

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Усынин Максим Валерьевич
Должность: Ректор
Дата подписания: 25.12.2020 14:35:58
Уникальный программный ключ:
f498e59e83f65dd7c3ce7bb8a25cbbabb33ebc58

**Частное образовательное учреждение высшего образования
«Международный Институт Дизайна и Сервиса»
(ЧОУВО МИДиС)**

Кафедра математики и информатики

**ФОНД
ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО
КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ
АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ
РАЗРАБОТКА КОМПЬЮТЕРНЫХ ИГР И МУЛЬТИМЕДИЙНЫХ
ПРИЛОЖЕНИЙ**

Направление подготовки: 09.03.03 Прикладная информатика
Направленность (профиль): Разработка компьютерных игр и приложений с
виртуальной и дополненной реальностью
Квалификация выпускника: Бакалавр
Год набора: 2020

Автор-составитель: Чеботарёв С.С.

Челябинск 2024

СОДЕРЖАНИЕ

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы	3
2. Показатели и критерии оценивания компетенций на различных этапах их формирования, описание шкал оценивания	6
3. 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	8
4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы	16

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ С УКАЗАНИЕМ ЭТАПОВ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс изучения дисциплины «Разработка компьютерных игр и мультимедийных приложений» направлен на формирование следующих компетенций:

Код и наименование компетенции выпускника	Код и наименование индикатора достижения компетенций
ПК-1 Способен кодировать на языках программирования (объектно-ориентированных, современных структурных языках, языках современных бизнес-приложений)	<p>ПК-1.1. Разрабатывает код информационных систем и баз данных информационных систем.</p> <p>ПК-1.2. Осуществляет верификацию кода, баз данных и структуры баз данных информационных систем</p> <p>ПК-1.3. Устраняет обнаруженные несоответствия с применением методик тестирования разрабатываемых информационных систем</p>
ПК-3 Способен проектировать и разрабатывать информационные системы в соответствии с требованиями заказчика	<p>ПК-3.1. Выполнять действия разработке прототипов информационных систем, мобильных и Web приложений</p> <p>ПК-3.2. Выполнять действия по проектированию, верификации информационных систем, мобильных и Web приложений в соответствии с требованиями заказчика.</p> <p>ПК 3.3. Владеть инструментами и методами разработки и тестирования баз данных информационных систем</p>
ПК-4 Способен проектировать информационные ресурсы (web, мобильных приложений) составлять формализованные описания решений, поставленных задач, в соответствии с требованиями, принятыми в организации нормативных документов	<p>ПК-4.1. Составлять формализованные описания решений поставленных задач в соответствии с требованиями, принятыми в организации нормативных документов, выполнять действия по проектированию структур баз данных и дизайну программных интерфейсов.</p> <p>ПК-4.2. Использовать существующие типовые решения и шаблоны информационных ресурсов (web, мобильных приложений), применять методы и средства проектирования и дизайна информационных ресурсов, баз данных и программных интерфейсов.</p> <p>ПК-4.3. Применять типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке информационных ресурсов (web, мобильных приложений)</p>
ПК-5 Способен разрабатывать мобильные приложения	<p>ПК-5.1. Осуществлять деятельность по разработке и отладке мобильных приложений.</p> <p>ПК-5.2. Работать со стандартными сервисами платформ и со встроенными устройствами для получения данных, использовать технологии для работы с различными протоколами обмена данными.</p> <p>ПК-5.3. Применять программные средства, технологии и платформы для разработки мобильных приложений, знать основы</p>

	информационной безопасности
--	-----------------------------

№ п/п	Код компетенции	Наименование компетенции	Этапы формирования компетенций
1.	ПК-1	Способен кодировать на языках программирования (объектно-ориентированных, современных структурных языках, языках современных бизнес-приложений)	<p><i>1 Этап – Знать:</i> ПК-1.1. - способы разработки кода информационных систем</p> <p><i>2 Этап – Уметь:</i> ПК-1.2. - осуществлять верификацию кода, баз данных и структуры баз данных информационных систем</p> <p><i>3 Этап – Владеть:</i> ПК-1.3. - навыками устранения обнаруженных несоответствий с применением методик тестирования разрабатываемых информационных систем</p>
2.	ПК-3	Способен проектировать и разрабатывать информационные системы в соответствии с требованиями заказчика	<p><i>1 Этап – Знать:</i> ПК-3.1. - способы разработки прототипов информационных систем, мобильных и Web приложений</p> <p><i>2 Этап – Уметь:</i> ПК-3.2. - выполнять действия по проектированию, верификации информационных систем, мобильных и Web приложений в соответствии с требованиями заказчика</p> <p><i>3 Этап – Владеть:</i> ПК-3.3. - инструментами и методами разработки и тестирования баз данных информационных систем</p>
3.	ПК-4	Способен проектировать информационные ресурсы (web, мобильных приложений) составлять формализованные описания решений, поставленных задач, в соответствии с требованиями, принятыми в организации нормативных документов	<p><i>1 Этап – Знать:</i> ПК-4.1. - формализованные описания решений поставленных задач в соответствии с требованиями, принятыми в организации нормативных документов; - действия по проектированию структур баз данных и дизайну программных интерфейсов</p> <p><i>2 Этап – Уметь:</i> ПК-4.2. - использовать существующие типовые решения и шаблоны информационных ресурсов (web, мобильных приложений); - применять методы и средства проектирования и дизайна информационных ресурсов, баз данных и программных интерфейсов</p> <p><i>3 Этап – Владеть:</i> ПК-4.3. - способами применения типовых решений, библиотек программных модулей, шаблонов, классов объектов, используемых при разработке информационных ресурсов (web,</p>

			мобильных приложений)
4.	ПК-5	Способен разрабатывать мобильные приложения	<i>1 Этап – Знать:</i> ПК-5.1. - технологии разработки и отладки мобильных приложений
	<i>2 Этап – Уметь:</i> ПК-5.2. - работать со стандартными сервисами платформ и со встроенными устройствами для получения данных, использовать технологии для работы с различными протоколами обмена данными;		
	<i>3 Этап – Владеть:</i> ПК-5.3. - программными средствами, технологиями для разработки мобильных приложений; - основами информационной безопасности		

2. ПОКАЗАТЕЛИ И КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

№ п/п	Код компетенции	Наименование компетенции	Критерии оценивания компетенций на различных этапах формирования	Шкала оценивания
1.	ПК-1	Способен кодировать на языках программирования (объектно-ориентированных, современных структурных языках, языках современных бизнес-приложений)	<i>1 Этап – Знать:</i> ПК-1.1. - способы разработки кода информационных систем	<p>Зачет «ЗАЧТЕНО»:</p> <ol style="list-style-type: none"> 1. Усвоение программного материала. 2. Знание сущности основных категорий и понятий. 3. Выполнение самостоятельной работы за семестр. 4. Точность и обоснованность выводов. 5. Точные, полные и логичные ответы на дополнительные вопросы. <p>«НЕ ЗАЧТЕНО»:</p> <ol style="list-style-type: none"> 1. Незнание значительной части программного материала. 2. Невыполнение самостоятельной работы за семестр. 3. Грубые ошибки при выполнении самостоятельной работы. 4. Неумение выделить главное, сделать выводы и обобщения. 5. Неправильные ответы на
			<i>2 Этап – Уметь:</i> ПК-1.2. - осуществлять верификацию кода, баз данных и структуры баз данных информационных систем	
			<i>3 Этап – Владеть:</i> ПК-1.3. - навыками устранения обнаруженных несоответствий с применением методик тестирования разрабатываемых информационных систем	
2	ПК-3	Способен проектировать и разрабатывать информационн	<i>1 Этап – Знать:</i> ПК-3.1. - способы разработки прототипов информационных систем, мобильных и	

		ые системы в соответствии с требованиями заказчика	<p>Web приложений</p> <p><i>2 Этап – Уметь:</i> ПК-3.2. - выполнять действия по проектированию, верификации информационных систем, мобильных и Web приложений в соответствии с требованиями заказчика</p> <p><i>3 Этап – Владеть:</i> ПК-3.3. - инструментами и методами разработки и тестирования баз данных информационных систем</p>	<p>дополнительные вопросы.</p> <p>Экзамен «ОТЛИЧНО»</p> <ol style="list-style-type: none"> 1. Глубокое и прочное усвоение программного материала. 2. Знание пакетов прикладных программ. 3. Знание основных принципов построения пакетов прикладных программ. 4. Знание основных задач прикладных программ. 5. Свободное владение пакетами прикладных программ. 6. Точность и обоснованность выводов. 7. Безошибочное выполнение практического задания. 8. Точные, полные и логичные ответы на дополнительные вопросы.
3	ПК-4	Способен проектировать информационные ресурсы (web, мобильных приложений) составлять формализованные описания решений, поставленных задач, в соответствии с требованиями, принятыми в организации нормативных документов	<p><i>1 Этап – Знать:</i> ПК-4.1. - формализованные описания решений поставленных задач в соответствии с требованиями, принятыми в организации нормативных документов; - действия по проектированию структур баз данных и дизайну программных интерфейсов</p> <p><i>2 Этап – Уметь:</i> ПК-4.2. - использовать существующие типовые решения и шаблоны информационных ресурсов (web, мобильных приложений); - применять методы и средства проектирования и дизайна информационных</p>	<p>«ХОРОШО»</p> <ol style="list-style-type: none"> 1. Хорошее знание программного материала. 2. Недостаточно полное изложение теоретического вопроса экзаменационного билета. 3. Наличие незначительных неточностей в употреблении терминов, классификаций. 4. Знание основных пакетов прикладных программ. 5. Неполнота представленного иллюстративного материала. 6. Точность и обоснованность выводов. 7. Логичное изложение вопроса, соответствие изложения научному стилю. 8. Негрубая ошибка при выполнении практического задания. 9. Правильные ответы на дополнительные вопросы. <p>«УДОВЛЕТВОРИТЕЛЬНО»</p> <ol style="list-style-type: none"> 1. Поверхностное усвоение

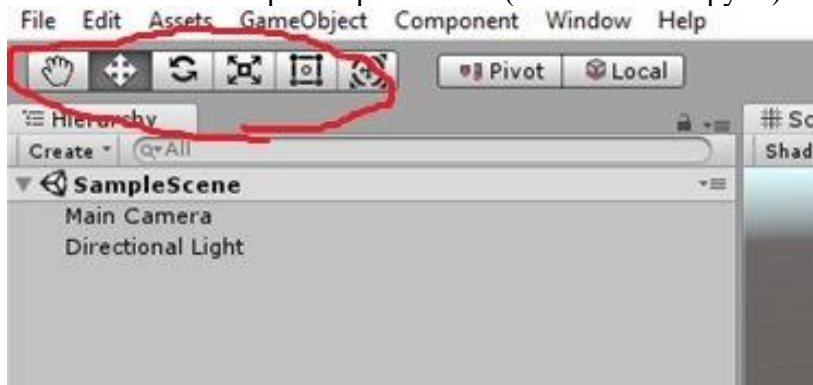
			ресурсов, баз данных и программных интерфейсов	<p>программного материала.</p> <p>2. Недостаточно полное изложение теоретического вопроса экзаменационного билета.</p> <p>3. Затруднение в приведении примеров, подтверждающих теоретические положения.</p> <p>4. Наличие неточностей в употреблении терминов, классификаций.</p> <p>5. Неумение четко сформулировать выводы.</p> <p>6. Отсутствие навыков научного стиля изложения.</p> <p>7. Грубая ошибка в практическом задании.</p> <p>8. Неточные ответы на дополнительные вопросы.</p>
			<p><i>3 Этап – Владеть:</i></p> <p>ПК-4.3. - способами применения типовых решений, библиотек программных модулей, шаблонов, классов объектов, используемых при разработке информационных ресурсов (web, мобильных приложений)</p>	
4	ПК-5	Способен разрабатывать мобильные приложения	<p><i>1 Этап – Знать:</i></p> <p>ПК-5.1. - технологии разработки и отладки мобильных приложений</p> <p><i>2 Этап – Уметь:</i></p> <p>ПК-5.2. - работать со стандартными сервисами платформ и со встроенными устройствами для получения данных, использовать технологии для работы с различными протоколами обмена данными;</p> <p><i>3 Этап – Владеть:</i></p> <p>ПК-5.3. - программными средствами, технологиями для разработки мобильных приложений;</p> <p>- основами информационной безопасности</p>	<p>«НЕУДОВЛЕТВОРИТЕЛЬНО»</p> <p>1. Незнание значительной части программного материала.</p> <p>2. Неспособность привести примеры пакетов прикладных программ</p> <p>3. Неумение выделить главное, сделать выводы и обобщения.</p> <p>4. Грубые ошибки при выполнении практического задания.</p> <p>5. Неправильные ответы на дополнительные вопросы.</p>

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТ ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

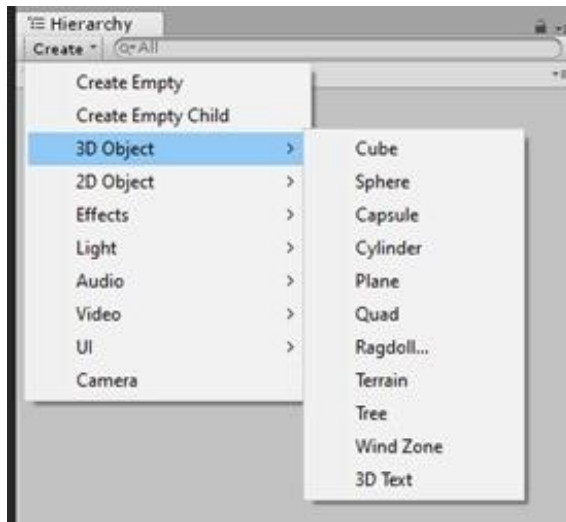
1 ЭТАП – ЗНАТЬ

Тестирование

1. Unity - это...
 название компьютерной игры
 название компании, занимающейся разработкой игр
 среда разработки компьютерных игр
 логотип, которым отмечают качественные игры
2. Что такое Игровой движок?
 Система программ для работы в офисе
 Базовое программное обеспечение компьютерной игры.
 Физика в компьютерных играх
3. Что такое Сцена? (Scene)
 Отдельный уровень игры в Юнити
 Отдельная часть уровня игры в Юнити
4. За что отвечает Иерархия?
 Список объектов на уровне и их права на управление друг другом.
 Все файлы проекта, разделенные по папкам
 Настройки выбранных объектов, их размер, положение в пространстве и т.д
5. За что отвечает Инспектор?
 Список объектов на уровне и их права на управление друг другом.
 Все файлы проекта, разделенные по папкам
 Настройки выбранных объектов, их размер, положение в пространстве и т.д
6. За что отвечает список файлов внизу?
 Список объектов на уровне и их права на управление друг другом.
 Все файлы проекта, разделенные по папкам
 Настройки выбранных объектов, их размер, положение в пространстве и т.д
7. Зажав правую кнопку мыши в окне сцены можно летать на WASD?
 Нет
 Да
8. За что отвечают первые три кнопки (после кнопки руки) на картинке?



- Перетащить объект, повернуть объект и поменять размер объекта
 Удалить объект, сохранить объект, копировать объект
9. Что за меню?



Список 3D объектов для игры, куб, сфера, капсула и так далее (правая кнопка мыши в Иерархии)

Список 2D объектов для игры, куб, сфера, капсула и так далее (левая кнопка мыши в Иерархии)

10. За что отвечает материал?

За внешний вид, цвет, текстуру

За внутренние свойства

11. Что такое Ассеты?

это элементы кода программ игрового движка

это компоненты игры, которые прикрепляются к объектам. (Графика, звуковое сопровождение, скрипты)

12. Что такое Префаб?

это сохраненная коллекция, заготовка, содержащая один или несколько объектов.

Это особый тип текстуры, или материала

13. За что отвечает коллайдер?

За его внешний вид

За физические границы объекта

14. За что отвечает Rigidbody

За физическое поведение объекта

За аудио объекта

15. Что такое Террейн?

Земля, форму которой можно менять

Физические границы объекта

16. Directional Light это...

Точечный источник

Глобальное освещение сцены

17. Что такое скрипт?

Это программа, инструкция поведения игровых объектов

Это тип игрового материала

18. За что отвечает AudioSource?

За отслеживание цвета

За проигрывание звука

19. Можно ли работать в бесплатной версии Unity?

Можно, но лишь до того, как ваш годовой доход не превысил 100 000 \$ за год

Да, можно без ограничений

Нет, всегда необходимо покупать платную версию

20. Для чего нужна вкладка Navigator?

она позволяет указать объекты, по которым можно ходить

она служит для создания игровых объектов на сцене
 вкладка позволяет создать агента и поверхность по которой сможет передвигаться агент
 вкладка позволяет создать агентов
 в ней создается искусственный интеллект для игры

21. Как называется официальный магазин от Unity?

Unity Store
 Packages Store
 Asset Store
 Projects Store
 Plugins Store

22. Можно ли вращать и передвигать камеру?

Можно всегда
 Нет, нельзя
 Можно, только если камера не основная

23. За что отвечает вкладка Lighting?

она позволяет указать лишь skybox для основной камеры
 она используется для настройки лишь света в игре
 вкладка отвечает за работу со светом на сцене, а также позволяет установить туман
 в ней можно настроить свойства Direction Light

24. Какие объекты добавляются по умолчанию?

имеется ввиду какие объекты добавлены сразу же на сцену при создании пустого 3D проекта
 объекты Main Camera и Directional light
 объекты по умолчанию не добавляются
 объекты Main Camera, Directional light и один пустой объект
 лишь объект Directional light
 лишь объект Main Camera

25. Какая вкладка отвечает за отображение сообщений?

вкладка Inspector
 вкладка Project
 вкладке Console
 вкладке Hierarchy

26. Какой компонент есть у каждого объекта?

Collider
 Rigidbody
 Mesh Renderer
 Light
 Transform

27. Какое из полей ниже будет доступно в Инспекторе?

int speed = 100;
 private int speed = 100;
 protected int speed = 100;
 static int speed = 100;
 public int speed = 100;

28. В чем отличия между Update и FixedUpdate?

Никаких отличий нет
 FixedUpdate имеет фиксированное время вызова, функция Update вызывается без фиксированного времени
 FixedUpdate работает лишь с физическими объектами, Update со всеми
 FixedUpdate вызывается лишь несколько раз за всю игру, Update постоянно

29. Можно ли создавать собственные функции в классе?

Да, можно создавать любые собственные функции

Нет, нельзя

Да, можно, но только с модификатором доступа public

Да, можно, но только с модификатором доступа private

30. В какой строке правильно присвоен компонент переменной?

rb = GetComponent <Rigidbody>;

rb = GetComponent <Rigidbody> ();

rb = GetComponent (Rigidbody) {};

rb = GetComponent (Rigidbody) <>;

rb = GetComponent (Rigidbody);

31. Какой метод Unity позволяет создавать объекты во время игры?

Функция - Instantiate();

Функция - Update();

Функция - Inst();

Функция - Coroutine();

Функция - Create();

32. Как можно уничтожить объект в Unity?

Delete(gameObject);

Clear(gameObject);

Empty(gameObject);

Destroy(gameObject);

33. В какой строке правильно выбирается случайное число из диапазона?

Random.Range (0 - 20);

Range (0, 20);

Random (0, 20);

Random.Range (0..20);

Random.Range (0, 20);

34. Какую директиву нужно дописать к переменной, чтобы она появилась в Unity?

Переменная:

private float num = 23;

[Show]

[SerializeField]

[VisibleInInspector]

[ShowInInspector]

35. Какие методы работают за время полного клика мышкой на объект?

* На блоке есть Collider, который позволяет отслеживать нажатия

Сработают все методы

OnMouseDown

Только OnMouseDown и OnMouseUp

OnMouseOver

OnMouseUp

36. Где создано поле, в которое можно указать компонент анимации?

public Animation some;

public Anim some;

public Play some;

public AnimationListener some;

37. Какой класс считается основным для наследования в Unity?

Класс UnityEngine

Класс Unity

Класс MonoBehaviour

Класс System

Класс Engine

38. Какая из функций ниже будет вызвана первой?

Функция Awake

Функция Start

Функция OnLoad

Функция FixedUpdate

Функция Update

39. Какой метод вызывается при удалении класса (объекта)?

Finish();

OnUnload();

Destroy();

OnFinish();

OnDestroy();

40. Какой класс отвечает за обработку нажатий?

имеется ввиду класс, через который мы можем отследить нажал ли пользователь на какую-то клавишу или что-то в этом роде

Класс GUI

Класс UI

Класс Screen

Класс Get

Класс Input

41. Какой символ служит для обращения к методу объекта?

Пример объекта: Collider col;

col->some();

col+some();

col.some();

col-some();

42. Какие существуют основные окна в редакторе Unity?

Scene

Game

Hierarchy

Project

Object

2 ЭТАП – УМЕТЬ

Комплект задач

- Что такое lerp? Напишите lerp для float.
- Что такое dot-product, cross-product векторов? Геометрический смысл этих операций.
- Расскажите про графический пайплайн, какие системы координат используются на этапах, какие преобразования осуществляются? Что такое drawcall?
- Какие способы представления ориентации объекта в пространстве вы знаете? Чем хороши кватернионы?
- Почему slerp работает по кратчайшей дуге?
- Какие модели освещения вы знаете? Расскажите о принципе работы какой-нибудь из них.
- Когда GPU делает sample текстуры каким образом выбирается mipmap-уровень?
- Какие данные в себе хранит меш?
- Для чего нужны нормали? Как нормали преобразуются из одной системы координат в другую (например, из локальных координат в мировые, координаты камеры)? Почему таким образом?

- Есть два вектора — a , b (не ортогональные). Необходимо получить вектор ортогональный плоскости образованной векторами a и b .
- Есть два вектора — a , b (не ортогональные). Необходимо получить вектор ортогональный a или b в плоскости образованной этими векторами.
- Каким образом по вектору нормали и вектору света определить вектор отраженного света?
- Напишите матрицу поворота вокруг любой из ос.

Кейс-задачи

Кейс-задача 1.

Задание:

Проектирование и разработка сцен:

создать сцену, в которой при нажатии на пробел в стенку кидается ящик, и после удара стенка исчезает.

Кейс-задача 2

Задание:

Проектирование и разработка сцен:

нужно создать сцену, в которой по нажатию пробела в поле зрения игрока начнёт падать ящик, в итоге запуская другой ящик в сторону мишени. После попадания мишень должна исчезнуть с каким-нибудь звуком. Запуск ящика – на основе рычага: один ящик уже лежит с одной стороны, ещё один падает на другую сторону рычага.

Кейс-задача 3.

Задание:

Проектирование и разработка сцен:

игрок должен уметь перемещаться вправо/влево и стрелять по трём мишеням. Мишени при попадании должны падать, а игроку за каждую сбитую мишень начисляется очко. По достижении трёх очков показывается экран окончания игры.

3 ЭТАП – ВЛАДЕТЬ

Темы групповых творческих проектов

- 1) Разработка игрового- приложения “Гонки”
- 2) Разработка игрового приложения “Тетрис”
- 3) Разработка игрового -приложения “Лабиринтов”
- 4) Разработка игрового приложения “Карты”
- 5) Разработка игрового -приложения “Игровой автомат”
- 6) Разработка игрового -приложения “Поиск Клада”

Примерный перечень тем курсовых работ

Проектирование и разработка компьютерной игры либо мультимедийного приложения с использованием Unity3D.

1. Проектирование и разработка игры Марио.
2. Проектирование и разработка игры Марио Ран.
3. Проектирование и разработка игры гонка в стиле Драг.
4. Проектирование и разработка игры 2048.
5. Проектирование и разработка игры симулятор студента.
6. Проектирование и разработка игры симулятор компании по разработке игр.
7. Проектирование и разработка тренажер проведения химических опытов.
8. Проектирование и разработка виртуальная прогулка по институту.
9. Проектирование и разработка виртуальной лаборатории по сбору электрической цепей.
10. Проектирование и разработка 2Д платформера.

11. Проектирование и разработка игры бильярд.

Вопросы к зачету

1. История появления и развития Unity.
2. Основные элементы интерфейса Unity3D.
3. Создание объектов в Unity3D.
4. Типы данных. Переменные.
5. Основные управляющие конструкции.
6. Основные методы управления ходом выполнения программы.
7. Переменные, имена и типы переменных, объявление и использование переменных, оператор присваивания, режим обязательного объявления типа переменных.
8. Работа с различными типами переменных.
9. Выражения и функции Visual C#.
10. Функции обмена данными.
11. Ветвления.
12. Повторения и циклы.
13. Виды циклов.
14. Вложенные циклы.
15. Создание различных поверхностей.
16. Настройка шейдеров.
17. Создание отражений. RenderToTexture.
18. Использование нескольких камер в сцене.
19. Работа с UI компонентами, создание своего интерфейса.
20. Изучение основных компонентов для реализации базовых алгоритмов игровых механик.
21. Реализация UI интерфейсов в игровом движке Unity, способы верстки игровых интерфейсов. Особенности интерфейсов в 2D и 3D играх.
22. Типы переменных, функций, условий и базовые классы Unity3D.
23. Работа с физикой в Unity.
24. Создание анимации, механизмы переключения анимаций.
25. Работа с анимациями.
26. Виды анимации и способы управления ними.
27. Работа с анимациями через скрипты.
28. Система Mecanim.
29. Создание анимационных контроллеров для персонажей.
30. Компонент Animator.
31. Типы скелетных анимаций.
32. Настройка переходов между анимациями. State Machine.
33. Формирование навыка определения наиболее подходящего паттерна в реализации игровой логики.
34. Создание игровых механик в паттерне MVC, деление игры на модели, представления и контроллеры.
35. Создание игровых механик в паттерне MVVM, деление игры на составные части паттерна.
36. Что такое API. Способы интеграции. Способы работы с API.

37. Что такое MonoBehaviour. Основополагающие компоненты MonoBehaviour. Подключаемые библиотеки MonoBehaviour.

Вопросы к экзамену

1. Параметры для управления поведением персонажа.
2. Программное управление состояниями.
3. Особенности переноса анимаций.
4. Аватар. Понятие маски аватара персонажа.
5. Настройка импортируемой модели в Unity3D.
6. Работа с ригом.
7. Одежда. Создание «мягкой» одежды для анимированного персонажа.
8. Взаимодействие с твердыми телами.
9. Обзор систем частиц.
10. Работа с материалами для частиц.
11. Подбор шейдера.
12. Системы частиц Shuriken и Legacy.
13. Отличия, особенности, специфика использования, назначение.
14. Физика частиц.
15. Сочетание нескольких источников частиц.
16. Примеры различных эффектов: магия, взрывы, «бафы» и т.д.
17. Назначение каждого из источников.
18. Свет, как элемент оформления уровня. Lightmapping в Unity3D.
19. Особенности. Виды карт света.
20. Знакомство с префабами, конфигурация, инициализация, удаление префабов.
21. Программирование систем управления сценой в стадии редактирования.
22. Используемые языки. Основные блоки.
23. Синтаксис. Создание входящих параметров для инспектора свойств.
24. Unity3D как среда для разработки игр.
25. Особенности базовых возможностей.
26. Работа с ресурсами для 3D-игр.
27. Поддержка устройств с разными разрешениями экрана.
28. Настройка экспорта финального билда.

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

1 ЭТАП – ЗНАТЬ

Критерии оценивания результатов теста

Полная версия тестовых вопросов содержится в электронно-информационной системе вуза. Студенты проходят тестирование в компьютерном классе. Оценка успешности прохождения теста определяется следующей сеткой: от 0% до 29% – «неудовлетворительно», от 30% до 59% – «удовлетворительно»; 60% – 79 % – «хорошо»; 80% -100% – «отлично».

2 ЭТАП – УМЕТЬ

Критерии оценивания результатов решения задач

Оценка	Критерии
«отлично»	Алгоритм составлен верно; программный код написан грамотно; использован оптимальный метод решения; программа выдаёт верный результат при разных входных данных
«хорошо»	Алгоритм составлен верно; программный код написан грубо, без соблюдения правил и рекомендаций; использован неоптимальный метод решения; программа иногда выдаёт неверный результат при некоторых граничных входных данных
«удовлетворительно»	Алгоритм составлен, но содержит ошибки но программа написана и иногда выдаёт верный результат
«неудовлетворительно»	Алгоритм составлен неверно;

Критерии оценивания решения кейс-задач

Оценка	Критерии
«отлично»	решение задачи осуществляется командой; разработана архитектура классов; разработано приложение; на защите кейс-задачи были получены верные ответы на все дополнительные вопросы
«хорошо»	решение задачи осуществляется командой; разработана архитектура классов; разработано приложение; на защите кейс-задачи при ответах на вопросы были допущены ошибки
«удовлетворительно»	разработана архитектура классов; разработано приложение, но в приложении имеются ошибки и недоработки; на защите кейс-задачи при ответах на вопросы были допущены ошибки
«неудовлетворительно»	разработана архитектура классов; не было разработано приложение

3 ЭТАП – ВЛАДЕТЬ

Критерии оценивания групповых творческих проектов

Оценка	Критерии
«отлично»	разработана архитектура классов; разработано приложение; разработанное приложение полностью соответствует техническому заданию и реализует требуемый функционал
«хорошо»	разработана архитектура классов; разработано приложение; разработанное приложение соответствует техническому заданию не в полном объёме имеются недоработки и ошибки
«удовлетворительно»	разработана архитектура классов; разработано приложение; разработанное приложение не соответствует техническому заданию работа не выполнена в полном объёме

«неудовлетворительно»	работа не выполнена
-----------------------	---------------------

Критерии оценивания курсовой работы

Оценка	Критерии
«отлично»	разработана архитектура классов; разработано приложение; разработанное приложение полностью соответствует техническому заданию и реализует требуемый функционал дано всестороннее и глубокое освещение избранной темы в тесной взаимосвязи с практикой, а её автор показал умение работать с различными видами источников, систематизировать, классифицировать, обобщать материал, формулируя выводы, соответствующие поставленным целям;
«хорошо»	разработана архитектура классов; разработано приложение; разработанное приложение соответствует техническому заданию не в полном объёме имеются недоработки и ошибки работа, отвечающая основным, предъявляемым к ней требованиям. Студент обнаруживает глубокие знания по предмету и владеет навыками научного исследования, но при этом имеются незначительные замечания по содержанию работы, по процедуре защиты (студент не может дать аргументировано ответы на вопросы);
«удовлетворительно»	разработана архитектура классов; разработано приложение; разработанное приложение не соответствует техническому заданию работа не выполнена в полном объёме в основном, соблюдены общие требования, но неполно раскрыты разделы плана, работа носит реферативный характер, отсутствуют аргументированные выводы. Автор курсовой работы посредственно владеет материалом, поверхностно отвечает на вопросы, в процессе защиты курсовой работы;
«неудовлетворительно»	установлен акт несамостоятельного выполнения работы, имеются принципиальные замечания по многим параметрам, содержание не соответствует теме, допущены грубые теоретические ошибки.

Критерии оценивания знаний на зачете

«ЗАЧТЕНО»:

1. Усвоение программного материала.
2. Знание сущности основных категорий и понятий.
3. Выполнение самостоятельной работы за семестр.
4. Точность и обоснованность выводов.
5. Точные, полные и логичные ответы на дополнительные вопросы.

«НЕ ЗАЧТЕНО»:

1. Незнание значительной части программного материала
2. Невыполнение самостоятельной работы за семестр.
3. Грубые ошибки при выполнении самостоятельной работы.

4. Неумение выделить главное, сделать выводы и обобщения.

5. Неправильные ответы на дополнительные вопросы.

Критерии оценивания знаний на экзамене

Оценка	Критерии
«отлично»	<ol style="list-style-type: none"> 1. Глубокое и прочное усвоение программного материала. 2. Знание пакетов прикладных программ. 3. Знание основных принципов построения пакетов прикладных программ. 4. Знание основных задач прикладных программ. 5. Свободное владение пакетами прикладных программ. 6. Точность и обоснованность выводов. 7. Безошибочное выполнение практического задания. 8. Точные, полные и логичные ответы на дополнительные вопросы.
«хорошо»	<ol style="list-style-type: none"> 1. Хорошее знание программного материала. 2. Недостаточно полное изложение теоретического вопроса экзаменационного билета. 3. Наличие незначительных неточностей в употреблении терминов, классификаций. 4. Знание основных пакетов прикладных программ. 5. Неполнота представленного иллюстративного материала. 6. Точность и обоснованность выводов. 7. Логичное изложение вопроса, соответствие изложения научному стилю. 8. Негрубая ошибка при выполнении практического задания. 9. Правильные ответы на дополнительные вопросы.
«удовлетворительно»	<ol style="list-style-type: none"> 1. Поверхностное усвоение программного материала. 2. Недостаточно полное изложение теоретического вопроса экзаменационного билета. 3. Затруднение в приведении примеров, подтверждающих теоретические положения. 4. Наличие неточностей в употреблении терминов, классификаций. 5. Неумение четко сформулировать выводы. 6. Отсутствие навыков научного стиля изложения. 7. Грубая ошибка в практическом задании. 8. Неточные ответы на дополнительные вопросы.
«неудовлетворительно»	<ol style="list-style-type: none"> 1. Незнание значительной части программного материала. 2. Неспособность привести примеры пакетов прикладных программ 3. Неумение выделить главное, сделать выводы и обобщения. 4. Грубые ошибки при выполнении практического задания. 5. Неправильные ответы на дополнительные вопросы.