

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования «Ростовский государственный экономический университет (РИНХ)»
Документ подписан: 18.04.2024 08:53:15
Информация о владельце:
ФИО: Макаренко Елена Николаевна
Должность: Ректор
Уникальный программный ключ:
c098bc0c1041cb2a4cf926cf171d6715d99a6ae00adc8e27b55cbe1e2dbd7c78

УТВЕРЖДАЮ
Директор Института магистратуры
Иванова Е.А.
«01» июня 2023г.

**Рабочая программа дисциплины
Программирование трехмерной графики**

Направление 09.04.04 Программная инженерия
магистерская программа 09.04.04.01 "Системное и прикладное программное обеспечение"

Для набора 2023 года

Квалификация
магистр

КАФЕДРА Информационные технологии и защита информации

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	1 (1.1)		Итого	
	УП	РП	УП	РП
Неделя	15 2/6			
Вид занятий	УП	РП	УП	РП
Лекции	8	8	8	8
Лабораторные	16	16	16	16
Практические	8	8	8	8
Итого ауд.	32	32	32	32
Контактная работа	32	32	32	32
Сам. работа	76	76	76	76
Итого	108	108	108	108

ОСНОВАНИЕ

Учебный план утвержден учёным советом вуза от 28.03.2023 протокол № 9.

Программу составил(и): к.э.н., доц., Плетняков В.А.

Зав. кафедрой: к.э.н., доц. Ефимова Е.В.

Методическим советом направления: д.э.н., проф., Тищенко Е.Н.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1	Изучение принципов работы с трехмерной компьютерной графикой, подходов и методов программирования для овладения знаниями в области разработки приложений, работающих с трехмерной графикой.
-----	---

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ПК-3 :Способен самостоятельно осуществлять руководство процессами разработки программного обеспечения

ПК-4 :Способен осуществлять управление программно-техническими и технологическими ресурсами

В результате освоения дисциплины обучающийся должен:

Знать:
методы и приемы алгоритмизации поставленных задач, методологии разработки программного обеспечения, основные принципы и методы управления персоналом (соотнесено с индикатором ПК-3.1); методологию управления проектами разработки программного обеспечения, лучшие практики управления разработкой программного обеспечения, принципы и методы управления персоналом (соотнесено с индикатором ПК-4.1)
Уметь:
использовать методы и приемы формализации задач, использовать выбранную среду программирования, применять коллективную среду разработки программного обеспечения и систему контроля версий, применять методы принятия управленческих решений (соотнесено с индикатором ПК-3.2); применять методы и средства оценки сложности, трудоемкости и сроков выполнения работ, применять методы и средства организации проектных данных, применять основные принципы и методы управления персоналом (соотнесено с индикатором ПК-4.2)
Владеть:
способами оценки результатов проверки работоспособности программного обеспечения, управленческими решениями по результатам проверки работоспособности выпусков программного продукта (соотнесено с индикатором ПК- 3.3); способами мониторинга и оценки по выбранным критериям (показателям) сложности, трудоемкости и сроков выполнения работ, методами принятия управленческих решений (соотнесено с индикатором ПК-4.3)

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетенции	Литература
	Раздел 1. Основы программирования трехмерной графики				
1.1	Тема 1.1. Основные понятия компьютерной графики. Понятие топологии объекта. Вершины, ребра, полигоны. Нормали. Координатный метод построения изображений. Примеры визуализации трехмерных объектов: шар, цилиндр, тор. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.2	Тема 1.1. Основные понятия компьютерной графики. Понятие топологии объекта. Вершины, ребра, полигоны. Нормали. Координатный метод построения изображений. Примеры визуализации трехмерных объектов: шар, цилиндр, тор. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.3	Тема 1.1. Основные понятия компьютерной графики. Понятие топологии объекта. Вершины, ребра, полигоны. Нормали. Координатный метод построения изображений. Примеры визуализации трехмерных объектов: шар, цилиндр, тор. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.4	Тема 1.1. Основные понятия компьютерной графики. Понятие топологии объекта. Вершины, ребра, полигоны. Нормали. Координатный метод построения изображений. Примеры визуализации трехмерных объектов: шар, цилиндр, тор. /Ср/	1	9	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.5	Тема 1.2. Структуры данных 3D-моделей. Виды полигонов: треугольник, квад, N-гон (N-Gon). Преобразования координат и объектов. Проекция. Текстуры. UV-развертка модели. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3

1.6	Тема 1.2. Структуры данных 3D-моделей. Виды полигонов: треугольник, квадрат, N-гон (N-Gon). Преобразования координат и объектов. Проекция. Текстуры. UV-развертка модели. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.7	Тема 1.2. Структуры данных 3D-моделей. Виды полигонов: треугольник, квадрат, N-гон (N-Gon). Преобразования координат и объектов. Проекция. Текстуры. UV-развертка модели. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.8	Тема 1.2. Структуры данных 3D-моделей. Виды полигонов: треугольник, квадрат, N-гон (N-Gon). Преобразования координат и объектов. Проекция. Текстуры. UV-развертка модели. /Ср/	1	9	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.9	Тема 1.3. Алгоритмы и методы трехмерной графики. Модели описания поверхностей. Модели освещения: Ламберт, Блинн-Фонг, физически корректный рендеринг (PBR). Визуализация объемных изображений. Закрашивание трехмерных поверхностей. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.10	Тема 1.3. Алгоритмы и методы трехмерной графики. Модели описания поверхностей. Модели освещения: Ламберт, Блинн-Фонг, физически корректный рендеринг (PBR). Визуализация объемных изображений. Закрашивание трехмерных поверхностей. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.11	Тема 1.3. Алгоритмы и методы трехмерной графики. Модели описания поверхностей. Модели освещения: Ламберт, Блинн-Фонг, физически корректный рендеринг (PBR). Визуализация объемных изображений. Закрашивание трехмерных поверхностей. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
1.12	Тема 1.3. Алгоритмы и методы трехмерной графики. Модели описания поверхностей. Модели освещения: Ламберт, Блинн-Фонг, физически корректный рендеринг (PBR). Визуализация объемных изображений. Закрашивание трехмерных поверхностей. /Ср/	1	9	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
Раздел 2. Разработка шейдеров					
2.1	Тема 2.1. Разработка графических программ. Языки CG, GLSL и HLSL и используемые ими структуры данных. Использование моделей освещения. Использование карт нормалей. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.2	Тема 2.1. Разработка графических программ. Языки CG, GLSL и HLSL и используемые ими структуры данных. Использование моделей освещения. Использование карт нормалей. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.3	Тема 2.1. Разработка графических программ. Языки CG, GLSL и HLSL и используемые ими структуры данных. Использование моделей освещения. Использование карт нормалей. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.4	Тема 2.1. Разработка графических программ. Языки CG, GLSL и HLSL и используемые ими структуры данных. Использование моделей освещения. Использование карт нормалей. /Ср/	1	9	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.5	Тема 2.2. Проходы. Кадровый буфер (Frame Buffer), буфер глубины (Z Buffer). Очередь визуализации (Render Queue). Положение камеры и ориентация нормалей полигонов, их скалярное произведение. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.6	Тема 2.2. Проходы. Кадровый буфер (Frame Buffer), буфер глубины (Z Buffer). Очередь визуализации (Render Queue). Положение камеры и ориентация нормалей полигонов, их скалярное произведение. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3

2.7	Тема 2.2. Проходы. Кадровый буфер (Frame Buffer), буфер глубины (Z Buffer). Очередь визуализации (Render Queue). Положение камеры и ориентация нормалей полигонов, их скалярное произведение. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.8	Тема 2.2. Проходы. Кадровый буфер (Frame Buffer), буфер глубины (Z Buffer). Очередь визуализации (Render Queue). Положение камеры и ориентация нормалей полигонов, их скалярное произведение. /Ср/	1	10	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.9	Тема 2.3. Создание шейдерных программ при разработке компьютерных игр. Использование шейдерных программ в Unity. Язык ShaderLab. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.10	Тема 2.3. Создание шейдерных программ при разработке компьютерных игр. Использование шейдерных программ в Unity. Язык ShaderLab. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.11	Тема 2.3. Создание шейдерных программ при разработке компьютерных игр. Использование шейдерных программ в Unity. Язык ShaderLab. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.12	Тема 2.3. Создание шейдерных программ при разработке компьютерных игр. Использование шейдерных программ в Unity. Язык ShaderLab. /Ср/	1	10	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.13	Тема 2.4. Написание шейдеров в Unity. Отражения. Альфа-канал. Прозрачность. Вершинный шейдер. Фрагментный шейдер. Визуальное программирование: Unity Shader Graph. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.14	Тема 2.4. Написание шейдеров в Unity. Отражения. Альфа-канал. Прозрачность. Вершинный шейдер. Фрагментный шейдер. Визуальное программирование: Unity Shader Graph. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.15	Тема 2.4. Написание шейдеров в Unity. Отражения. Альфа-канал. Прозрачность. Вершинный шейдер. Фрагментный шейдер. Визуальное программирование: Unity Shader Graph. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.16	Тема 2.4. Написание шейдеров в Unity. Отражения. Альфа-канал. Прозрачность. Вершинный шейдер. Фрагментный шейдер. Визуальное программирование: Unity Shader Graph. /Ср/	1	10	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.17	Тема 2.5. Графические библиотеки OpenGL и DirectX. Обзор графической библиотеки OpenGL. Пример программы OpenGL. Обзор графической библиотеки DirectX. Примеры трехмерных построений объектов. Визуальные эффекты. /Лек/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.18	Тема 2.5. Графические библиотеки OpenGL и DirectX. Обзор графической библиотеки OpenGL. Пример программы OpenGL. Обзор графической библиотеки DirectX. Примеры трехмерных построений объектов. Визуальные эффекты. /Лаб/	1	2	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.19	Тема 2.5. Графические библиотеки OpenGL и DirectX. Обзор графической библиотеки OpenGL. Пример программы OpenGL. Обзор графической библиотеки DirectX. Примеры трехмерных построений объектов. Визуальные эффекты. /Пр/	1	1	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.20	Тема 2.5. Графические библиотеки OpenGL и DirectX. Обзор графической библиотеки OpenGL. Пример программы OpenGL. Обзор графической библиотеки DirectX. Примеры трехмерных построений объектов. Визуальные эффекты. /Ср/	1	10	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3
2.21	/Зачёт/	1	0	ПК-3 ПК-4	Л1.1 Л1.2 Л1.3Л2.1 Л2.2 Л2.3

4. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Структура и содержание фонда оценочных средств для проведения текущей и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**5.1. Основная литература**

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л1.1	Боресков А. В.	Графика трехмерной компьютерной игры на основе OpenGL: практическое пособие	Москва: Диалог-МИФИ, 2004	https://biblioclub.ru/index.php?page=book&id=89378 неограниченный доступ для зарегистрированных пользователей
Л1.2	Васильев С. А.	OpenGL: компьютерная графика: учебное пособие	Тамбов: Тамбовский государственный технический университет (ТГТУ), 2012	https://biblioclub.ru/index.php?page=book&id=277936 неограниченный доступ для зарегистрированных пользователей
Л1.3	Лейкова, М. В., Бычкова, И. В.	Инженерная компьютерная графика. Методика решения проекционных задач с применением 3D-моделирования: учебное пособие	Москва: Издательский Дом МИСиС, 2016	http://www.iprbookshop.ru/64175.html неограниченный доступ для зарегистрированных пользователей

5.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Колич-во
Л2.1		Программные продукты и системы: журнал	Тверь: Центрпрограммсистем, 2017	https://biblioclub.ru/index.php?page=book&id=459225 неограниченный доступ для зарегистрированных пользователей
Л2.2	Жуков, Ю. Н.	Инженерная компьютерная графика: учебник	Томск: Томский государственный университет систем управления и радиоэлектроники, 2010	http://www.iprbookshop.ru/14009.html неограниченный доступ для зарегистрированных пользователей
Л2.3	Задорожный А. Г., Персова М. Г., Кошкина Ю. И.	Введение в трехмерную компьютерную графику с использованием библиотеки OpenGL: учебное пособие	Новосибирск: Новосибирский государственный технический университет, 2018	https://biblioclub.ru/index.php?page=book&id=575673 неограниченный доступ для зарегистрированных пользователей

5.3 Профессиональные базы данных и информационные справочные системы

Официальная документация по библиотеке OpenGL. - [Электронный ресурс]. - <https://www.opengl.org/>

ИСС «КонсультантПлюс»

ИСС «Гарант» <http://www.internet.garant.ru/>

5.4. Перечень программного обеспечения

Visual Studio Code

5.5. Учебно-методические материалы для студентов с ограниченными возможностями здоровья

При необходимости по заявлению обучающегося с ограниченными возможностями здоровья учебно-методические материалы предоставляются в формах, адаптированных к ограничениям здоровья и восприятия информации. Для лиц с нарушениями зрения: в форме аудиофайла; в печатной форме увеличенным шрифтом. Для лиц с нарушениями слуха: в форме электронного документа; в печатной форме. Для лиц с нарушениями опорно-двигательного аппарата: в форме электронного документа; в печатной форме.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Помещения для всех видов работ, предусмотренных учебным планом, укомплектованы необходимой специализированной учебной мебелью и техническими средствами обучения:

- столы, стулья;
- персональный компьютер / ноутбук (переносной);
- проектор;
- экран / интерактивная доска
Лабораторные занятия проводятся в компьютерных классах, рабочие места в которых оборудованы необходимыми лицензионными и/или свободно распространяемыми программными средствами и выходом в Интернет.

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Методические указания по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

1. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.1 Критерии оценивания компетенций

ЗУН, составляющие компетенцию	Показатели оценивания	Критерии оценивания	Средства оценивания*
ПК-3: Способен самостоятельно осуществлять руководство процессами разработки программного обеспечения			
З: методы и приемы алгоритмизации поставленных задач, методологии разработки программного обеспечения, основные принципы и методы управления персоналом (соотнесено с индикатором ПК-3.1);	изучает основную и дополнительную литературу, лекционный материал, использует профессиональные базы данных для изучения основ формирования входных массивов в процессе «интеллектуальной» разработки при подготовке к экзамену и опросу, при выборе тематики индивидуального задания	полнота и содержательность ответа на опросе и экзамене, умение приводить примеры, умение отстаивать свою позицию; соответствие ответов материалам лекций и учебной литературы, сведениям из информационных ресурсов Интернет; актуальность выбора исходных данных для моделей индивидуального задания	Э (вопрос 1-16) О (вопрос 1-6) ИЗ (тематика 1-8)
У: использовать методы и приемы формализации задач, использовать выбранную среду программирования, применять коллективную среду разработки программного обеспечения и систему контроля версий, применять методы принятия управленческих решений (соотнесено с индикатором ПК-3.2)	применяет методы управления разработкой программного обеспечения в условиях интеллектуализации при решении лабораторных, практико-ориентированных и индивидуальных заданий	правильность применения методов модернизации программного обеспечения в условиях интеллектуализации в лабораторных и практико-ориентированных заданиях; обоснованность применения выбранного метода в индивидуальном задании	ЛЗ (задание 1-4) ПОЗЭ (вопросы 1-5) ИЗ (тематика 1-8)
В: способами оценки результатов проверки работоспособности программного обеспечения,	применяет методы управления разработкой программного обеспечения в условиях интеллектуализации для решения	объем и индивидуальность выполнения задания с использованием современного инструментария и применения методов рефакторинга; корректность интерпретации	ПЗ (задание 1-2) ПОЗЭ (вопросы 1-5) ИЗ

управленческими решениями по результатам проверки работоспособности выпусков программного продукта (соотнесено с индикатором ПК-3.3)	практических, практико-ориентированных и индивидуальных заданий с использованием современного инструментария	полученных результатов в задании	(тематика 1-8)
ПК-4: Способен осуществлять управление программно-техническими и технологическими ресурсами			
З: методологию управления проектами разработки программного обеспечения, лучшие практики управления разработкой программного обеспечения, принципы и методы управления персоналом (соотнесено с индикатором ПК-4.1)	изучает основную и дополнительную литературу, лекционный материал, использует профессиональные базы данных для изучения основ разработки компонентов систем управления базами данных в условиях интеллектуализации при подготовке к экзамену и опросу, при выборе тематики индивидуального задания	полнота и содержательность ответа на опросе и экзамене, умение приводить примеры, умение отстаивать свою позицию; соответствие ответов материалам лекций и учебной литературы, сведениям из информационных ресурсов Интернет; актуальность выбора исходных данных для моделей индивидуального задания	Э (вопрос 1-16) О (вопрос 1-6) ИЗ (тематика 1-8)
У: применять методы и средства оценки сложности, трудоемкости и сроков выполнения работ, применять методы и средства организации проектных данных, применять основные принципы и методы управления персоналом (соотнесено с индикатором ПК-4.2)	применяет методы и шаблоны разработки компонентов систем управления базами данных в условиях интеллектуализации для решения лабораторных, практико-ориентированных и индивидуальных заданий	правильность применения методов методы и шаблоны разработки компонентов систем управления базами данных в условиях интеллектуализации в лабораторных и практико-ориентированных заданиях; обоснованность применения выбранного метода в индивидуальном задании	ЛЗ (задание 1-4) ПОЗЭ (вопросы 1-5) ИЗ (задание 1-8) тематика
В: способами мониторинга и оценки по выбранным критериям (показателям) сложности, трудоемкости и	применяет методы разработки компонентов систем управления базами данных в условиях интеллектуализации для решения практических, практико-ориентированных	объем и индивидуальность выполнения задания с использованием современного инструментария и применения методов разработки компонентов систем управления базами данных;	ПЗ (задание 1-2) ПОЗЭ (вопросы 1-5) ИЗ (тематика 1-8)

сроков выполнения работ, методами принятия управленческих решений (соотнесено с индикатором ПК-4.3)	х и индивидуального заданий с использованием современного инструментария	корректность интерпретации полученных результатов в задании	
---	--	---	--

* О – опрос; Э – вопросы к экзамену; ПОЗЭ - практико-ориентированные задания к экзамену; ЛЗ – лабораторные задания; ПЗ – практические задания; ИЗ – индивидуальное задание

1.2. Шкала оценивания

Текущий контроль успеваемости и промежуточная аттестация осуществляется в рамках накопительной балльно-рейтинговой системы в 100-балльной шкале:

84-100 баллов (оценка «отлично»)

67-83 баллов (оценка «хорошо»)

50-66 баллов (оценка «удовлетворительно»)

0-49 баллов (оценка «неудовлетворительно»)

2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Вопросы к зачету

1. Основные понятия компьютерной графики. Понятие топологии объекта.
2. Вершины, ребра, полигоны. Нормали.
3. Структуры данных 3D-моделей.
4. Виды полигонов: треугольник, квад, N-гон (N-Gon).
5. Преобразования координат и объектов. Проекция.
6. Текстуры. UV-развертка модели.
7. Модели освещения: Ламберт, Блинн-Фонг.
8. Модель освещения физически корректный рендеринг (PBR).
9. Языки CG, GLSL и HLSL и используемые ими структуры данных.
10. Шейдеры: использование моделей освещения. Использование карт нормалей.
11. Проходы. Кадровый буфер (Frame Buffer), буфер глубины (Z Buffer).
12. Очередь визуализации (Render Queue).
13. Положение камеры и ориентация нормалей полигонов, их скалярное произведение.
14. Использование шейдерных программ в Unity. Язык ShaderLab.
15. Поверхностные шейдеры (surface shaders).
16. Отражения. Альфа-канал. Прозрачность.
17. Вершинный шейдер (vertex shader).
18. Фрагментный шейдер (fragment shader).
19. Визуальное программирование: Unity Shader Graph.
20. Графические библиотеки OpenGL и DirectX.

Типовые практико-ориентированные задания к экзамену*

1. Написание шейдеров: напишите простейший шейдер для визуализации 3D-объекта.
2. Разработка кроссплатформенных приложений: реализуйте Emission в шейдере.
3. Разработка кроссплатформенных приложений: реализуйте маски для текстур.
4. Разработка кроссплатформенных приложений: напишите шейдер с картой нормалей (Normal Map).

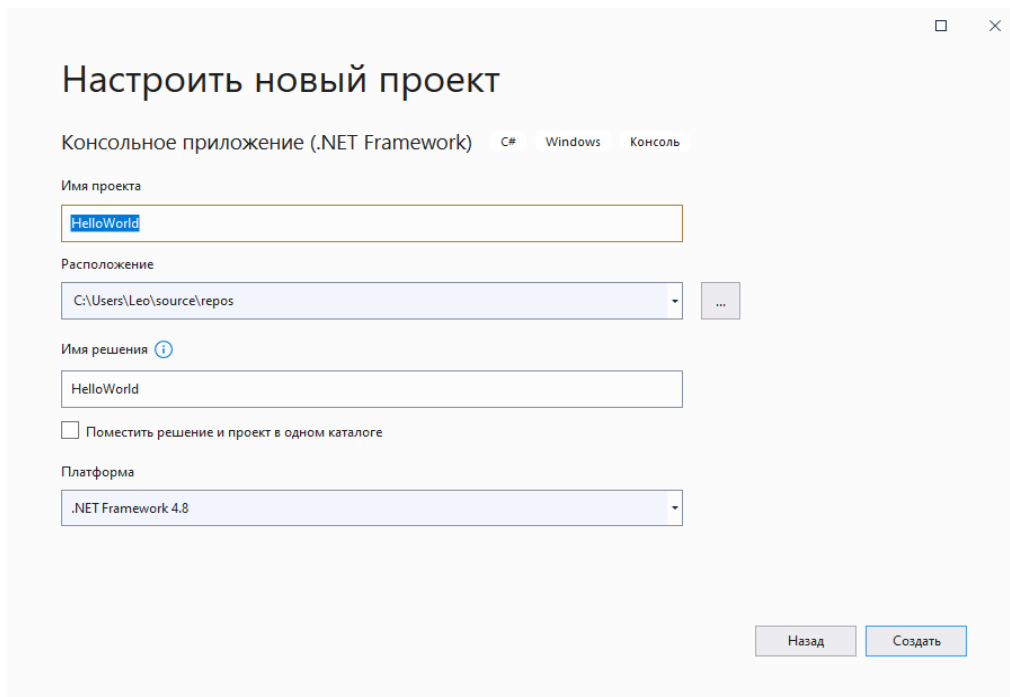
5. Разработка кроссплатформенных приложений: добавьте в шейдер отражения других объектов.

Примечание *: Практические задачи на экзамене также могут выбираться преподавателем случайным образом из перечня лабораторных и практических заданий для текущего контроля изучения дисциплины.

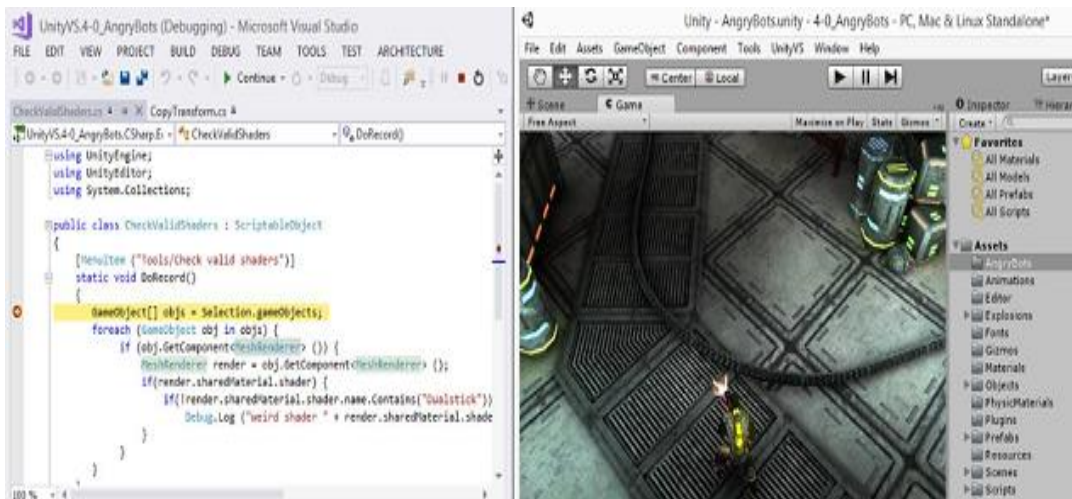
1. Написание шейдеров: напишите простейший шейдер для визуализации 3D-объекта.

```
Shader "Displacement/Displacement_Wave"
{
    Properties
    {
        [PerRendererData]
        _MainTex ("Main Texture", 2D) = "white" {}
        _Color ("Color" , Color) = (1,1,1,1)
    }
    SubShader
    {
        Tags
        {
            "RenderType" = "Transparent"
            "Queue" = "Transparent"
        }
        Cull Off
        Blend SrcAlpha OneMinusSrcAlpha
        Pass
        {
            CGPROGRAM
            #pragma vertex vert
            #pragma fragment frag
            #include "UnityCG.cginc"
            struct appdata
            {
                float4 vertex : POSITION;
                float2 uv : TEXCOORD0;
                float4 color : COLOR;
            };
            struct v2f
            {
                float4 vertex : SV_POSITION;
                float2 uv : TEXCOORD0;
                float4 color : COLOR;
            };
            fixed4 _Color;
            sampler2D _MainTex;
            v2f vert (appdata v)
            {
                v2f o;
                o.uv = v.uv;
                o.color = v.color;
                o.vertex = UnityObjectToClipPos(v.vertex);
                return o;
            }
            fixed4 frag (v2f i) : SV_Target
            {
                fixed4 texColor = tex2D(_MainTex, i.uv)*i.color;
                return texColor;
            }
            ENDCG
        }
    }
}
```

2. Разработка кроссплатформенных приложений: реализуйте Emission в шейдере.



3. Разработка кроссплатформенных приложений: реализуйте маски для текстур.



4. Разработка кроссплатформенных приложений: напишите шейдер с картой нормалей (Normal Map).

```
uniform sampler2D normalMap;
```

```
void main()
```

```
{
    // выборка вектора из карты нормалей с областью значений [0,1]
    normal = texture(normalMap, fs_in.TexCoords).rgb;
    // перевод вектора нормали в интервал [-1,1]
    normal = normalize(normal * 2.0 - 1.0);

    [...]
    // вычисление освещения...
}
```

5. Разработка кроссплатформенных приложений: добавьте в шейдер отражения других объектов.

```
void mainImage( out vec4 fragColor, in vec2 fragCoord )
```

```
{
    vec2 xy = fragCoord.xy / iResolution.xy; // Соединяем это в одну строку
```

```

vec4 texColor = texture(iChannel0,xy); // Получаем от iChannel0 пиксель в координате xy
    texColor.r *= abs(sin(iGlobalTime));
texColor.g *= abs(cos(iGlobalTime));
texColor.b *= abs(sin(iGlobalTime) * cos(iGlobalTime));
fragColor = texColor; // Присваиваем экранному пикселю этот цвет
}

```

Критерии оценивания:

- 84-100 (34-40 за ответ на 2 теоретических вопроса, 50-60 за решение практико-ориентированного задания) баллов (оценка «отлично») – изложенный материал фактически верен, наличие глубоких исчерпывающих знаний в объеме пройденной программы дисциплины в соответствии с поставленной программой курса целью обучения; правильные, уверенные действия по применению полученных навыков и умений при решении практико-ориентированного задания, грамотное и логически стройное изложение материала при ответе, усвоение основной и знакомство с дополнительной литературой;

- 67-83 (27-33 за ответ на 2 теоретических вопроса, 40-50 за решение практико-ориентированного задания) баллов (оценка «хорошо») – наличие твердых и достаточно полных знаний в объеме пройденной программы дисциплины в соответствии с целью обучения, правильные действия по применению навыком и умений при решении практико-ориентированного задания, четкое изложение материала, допускаются отдельные логические и стилистические погрешности, обучающийся усвоил основную литературу, рекомендованную в рабочей программе дисциплины;

- 50-66 (20-26 за ответ на 2 теоретических вопроса, 30-40 за решение практико-ориентированного задания) баллов (оценка «удовлетворительно») – наличие твердых знаний в объеме пройденного курса в соответствии с целью обучения, изложение ответов с отдельными ошибками, уверенно исправленными после дополнительных вопросов; правильные в целом действия по применению навыков и умений при решении практико-ориентированного задания;

- 0-49 (0-19 за ответ на 2 теоретических вопроса, 0-30 за решение практико-ориентированного задания) баллов (оценка «неудовлетворительно») – ответы не связаны с вопросами, наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять умения и навыки при решении практико-ориентированного задания, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

Перечень теоретических типовых вопросов для опроса*

1. Какие шейдерные языки вы знаете?
2. Что такое буфер глубины?
3. Что такое векторы нормалей?
4. Опишите механизм очередей визуализации.
5. Что такое UV-развертка?
6. Какие модели освещения вы знаете?

Примечание *: опрос проводится при проверке всех лабораторных и практических заданий для выявления знаний при изучении соответствующих тем дисциплины в рамках текущей аттестации.

Критерии оценки:

- 1 балл выставляется обучающемуся, если изложенный материал фактически верен и логически обоснован.

Максимальное количество баллов: 10 баллов.

Лабораторные задания

Тематика лабораторных заданий

Лабораторное задание № 1 (5 баллов)

Разработка шейдера с использованием модели освещения Lambert.

Лабораторное задание № 2 (5 баллов)

Реализуйте свечение границ 3D-модели при ее визуализации.

Лабораторное задание № 3 (5 баллов)

Напишите шейдер, использующий Alpha-канал в SurfaceOutput.

Лабораторное задание № 4 (5 баллов)

Имея код на языке CG продублируйте эффект шейдера с помощью Unity Shader Graph.

Критерии оценивания:

Баллы по каждому заданию проставлены в скобках.

Неправильное выполнение задания – 0 баллов.

Максимальное количество баллов, которые могут быть получены обучающимся, - 20.

Практические задания

Тематика практических заданий

Практическое задание № 1 (10 баллов)

Напишите шейдер для физически корректного рендеринга, подход Standard Specular.

Практическое задание № 2 (10 баллов)

Напишите шейдер для физически корректного рендеринга, подход Standard Metallic.

Критерии оценивания:

Баллы по каждому заданию проставлены в скобках.

Неправильное выполнение задания – 0 баллов.

Максимальное количество баллов, которые могут быть получены обучающимся, - 20.

Индивидуальное задание

Типовая тематика индивидуального задания

1. Добавьте использование масок для текстур к шаблону поверхностного шейдера, используемого в Default Render Pipeline в игровом движке Unity.
2. Напишите двухсторонний (double sided) шейдер.
3. Напишите шейдер, реализующий эффект голограммы с помощью скалярного произведения нормалей полигонов и вектора ориентации камеры.
4. Реализуйте самоосвещение по входным данным (текстуре) в шейдере, создаваемом шаблону поверхностного шейдера, используемого в Default Render Pipeline в игровом движке Unity.
5. Напишите шейдер, использующий карту нормалей с возможностью числового управлением ее интенсивности.
6. Добавьте смешивание (blending) текстур в шейдер.
7. Отключите в шейдере запись в буфер глубины.
8. Напишите шейдер, изменяющий вид 3D-модели в зависимости от ее мировых координат.

Задачей данного раздела является практическая реализация освоенных принципов программирования нейронных сетей, а также приобретение обучающимися навыков получения исходных данных, разработки, обучения, модификации и симуляции нейронных сетей.

Максимальное количество баллов за проект – 50 баллов.

Тематику исследования можно сформулировать самостоятельно, предварительно согласовав с преподавателем.

Обучающимся может быть выбран любой стек ИТ-технологий, направленный на «интеллектуальное» программирование, теорию адаптивных алгоритмов (самостоятельное решение, программирование с использованием библиотек, анализ производительности, времени обучения, адекватности моделей) и т.д.

Требования к оформлению индивидуального задания приведены в Приложении 2.

Критерии оценки:

- 42-50 балла – разработанное алгоритмическое решение, его функциональные возможности соответствуют требованиям индивидуального задания; текстовое описание составлено в полном объеме; модель адекватна для всех типовых экспериментов; алгоритм работоспособен на всех наборах исходных данных; обучающийся показал свободное владение тематикой проекта, знание используемого инструментария; изложенный материал фактически верен, наличие глубоких исчерпывающих знаний в рамках пройденной программы; правильные, уверенные действия по применению полученных умений и навыков на практике; усвоение основной и дополнительной литературы, работа с профессиональными базами данных;

- 0-41 баллов – разработанное алгоритмическое решение, его функциональные возможности не в полной мере соответствуют требованиям индивидуального задания; текстовое описание составлено не в полном объеме и не достаточно аккуратно; модель адекватна не для всех типовых экспериментов; алгоритм работоспособен не на всех наборах исходных данных; обучающийся показал достаточно слабые знания по тематике проекта; отсутствие материала из основной и дополнительной литературы, отсутствие индивидуальности в представленном материале; наличие заимствований существующих аналогичных проектов, приведенных в профессиональных базах данных.

3 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедуры оценивания включают в себя текущий контроль и промежуточную аттестацию.

Текущий контроль успеваемости проводится с использованием оценочных средств, представленных в п. 2 данного приложения. Результаты текущего контроля доводятся до сведения студентов до промежуточной аттестации.

Промежуточная аттестация проводится в форме экзамена.

Экзамен проводится по расписанию промежуточной аттестации в письменном виде. Количество вопросов в экзаменационном билете – 3. Проверка ответов и объявление результатов производится в день экзамена.

Результаты аттестации заносятся в экзаменационную ведомость и зачетную книжку студента. Студенты, не прошедшие промежуточную аттестацию по графику сессии, должны ликвидировать задолженность в установленном порядке.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Учебным планом предусмотрены следующие виды занятий:

- лекции;
- лабораторные занятия;
- практические занятия.

В ходе лекционных занятий рассматриваются теоретические вопросы с учетом практико-ориентированности изучаемой дисциплины, даются рекомендации для самостоятельной работы и подготовке к лабораторным и практическим занятиям.

В ходе лабораторных и практических занятий углубляются и закрепляются знания студентов по ряду рассмотренных на лекциях вопросов, развиваются навыки программирования, применения интеллектуальных методов и инструментария разработки программных проектов.

При подготовке к лабораторным и практическим занятиям каждый обучающийся должен:

- изучить рекомендованную учебную литературу;
- изучить практические примеры, рассмотренные на лекциях.

В процессе подготовки к лабораторным и практическим занятиям студенты могут воспользоваться консультациями преподавателя.

По согласованию с преподавателем студент может подготовить индивидуальное задание.

Вопросы, не рассмотренные на лекциях, лабораторных и практических занятиях, должны быть изучены студентами в ходе самостоятельной работы. Контроль самостоятельной работы студентов над учебной программой курса осуществляется в ходе занятий методом устного опроса или при выполнении лабораторных и практических заданий с учетом индивидуальности представленного решения. В ходе самостоятельной работы каждый студент обязан прочитать основную и по возможности дополнительную литературу по изучаемой теме, дополнить конспекты лекций недостающим материалом, выписками из рекомендованных первоисточников. Выделить непонятные термины, найти их значение в энциклопедических словарях.

Для подготовки к занятиям, текущему контролю и промежуточной аттестации студенты могут воспользоваться электронно-библиотечными системами. Также обучающиеся могут взять на дом необходимую литературу на абонементе университетской библиотеки или воспользоваться читальными залами.

Методические рекомендации по оформлению индивидуального задания.

Индивидуальное задание выполняется с учетом приобретенных знаний, навыков и умений по данной дисциплине и интереса обучающегося.

Основными этапами выполнения индивидуального задания являются:

1. Постановка задачи моделирования.
2. Описание предметной области.
3. Обоснование актуальности проекта.
4. Разработка входных переменных проекта.
5. Разработка выходных переменных проекта.
6. Программная реализация проекта.
7. Апробация результатов.
8. Составление отчета.

Работа должна содержать теорию относительно выбранной предметной области: понятия, определения, классификации, возможности применения, плюсы и недостатки; анализ количественных данных модели, а также обязательно практика применения выбранного инструментария, включая скрины этапов разработки, содержательную интерпретацию полученных результатов.

Оформление индивидуального задания должно соответствовать требованиям государственных стандартов, в т.ч. и методических рекомендаций вуза (кафедры). Текст работы должен быть набран на белой бумаге формата А4 с одной стороны листа. Размер шрифта: 12, интервал: 1,5. Поля: левое – 30 мм, правое – 10 мм, верхнее – 20 мм, нижнее – 20 мм.