

Департамент информатизации Тюменской области
Государственное автономное учреждение дополнительного образования
Тюменской области
«Региональный информационно-образовательный центр»

СОГЛАСОВАНО

Заместитель Губернатора
Тюменской области,
директор Департамента
информатизации
Тюменской области



С.И. Логинов
«10» августа 2026 г.

УТВЕРЖДАЮ

Заместитель директора
ГАУ ДО ТО «РИО-Центр»



О.А. Кононенко
«10» августа 2026 г.

УЧЕБНАЯ ПРОГРАММА
«Разработка игр на движке Unreal Engine»

Трудоемкость программы – 112 академических часов

Форма обучения – очная, очно-заочная с применением дистанционных образовательных технологий.

Режим занятий – 2 академических часа в день

Начальные навыки: базовые навыки работы на персональном компьютере

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цель обучения: сформировать у слушателей знания, умения и навыки

- работы в движке Unreal Engine, необходимые для разработки игр для ПК и мобильных устройств;
- работы с использованием графического редактора кода Blueprint и языка программирования C++;
- объектно-ориентированного программирования, необходимые для дальнейшего изучения объектно-ориентированных языков программирования и современных технологий по разработке, распространению и поддержке программного обеспечения.

Данная программа предназначена для обучающихся 13-18 лет.

Компетенции на выходе:

Знания:

- синтаксис языка C++;
- устройство движка Unreal Engine;
- основы нодовой системы визуального написания кода Blueprint;
- основные знания основ анимации;
- основные знания для создания UX/UI элементов и VFX эффектов;
- импорт и сборка ассетов, локаций и шаблонных проектов в игровых движках.

Умения:

- использовать операторы языка C++;
- писать скрипты;
- пользоваться библиотечными функциями;
- использовать язык C++ для разработки игр и приложений;
- использовать движок для проектирования интерфейса приложений;
- писать функции;
- использовать различные 2D и 3D ассеты (элементы окружения и инвентаря («пропсы»), персонажей, и т.п.);
- создавать риг, «скиннинг» (оснастку костями) и базовую анимацию;
- создавать UX/UI элементы и VFX эффекты для игр;
- подготавливать и экспортировать ассеты для дальнейшего использования в игровых движках;
- собирать локации в игровом движке на готовых шаблонах.

Приёмы работы:

- с основными инструментами движка Unreal Engine;
- с языком программирования C++;
- со вспомогательными программами.

Занятия могут вестись как в очной форме, так и с применением дистанционных образовательных технологий (ДОТ). При очном обучении слушатели посещают аудиторные занятия, прослушивают лекции и выполняют практические задания по темам совместно с тренером и самостоятельно в рамках аудиторной работы. При

обучении с применением дистанционных технологий учебный процесс проходит в формате веб-занятия (вебинары, дистанционные уроки). В формате вебинаров прослушивается лекционный материал, практические задания выполняются параллельно с тренером (при наличии технической возможности), разбираются вопросы в режиме реального времени. Также предусматривается самостоятельная внеаудиторная работа (очно-заочная форма с применением ДОТ).

УЧЕБНЫЙ ПЛАН

№ п/п	Наименование разделов	Всего (ак.ч.)	в том числе:		Формы контроля
			аудиторные занятия	самостоятельная работа	
1	Введение	2	2		Выполнение практических заданий
2	Основы работы с движком Unreal Engine	4	4		
3	Программирование игровой логики с использованием Blueprint	8	8		
4	Знакомство с основными системами Unreal Engine	26	26		
5	Разработка полноценной игры с использованием Blueprint	28	28		
6	Программирование игровой логики с использованием C++	10	10		
7	Проектная деятельность	32	32		Выполнение теста
8	Итоговое тестирование	2	2		
Итого		112	112		

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ п/п	Наименование тем	Всего часов (ак.ч.)	в том числе:	
			Теория	Прак- тика
Раздел 1. Введение				
1.1	Знакомство. Техника безопасности	0,5	0,5	0
1.2	Введение в GameDev индустрию	1	1	0
1.3	Установка необходимого софта	0,5	0,5	0
Итого по разделу		2	2	0
Раздел 2. Основы работы с движком Unreal Engine				
2.1	Знакомство с интерфейсом движка	1	1	0
2.2	Инструменты для манипулирования объектами в пространстве	1	0,5	0,5
2.3	Компоненты движка. Иерархия. Создание первого Actor	1	0,5	0,5

2.4	Материалы. Карты для улучшения визуальной составляющей материала. Создание своего материала	1	0,5	0,5
Итого по разделу		4	2,5	1,5
Раздел 3. Программирование игровой логики с использованием Blueprint				
3.1	Создание игровой логики. Blueprint. Объект	1	1	0
3.2	Переменные. Типы данных. Основные операции	1	1	0
3.3	Условные операторы. Циклы. Функции и макросы	1	1	0
3.4	Разработка игры в жанре Runner	5	2	3
Итого по разделу		8	5	3
Раздел 4. Знакомство с основными системами Unreal Engine				
4.1	Класс Character. Input система	2	1,5	0,5
4.2	Skeletal Mesh. Скелеты и кости моделей	1	1	0
4.3	Анимация в Unreal Engine. "Оживляем" персонажа. Добавление анимаций	3	2	1
4.4	Работа со звуком в движке	2	1	1
4.5	UX/UI. Работа с пользовательским интерфейсом	4	3	1
4.6	Landscape. Foliage. Water System	4	3	1
4.7	AI. NavMesh. Поиск пути на местности	2	1	1
4.8	VFX. Создание эффектов с помощью Niagara	4	2	2
4.9	MetaHuman. Создание реалистичных персонажей	4	2	2
Итого по разделу		26	16,5	9,5
Раздел 5. Разработка полноценной игры с использованием Blueprint				
5.1	Создание ландшафта игровой локации	2	0	2
5.2	Создание игрока. Перемещение, звуки, анимация	2	1	1
5.3	Разработка системы характеристик игрока	4	0	4
5.4	Инвентарь. Реализация системы хранения предметов	4	2	2
5.5	Система взаимодействия. Подбор ресурсов, взаимодействие с предметами	4	1	3
5.6	Крафтинг. Создание объектов с использованием ресурсов	3	0	3
5.7	Система строительства. Строительство зданий	3	1	2
5.8	ИИ животных. Добавление враждебных мобов	2	1	1
5.9	PCG Spawner. Процедурная генерация объектов	2	1	1
5.10	Система смены дня и ночи. Генерация погодных условий	2	1	1
Итого по разделу		28	8	20
Раздел 6. Программирование игровой логики с использованием C++				
6.1	Изучение основ C++. Сравнение с Blueprint	4	2	2
6.2	Реализация мини-игры с использованием C++	6	3	3
Итого по разделу		10	5	5
Раздел 7. Проектная деятельность				

7.1	Разработка выпускной работы	1	0	1
7.2	Разработка концепта выпускного проекта	30	0	30
7.3	Полировка игры. Подготовка к сборке. Финальный билд	1	0	1
Итого по разделу		32	0	32
Раздел 8. Итоговое тестирование				
8.1	Подготовка к защите работы	1	1	0
8.2	Защита работы, итоговое тестирование	1	0	1
Итого по разделу		2	1	1
ВСЕГО		112	40	72

СОДЕРЖАНИЕ ПРОГРАММЫ

1. Введение

1.1. Знакомство. Техника безопасности

Знакомство с группой. Обсуждение формата занятий. Сбор ожиданий. Обсуждение техники безопасности и правил на занятиях.

1.2. Введение в GameDev индустрию

Что такое GameDev? Какие есть специальности и направления в индустрии разработки видеоигр. Краткая история видеоигровой индустрии.

1.3. Установка необходимого софта

Краткий обзор необходимого программного обеспечения. Что такое движок? Что такое IDE? Разбор процесса установки программного обеспечения на ПК.

2. Знакомство с движком Unreal Engine

2.1. Знакомство с интерфейсом движка

Основы интерфейса движка Unreal Engine. Окна Viewport, Content Browser, Details, Outliner, Place Actor.

2.2. Инструменты для манипулирования объектами в пространстве

Обзор инструментов для манипулирования объектами в пространстве. Что такое Move Tool, Rotate Tool, Scale Tool. Постройка локации с использованием примитивов.

2.3. Компоненты движка. Иерархия. Создание первого Actor

Что такое компонент? Иерархия компонентов. Создание первого Actor из примитивов.

2.4. Материалы. Карты для улучшения визуальной составляющей материала. Создание своего материала

Что такое материалы? Создание первого материала. Что такое карты текстур и какие они бывают? Использование карт Base Color, Normal, Metallic, Roughness, Emissive Color.

3. Программирование игровой логики с использованием Blueprint

3.1. Создание игровой логики. Blueprint. Объект

Знакомство с Blueprint. Как программируется игровая логика? Понятие объекта в игровых движках.

3.2. Переменные. Типы данных. Основные операции

Хранение данных в программе. Что такое переменная? Типы данных в программировании. Основные операции над данными.

3.3. Условные операторы. Циклы. Функции и макросы

Изучение основных конструкций для написания игровой логики. Что такое условие и условный оператор? Циклы for и while. Что такое функция и макрос?

3.4. Разработка игры в жанре Runner

Применение полученных знаний на примере разработки игры «Runner». Спавн объектов. Обработка столкновений.

4. Знакомство с основными инструментами Unreal Engine

4.1. Класс Character. Input система

Знакомство с классом Character. Создание шаблона для игрока. Знакомство с Input системой. Что такое Mapping? Создание базового управления персонажем: движение, управление камерой, прыжок.

4.2. Skeletal Mesh. Скелеты и кости моделей

Что такое Skeletal Mesh? Что такое скелет и для чего он нужен. Кости.

4.2. Анимация в Unreal Engine. “Оживляем” персонажа. Добавление анимаций

Основы анимации в Unreal Engine. Что такое Animation Blueprint. Смешивание анимаций. Добавление анимаций персонажу.

4.3. Работа со звуком в движке

Основы работы со звуком в Unreal Engine. Добавление звуков для персонажа.

4.4. UX/UI. Работа с пользовательским интерфейсом

Что такое UX/UI? Основы создания UI элементов в Unreal Engine. Что такое Widget.

4.5. Landscape. Foliage. Water System

Разбор инструментов для создания ландшафта в Unreal Engine. Режим Landscape. Режим Foliage. Основы работы с Water System. Создание собственного ландшафта.

4.6. AI. NavMesh. Поиск пути на местности

Знакомство с NavMesh. Поиск пути на местности. Разработка базового ИИ.

4.7. VFX. Создание эффектов с помощью Niagara

Что такое VFX? Знакомство с Niagara System. Принципы создания собственных эффектов.

4.8. MetaHuman. Создание реалистичных персонажей

Что такое MetaHuman? Использование редактора персонажей. Добавление персонажей в движок. Принципы анимирования и использования MetaHuman.

5. Разработка полноценной игры на Blueprint'ax

5.1. Создание ландшафта игровой локации

Разработка ландшафта для дальнейших игровых действий. Создание острова и океана вокруг. Высаживание растительности.

5.2. Создание игрока. Перемещение, звуки, анимация

Разработка главного героя игры. Реализация перемещения, анимаций движений, звуков.

5.3. Разработка системы характеристик игрока

Реализация системы характеристик для игрока: здоровье, голод, выносливость.

5.4. Инвентарь. Реализация системы хранения предметов

Реализация системы инвентаря и хранения предметов. Разработка UI для инвентаря. Реализация представления предметов. Отображение предметов в слотах.

5.5. Система взаимодействия. Подбор ресурсов, взаимодействие с предметами

Реализация интерактивных предметов. Что такое интерфейс? Реализация функционала для интерактивных предметов. Подбор ресурсов и добавление их в инвентарь.

5.6. Крафтинг. Создание объектов с использованием ресурсов

Реализация системы крафта. Создание предметов из собранных материалов.

5.7. Система строительства. Строительство зданий

Реализация системы строительства зданий. Выбор постройки. Спавн элементов. Привязка к сетке.

5.8. ИИ животных. Добавление враждебных мобов

Реализация ИИ для мобов. Поведенческие паттерны. Работа с Behaviour Tree. Реализация враждебных мобов в игре.

5.9. PCG Spawner. Процедурная генерация объектов

Что такое PCG Spawner? Процедурная генерация объектов.

5.10. Система смены дня и ночи. Генерация погодных условий

Реализация цикла смены дня и ночи. Обновление Skybox. Генерация погодных условий.

6. Программирование игровой логики с использованием C++

6.1. Изучение основ C++. Сравнение с Blueprint

Знакомство с языком программирования C++. Сравнение C++ и Blueprint. Изучение основ языка программирования C++.

6.2. Реализация мини-игры с использованием C++

Закрепление на практике полученных знаний на примере реализации игры.

7. Проектная деятельность

7.1. Разработка концепта выпускного проекта.

Подготовка к выполнению проектной работы. Анализ понравившихся идей. Разработка концепта для выпускного проекта: идея, стилистика, сеттинг и т.д. Презентация идеи. Написание дизайн-документа. Написание сценария на 5-10 минут геймплея.

7.2. Работа над проектом. Разработка игровых механик

Работа над выпускным проектом: разработка игровых механик, поиск или создание ассетов и т.д.

7.3. Полировка игры. Подготовка к сборке. Финальный билд

Полировка геймплея игры. Доработка игровых механик. Подготовка к сборке под Windows. Финальный билд. Тест игры.

8. Итоговое тестирование

8.1. Подготовка к защите работы

Подготовка к защите работы, итоговому тестированию. Искусство презентации проекта. Что такое Elevator-speech? На что стоит обратить внимание при презентации проекта публике.

8.2. Защита работы, итоговое тестирование

Прохождение теста. Защита работы перед экспертами. Сбор обратной связи.