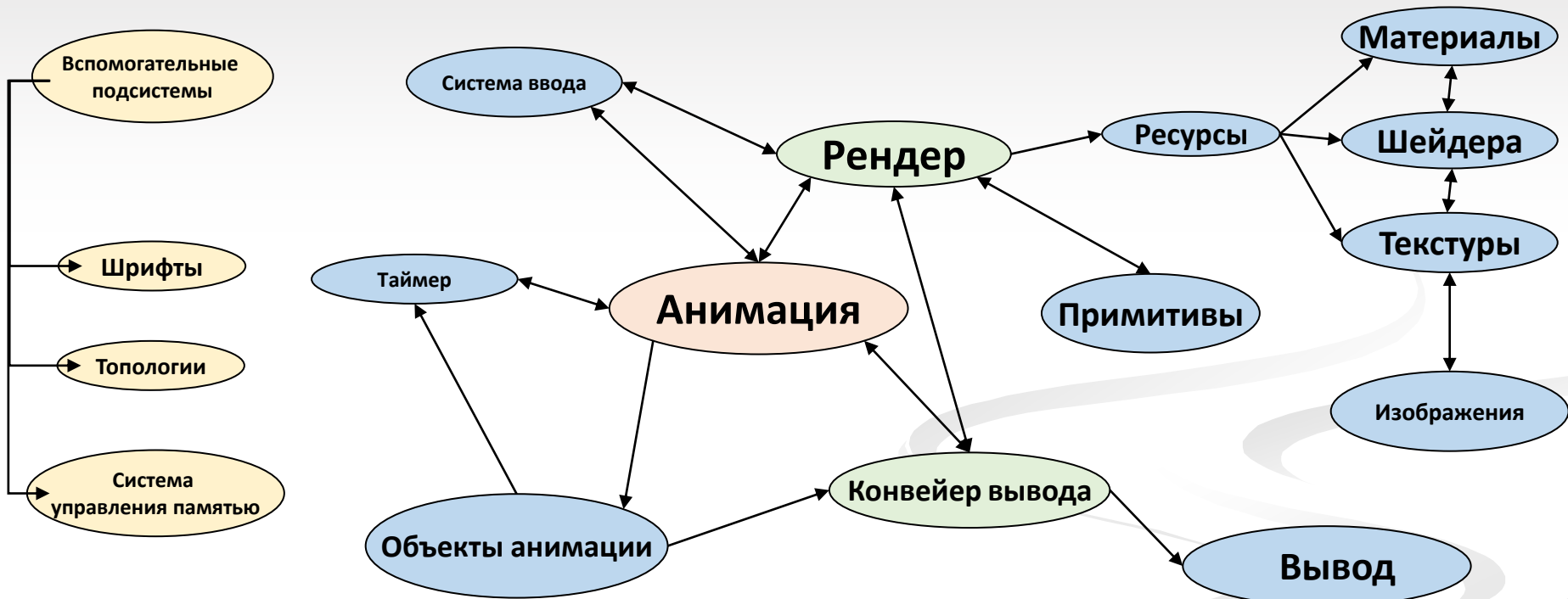
преподаватель информатики и программирования СПб губернаторского физико-математического лицея № 30

Computer Graphics Support Group

Санкт-Петербургский губернаторский физико-математический лицей № 30



# Структура проекта



# Система анимации

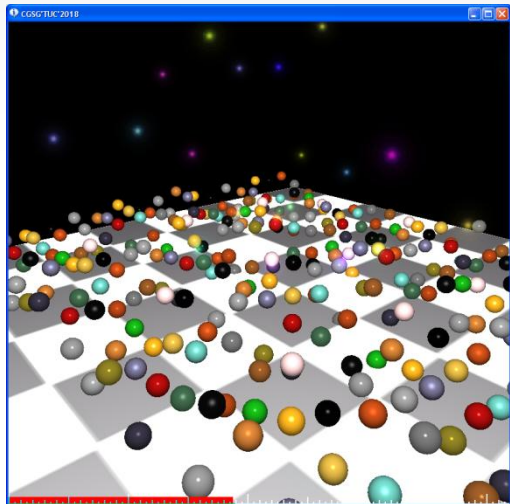
## Система управления объектами анимации

### Базовые поля структуры объектов анимации:

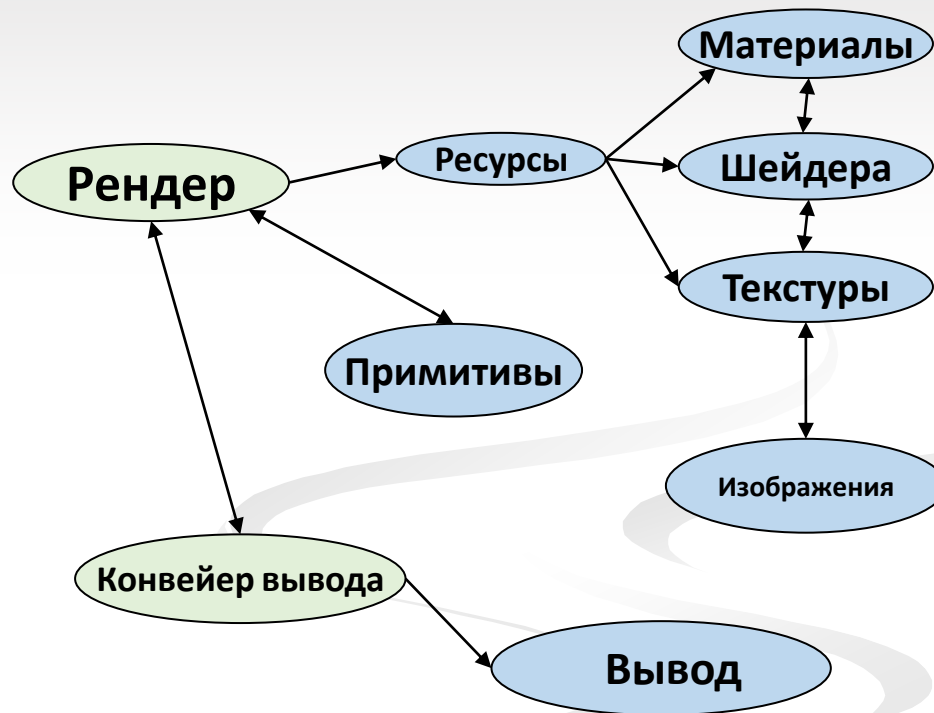
- Инициализация
- Деинициализация
- Обработка сообщений
- Отрисовка
- Регистрация отложенного вывода
- Очередь в списке объектов отложенной отрисовки

### Хранение объектов анимации:

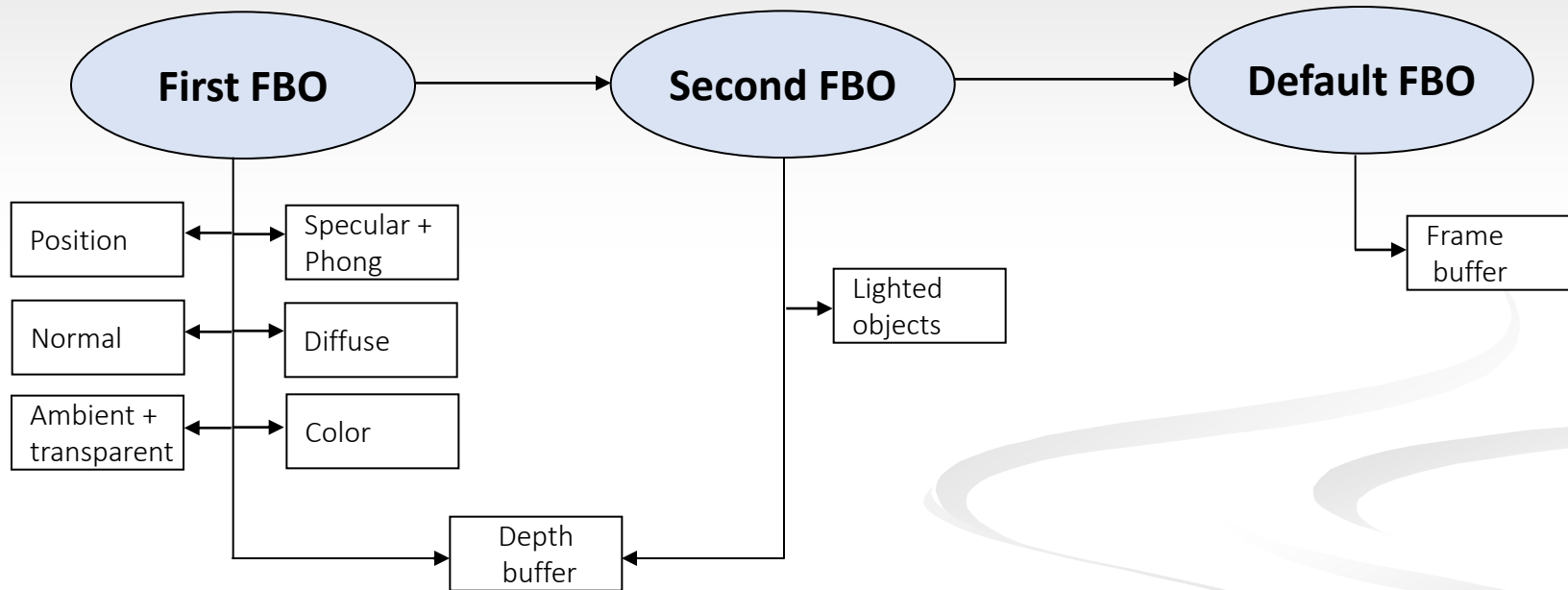
- Основной контейнер
- Контейнер объектов отложенного вывода
- Массив объектов отложенного вывода



# Система рендеринга



# Конвейер вывода

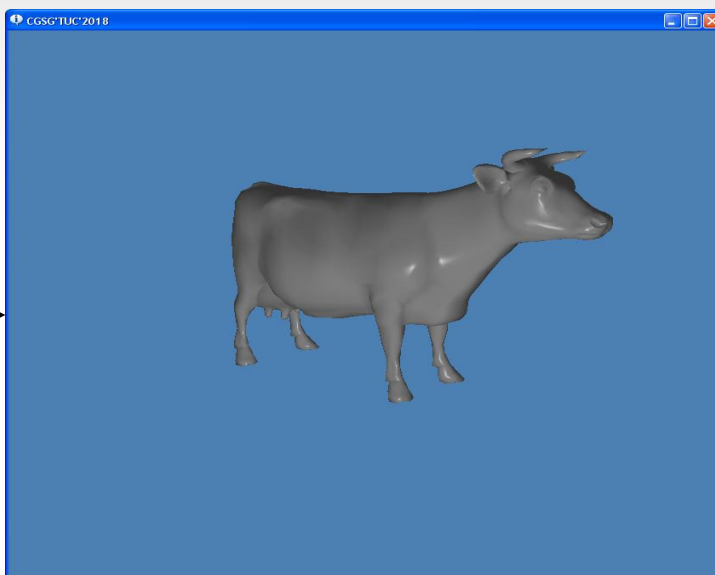




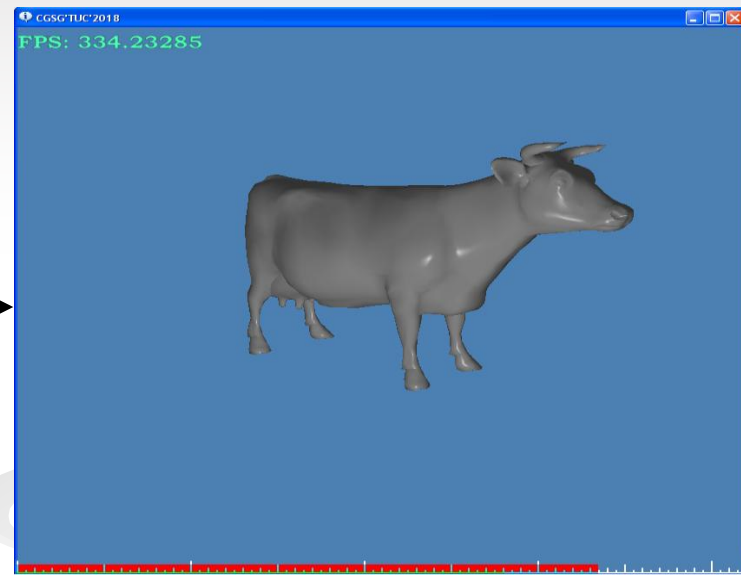
# Конвейер вывода: G-buffers



Заполнение  
буферов кадра



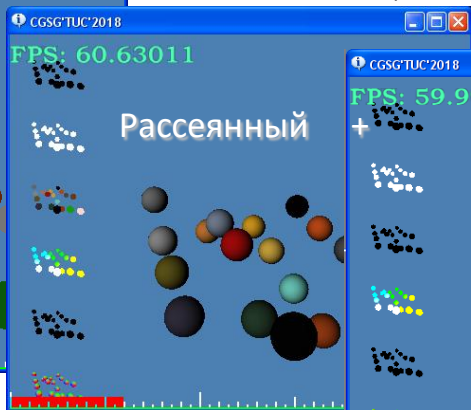
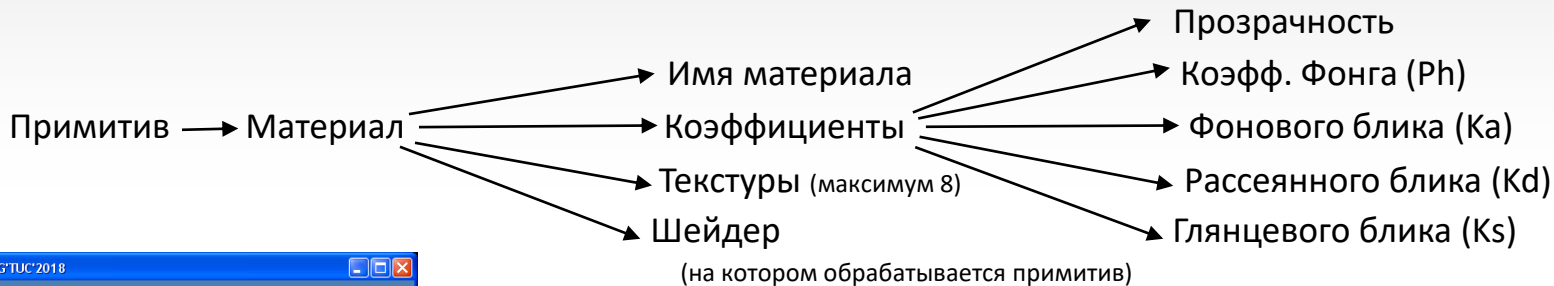
Вывод объектов и их освещение



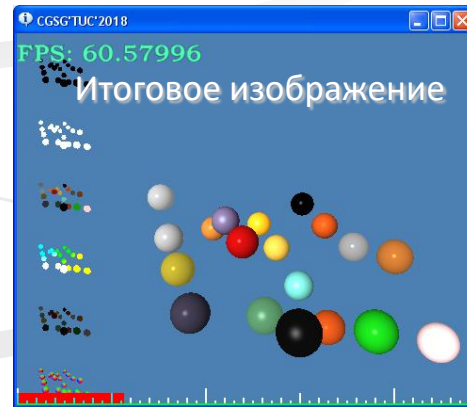
Отложенный вывод  
неосвещаемых объектов

# Материалы поверхности

- Каждый примитив по разному освещается и обрабатывается.



=



# Отложенное освещение

Заполняем буфер кадра (G-buffers)



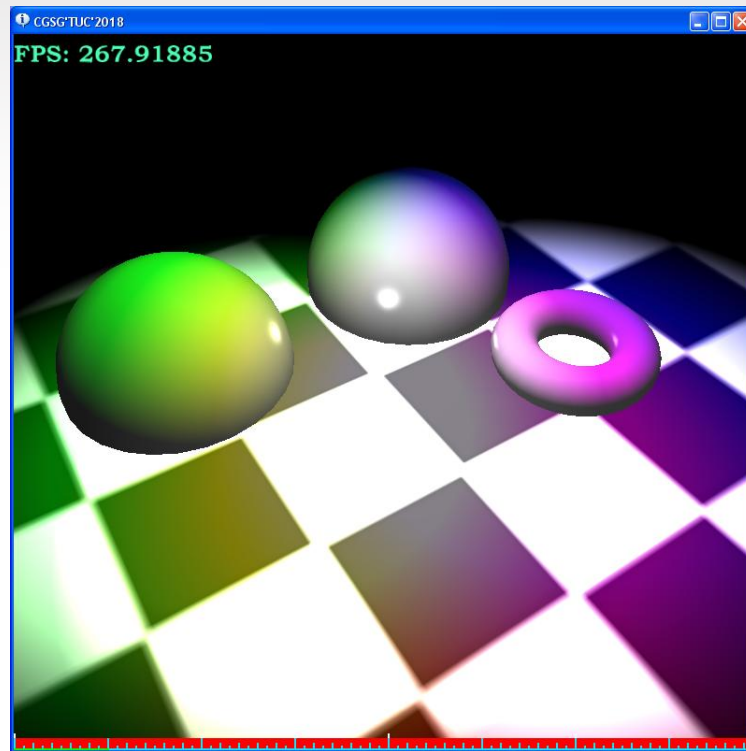
Обнуляем цветные плоскости



Включаем режим смешивания цветов



Применяем модель освещения для каждой точки на экране



Пример смешивания цветов  
конического источников света

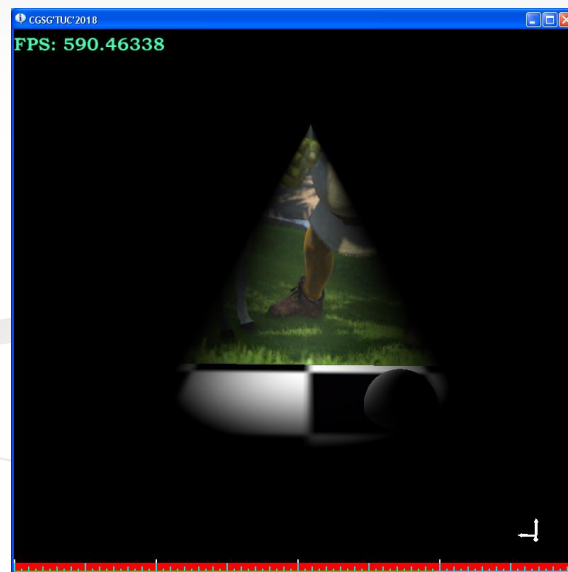
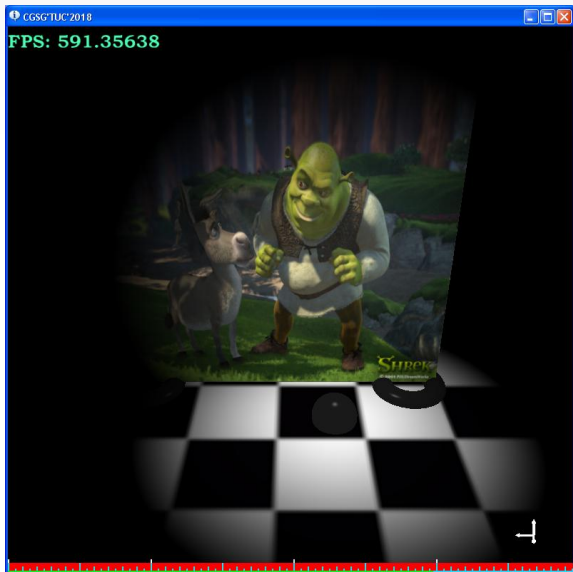


# Источники света

Точечный (Point)

Направленный (Direction)

Конический (Spot)



# Отложенное построение

*Геометрические объекты, выводимые в конце кадра*



Контейнер для отложенных объектов



Рисование в конце кадра

*Отложенное построение используют:*

- lights
- deferred primitives
- particles
- font
- lens flare



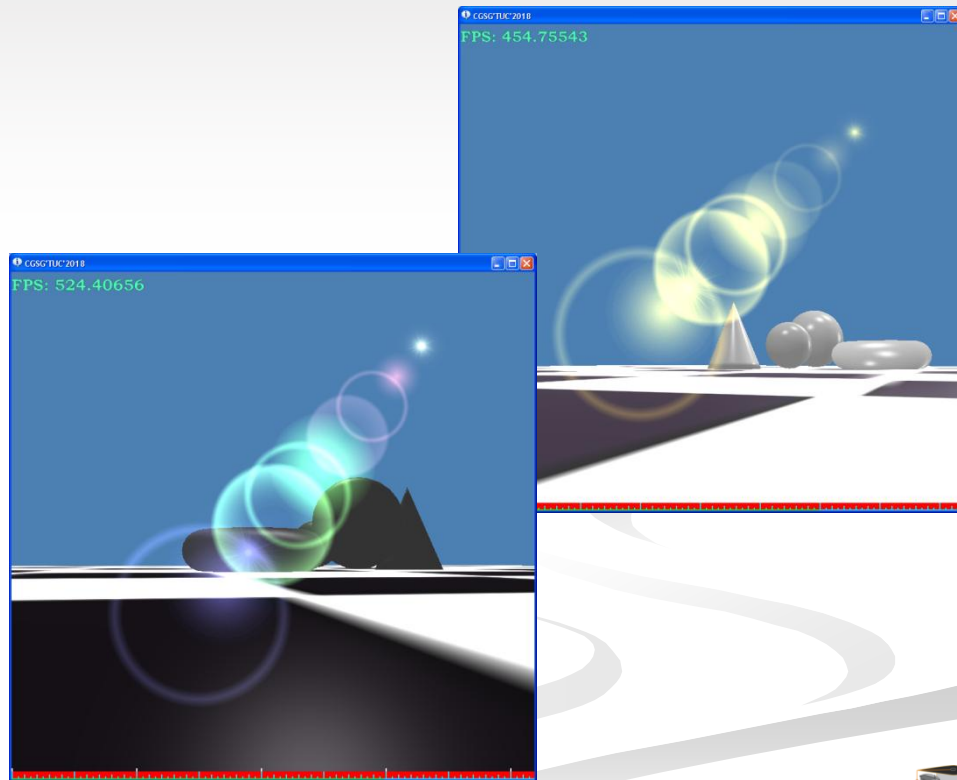
Computer Graphics Support Group

Санкт-Петербургский губернаторский физико-математический лицей № 30

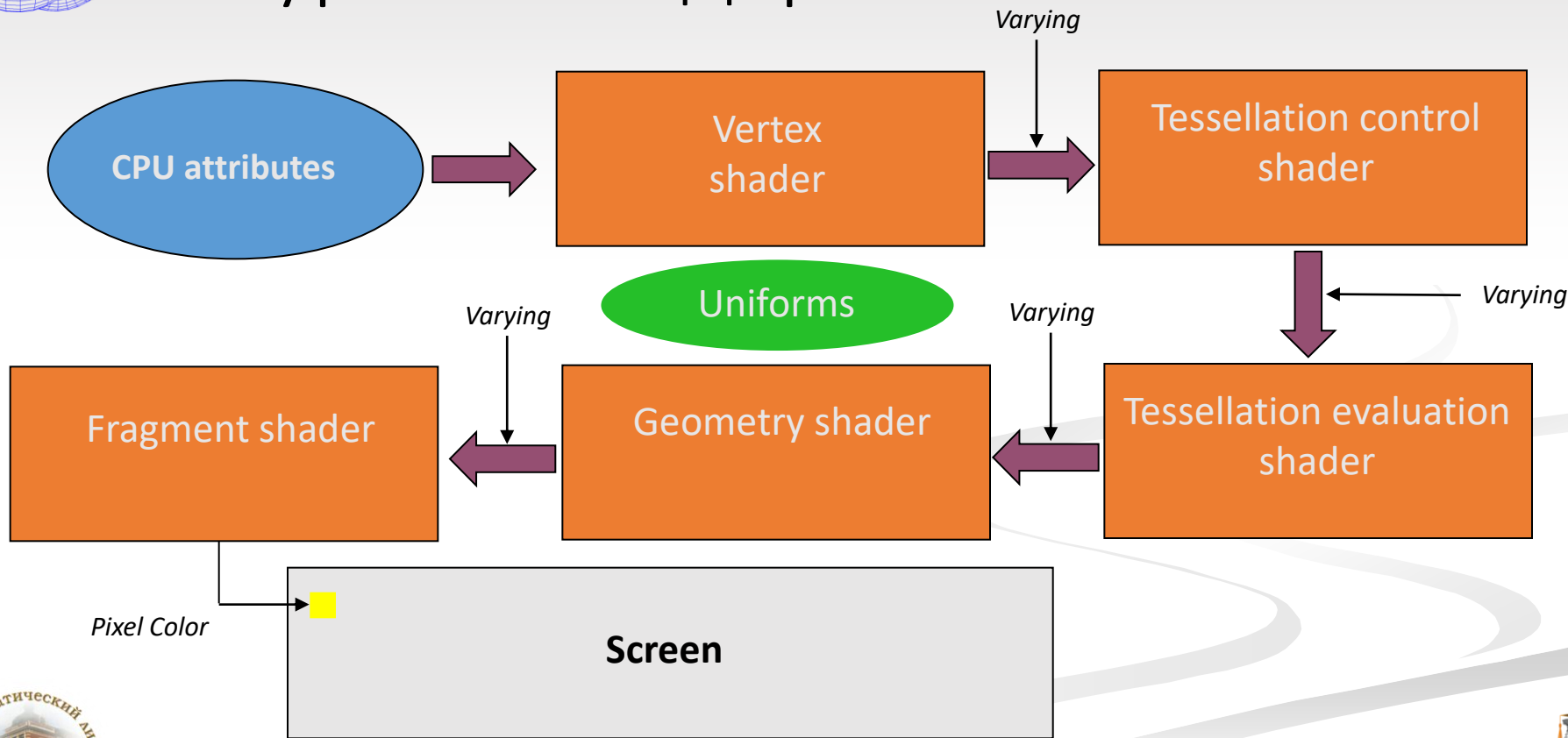
# Отложенное построение

## *Lens Flares*

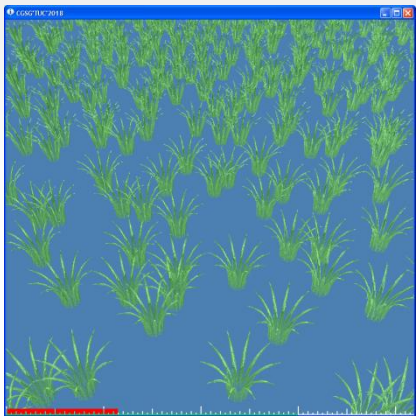
- Возможность вывода в плоскости экрана блика от источника света
- Придание реалистичности композиции
- Быстрое выполнение за счет выполнения построения на видеокарте



# Ресурсы: Шейдера



# Шейдера: *Vertex + Tessellation + Geometry + Fragment*



Вся трава  
подвергается  
процессу освещения.

**Vertex:** Прием и передача координат дальше по конвейеру.



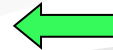
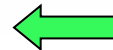
**Tessellation:** Построение ломанных линий, которые являются каркасом будущей травы.



**Geometry:** Вокруг этих ломанных генерируется оболочка, являющаяся корпусом травы.



**Fragment:** Заполнение геометрических буферов для последующего вывода.

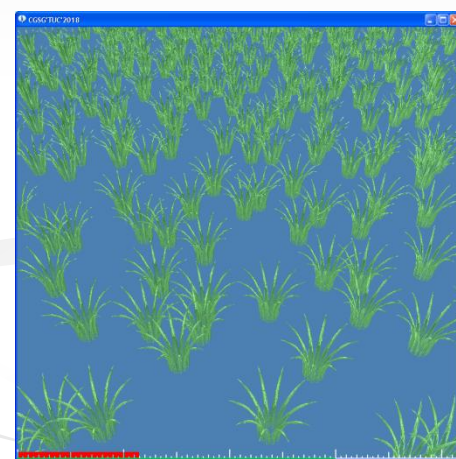
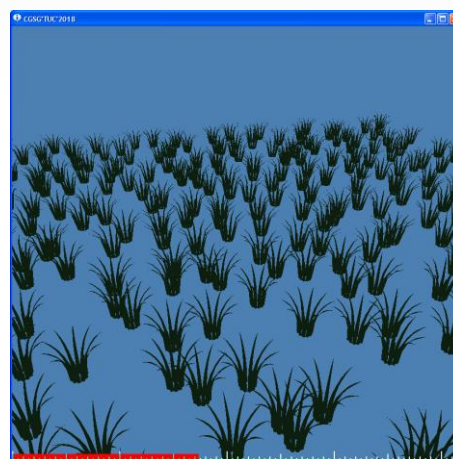
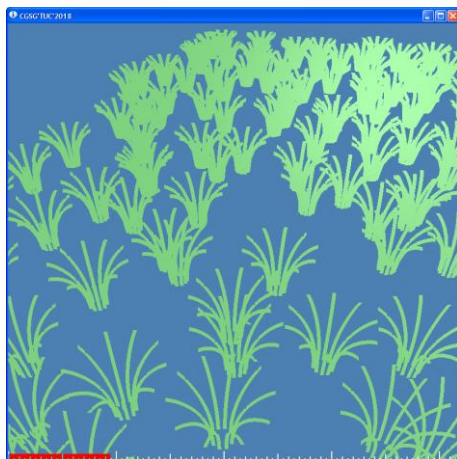
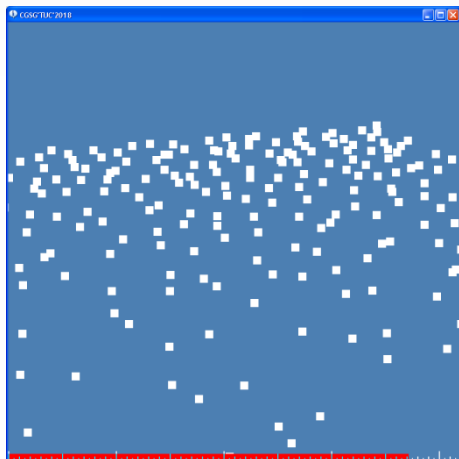


Адаптивная тесселяция.



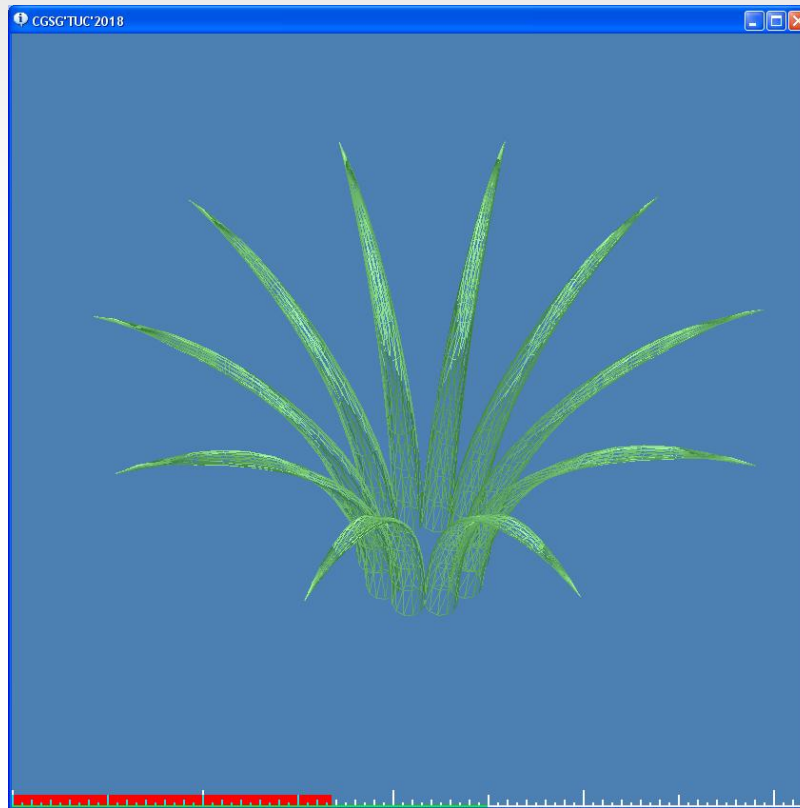
# Шейдера: *Vertex + Tessellation + Geometry + Fragment*

Поэтапная генерация травы.

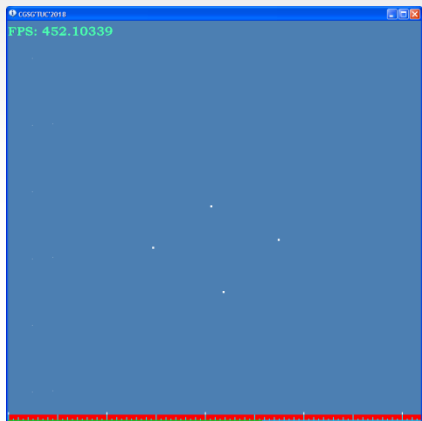


# Шейдера: *Vertex + Tessellation + Geometry + Fragment*

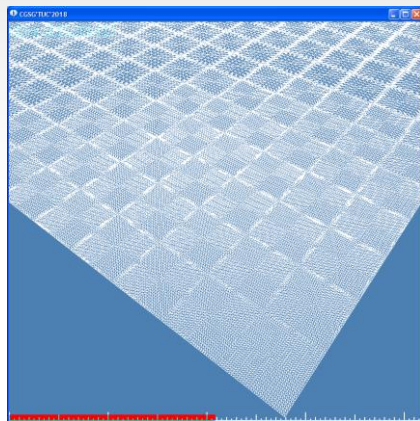
Адаптивная  
тесселяция



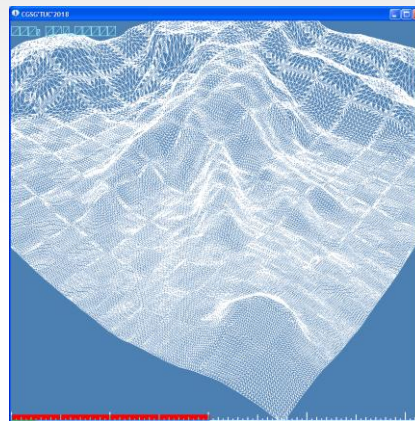
# Шейдера: *Vertex + Tessellation + Fragment*



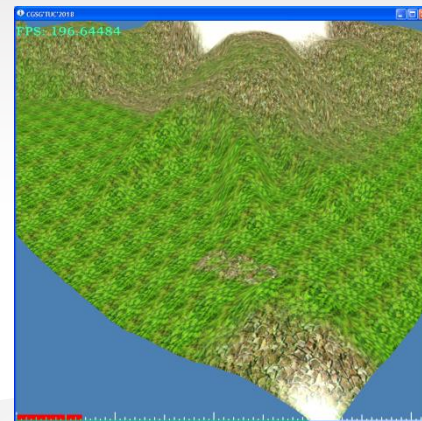
*Vertex*



*Control*



*Evaluation*



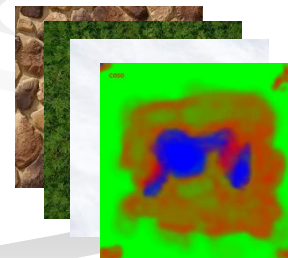
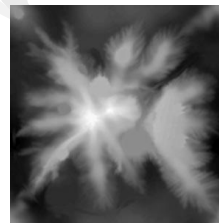
*Fragment*

Vert: Проброс вершин.

Ctrl: Задание количества разбиений.

Eval: Размещение вершин и расчет нормалей.

Frag: Наложение текстур по маске.





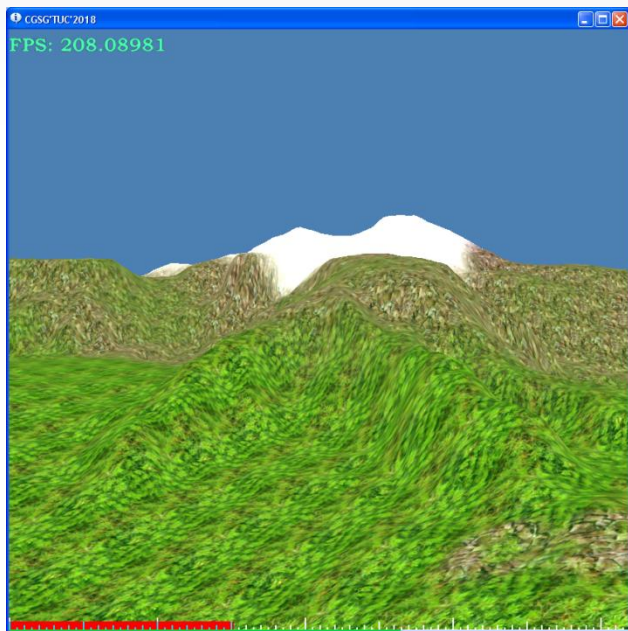
# Ресурсы: Текстуры

Используемые виды:

- 2D Textures
- 2D Textures x 6 (Cube-Map)

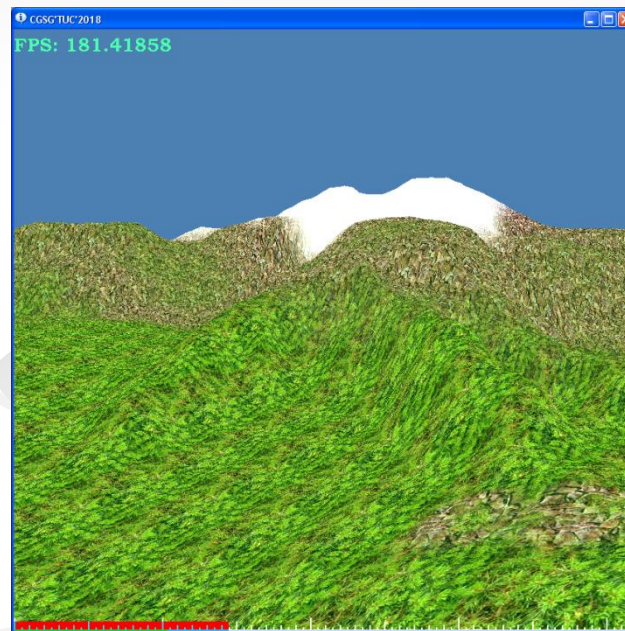
Поддерживаемые форматы:

- R32UI, RGBA8
- R32F, RGB32F, RGBA32F



Вкл.

MIP mapping



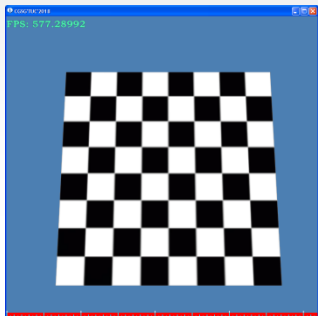
Выкл.

Computer Graphics Support Group

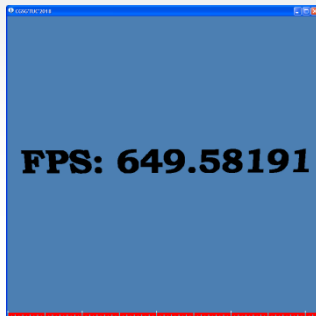
Санкт-Петербургский губернаторский физико-математический лицей № 30

# Текстуры

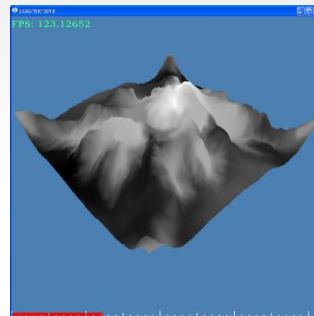
## Использование



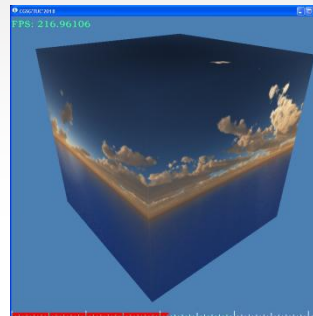
Текстурирование  
объектов



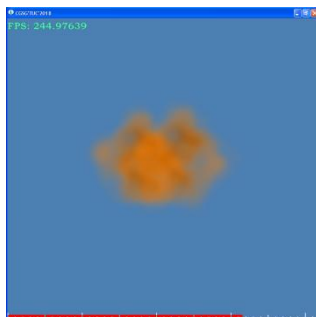
Вывод текста



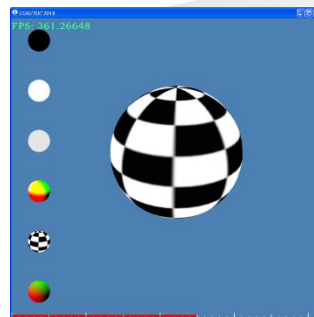
Построение  
карт высот



Skybox



Система частиц



FBO



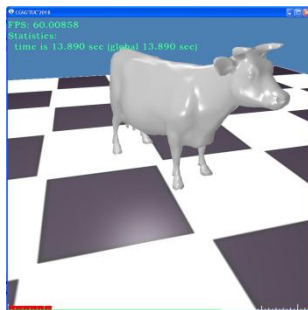
# Топологии

## Топологии

Points



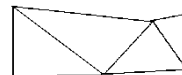
Triangle mesh



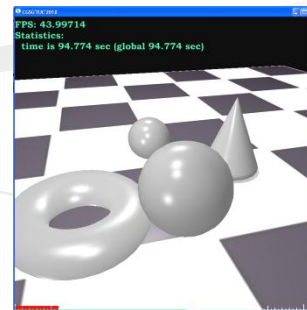
Line strip



Triangle strip



Grid

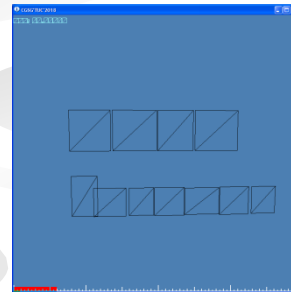
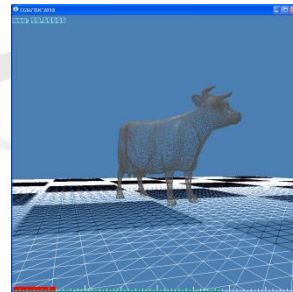
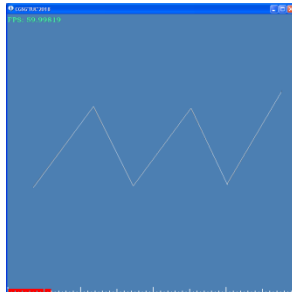
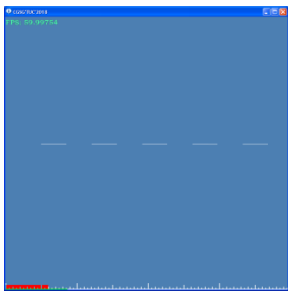
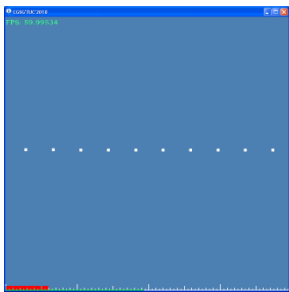
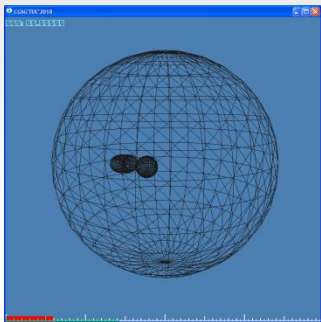
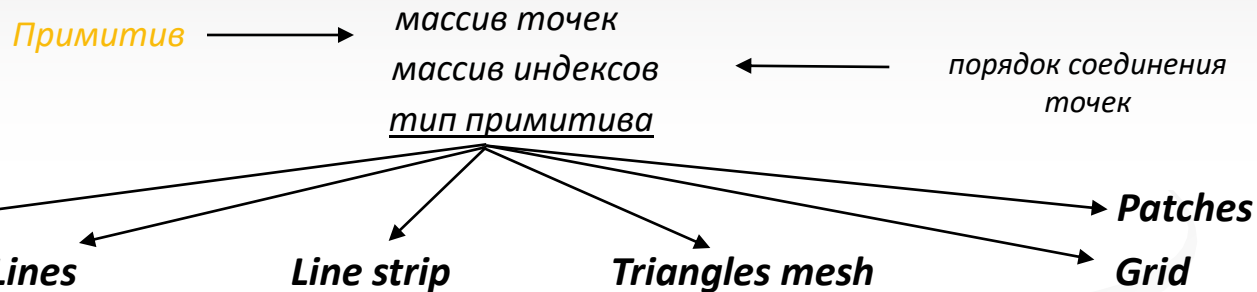


Computer Graphics Support Group

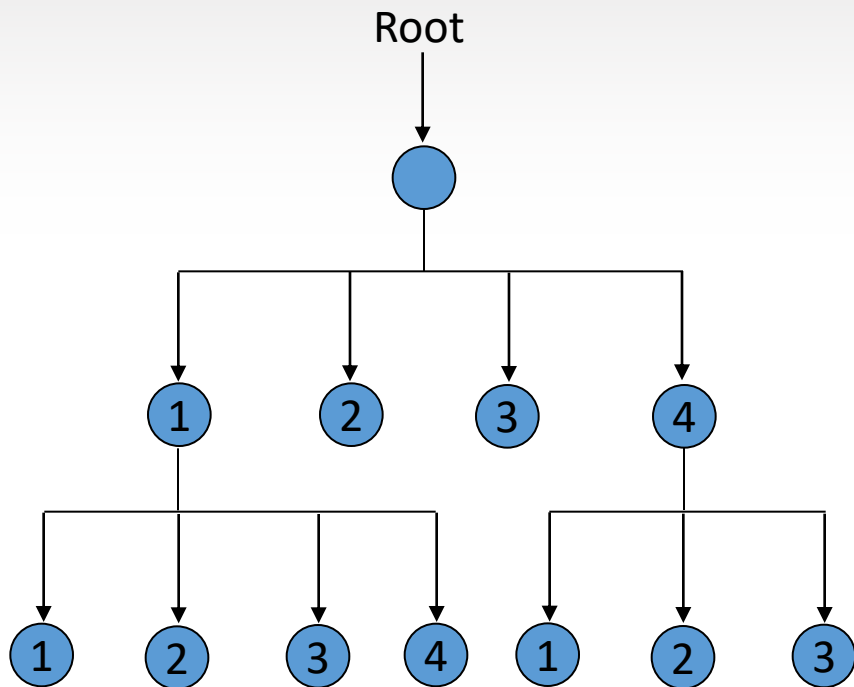
Санкт-Петербургский губернаторский физико-математический лицей № 30

# Примитивы

*Примитив* – это простейший геометрический объект.



# Контекстное моделирование



# Подсистема управления памятью



## Преимущества:

- Возможность хранения в едином пространстве
- Легкость выполнения работ с системой хранения

- Авторы: 18
- Языки: C/GLSL  
(вершинные, тесселяционные, геометрические, фрагментные шейдера)
- Время разработки: декабрь 2017-апрель 2018
- Размер кода: 600 KiB  
(60 x \*.c – 340 KiB, 30 x \*.h – 130 KiB, 80 x \*.glsl – 130 KiB)
- Строк кода: 21000  
(\*c – 11500, \*.h - 4700, \*.glsl – 4800)



# Разработка кода

- **Иванов Тимофей (8-1)**  
подсистемы ввода и синхронизации по времени, наложение теней
- **Амбросовская Дарья (8-2)**  
подсистема управления материалами
- **Фокеев Борис (8-2)**  
построение небесных оболочек
- **Подкопаев Александр (8-4)**  
прикладные примеры объектов анимации
- **Уросова Софья (8-4)**  
подсистема управления памятью, эффекты в плоскости экрана
- **Григорович Вячеслав (9-5)**  
подсистема вывода, иерархическое (контекстное) моделирование
- **Крейнин Матвей (9-5)**  
подсистема освещения
- **Писарев Евгений (9-5)**  
подсистема вывода, иерархическое (контекстное) моделирование
- **Синяков Степан (9-5)**  
подсистема управления примитивами
- **Файзуллин Музаффар (10-1)**  
подсистема управления анимацией
- **Шабанов Никита (10-3)**  
система вывода в геометрические буфера, экранные эффекты
- **Мосягин Олег (10-5)**  
подсистема управления шейдерами, процедурно генерируемые объекты
- **Сорин Алексей (10-5)**  
подсистема топологий
- **Сорин Николай (10-5)**  
подсистема управления изображениями
- **Филиппов Денис (10-5)**  
система вывода в геометрические буфера, визуализация частиц
- **Кожухаров Никита (10-6)**  
библиотека пространственной математики, объекты управления камерой
- **Пономаренко Ульяна (10-6)**  
прикладные примеры объектов анимации
- **Тарасов Денис (10-6)**  
подсистема управления текстурами, тесселяционные объекты