

# Компьютерная графика

Лекция 13: состояние OpenGL (напоминание), матрицы проекций (напоминание), рендеринг в subwindow, дистрибуция приложений на OpenGL

2021

# Настройка графического конвейера

- ▶ Графический конвейер (pipeline) - набор всех операций, происходящих с данными от момента вызова `glDraw*` до появления пикселей на экране (или текстуре/рендербуфере)

# Настройка графического конвейера

- ▶ Графический конвейер (pipeline) - набор всех операций, происходящих с данными от момента вызова `glDraw*` до появления пикселей на экране (или текстуре/рендербуфере)
- ▶ Графический конвейер = programmable pipeline + fixed-function pipeline

# Настройка графического конвейера

- ▶ Графический конвейер (pipeline) - набор всех операций, происходящих с данными от момента вызова `glDraw*` до появления пикселей на экране (или текстуре/рендербуфере)
- ▶ Графический конвейер = programmable pipeline + fixed-function pipeline
- ▶ Настройка programmable pipeline: шейдеры (шейдерная программа)

# Настройка графического конвейера

- ▶ Графический конвейер (pipeline) - набор всех операций, происходящих с данными от момента вызова `glDraw*` до появления пикселей на экране (или текстуре/рендербуфере)
- ▶ Графический конвейер = programmable pipeline + fixed-function pipeline
- ▶ Настройка programmable pipeline: шейдеры (шейдерная программа)
- ▶ Настройка fixed-function pipeline: включение/выключение (`glEnable/glDisable`) конкретных операций и их специфическая настройка

# Настройка fixed-function pipeline

- ▶ Depth clamp
  - ▶ По умолчанию, все примитивы обрезаются по уравнению  $z \leq |w|$
  - ▶ Можно заменить обрезание clamping'ом через `glEnable(GL_DEPTH_CLAMP)`

# Настройка fixed-function pipeline

- ▶ Depth clamp
  - ▶ По умолчанию, все примитивы обрезаются по уравнению  $z \leq |w|$
  - ▶ Можно заменить обрезание clamping'ом через `glEnable(GL_DEPTH_CLAMP)`
- ▶ Culling
  - ▶ Можно не рисовать back-facing или front-facing полигоны
  - ▶ Включить: `glEnable(GL_CULL_FACE)`
  - ▶ Настроить, что **не** рисуется: `glCullFace`
  - ▶ Настроить, что считается back-facing, а что front-facing: `glFrontFace`

# Настройка fixed-function pipeline

- ▶ Depth clamp
  - ▶ По умолчанию, все примитивы обрезаются по уравнению  $z \leq |w|$
  - ▶ Можно заменить обрезание clamping'ом через `glEnable(GL_DEPTH_CLAMP)`
- ▶ Culling
  - ▶ Можно не рисовать back-facing или front-facing полигоны
  - ▶ Включить: `glEnable(GL_CULL_FACE)`
  - ▶ Настроить, что **не** рисуется: `glCullFace`
  - ▶ Настроить, что считается back-facing, а что front-facing: `glFrontFace`
- ▶ Viewport
  - ▶ Настроить перевод из NDC (normalized device coordinates, [-1..1]) в пиксельные координаты: `glViewport`
  - ▶ Обычно нужно делать каждый раз при изменении размеров окна или при переключении фреймбуферов



# Настройка fixed-function pipeline

## ▶ Depth test

- ▶ Можно не рисовать пиксели, находящиеся сзади уже нарисованных пикселей
- ▶ Включить: `glEnable(GL_DEPTH_TEST)`
- ▶ Настроить: `glDepthFunc`
- ▶ Настроить преобразование из NDC в  $[0, 1]$ : `glDepthRangef`
- ▶ Включить/выключить запись значений глубины: `glDepthMask`

# Настройка fixed-function pipeline

## ▶ Depth test

- ▶ Можно не рисовать пиксели, находящиеся сзади уже нарисованных пикселей
- ▶ Включить: `glEnable(GL_DEPTH_TEST)`
- ▶ Настроить: `glDepthFunc`
- ▶ Настроить преобразование из NDC в  $[0, 1]$ : `glDepthRange`
- ▶ Включить/выключить запись значений глубины: `glDepthMask`

## ▶ Stencil test

- ▶ Включить: `glEnable(GL_STENCIL_TEST)`
- ▶ Настроить: `glStencilFunc`, `glStencilOp`, `glStencilMask`

# Настройка fixed-function pipeline

- ▶ Depth test
  - ▶ Можно не рисовать пиксели, находящиеся сзади уже нарисованных пикселей
  - ▶ Включить: `glEnable(GL_DEPTH_TEST)`
  - ▶ Настроить: `glDepthFunc`
  - ▶ Настроить преобразование из NDC в  $[0, 1]$ : `glDepthRangef`
  - ▶ Включить/выключить запись значений глубины: `glDepthMask`
- ▶ Stencil test
  - ▶ Включить: `glEnable(GL_STENCIL_TEST)`
  - ▶ Настроить: `glStencilFunc`, `glStencilOp`, `glStencilMask`
- ▶ Scissor test
  - ▶ Можно не рисовать пиксели, находящиеся вне некоторого прямоугольника
  - ▶ Включить: `glEnable(GL_SCISSOR_TEST)`
  - ▶ Настроить: `glScissor`

# Настройка fixed-function pipeline

- ▶ Color mask
  - ▶ Настроить запись в конкретные цветовые каналы:  
`glColorMask`

# Настройка fixed-function pipeline

- ▶ Color mask
  - ▶ Настроить запись в конкретные цветовые каналы:  
`glColorMask`
- ▶ Blending
  - ▶ Можно записывать значение некоторой функции от входного цвета и уже записанного цвета
  - ▶ Включить: `glEnable(GL_BLEND)`
  - ▶ Настроить: `glBlendFunc/glBlendFuncSeparate`, `glBlendEquation`, `glBlendColor`

# Настройка fixed-function pipeline

- ▶ Color mask
  - ▶ Настроить запись в конкретные цветовые каналы: `glColorMask`
- ▶ Blending
  - ▶ Можно записывать значение некоторой функции от входного цвета и уже записанного цвета
  - ▶ Включить: `glEnable(GL_BLEND)`
  - ▶ Настроить: `glBlendFunc/glBlendFuncSeparate`, `glBlendEquation`, `glBlendColor`
- ▶ Color logical operation
  - ▶ Можно записывать результат некоторой побитовой операции от входного цвета и уже записанного цвета
  - ▶ Выключает blending
  - ▶ Включить: `glEnable(GL_COLOR_LOGIC_OP)`
  - ▶ Настроить: `glLogicOp`