

Автономная некоммерческая организация
«Красноярский детский технопарк «Кванториум»

РЕКОМЕНДОВАНО
методическим советом

Протокол № 13
от «30» мая 2025 г.

УТВЕРЖДАЮ

Генеральный директор
Кениг С.Р.

Приказ № 42
от «30» мая 2025 г.



Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Разработка программного обеспечения с использованием Unity»

Срок реализации:

1 год

Возраст детей:

13-18 лет

Составитель программы:

Цыганов М.В.

г. Красноярск, 2025 г.

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Разработка программного обеспечения с использованием Unity» (далее – программа), технической направленности, нацелена на изучение языка программирования C#, так как он является одним из наиболее востребованных на рынке труда и после его изучения, можно легко осваивать другие языки программирования. Помимо изучения языка программирования, обучающиеся знакомятся с такой сферой его практического применения, как разработка мобильных приложений.

Обучающиеся создадут свои первые приложения и игры, углубят знания в области объектно-ориентированного программирования.

Особенностью программы является то, что она направлена на развитие интереса у обучающихся к научно-техническому творчеству, на формирование первоначальных умений и навыков в проектировании и разработке программного обеспечения.

По результатам обучения, обучающиеся овладеют основами программирования на языке C#, основами разработки приложений на Unity.

1.1 Актуальность программы

Данная программа нацелена на обучающихся, желающих углубиться в изучение программирования, позволит увидеть всю многогранность информационных технологий.

Изучение основ программирования способствуют формированию современного научного мировоззрения, развитию интеллектуальных способностей и познавательных интересов обучающихся, развивает способности к самообразованию, умению ставить задачи, планировать их выполнение, умению разбивать решение задачи на подзадачи, способность работать с информацией: находить, оценивать и использовать информацию из различных источников. Программирование - это не просто знание какого-то языка или технологии, это способность сформулировать проблему, найти возможные решения этой проблемы путём определения алгоритмов её решения. После прохождения данной программы обучающемуся будет проще определить наиболее интересующую его область информационных технологий и в дальнейшем уже осознанно подходить к выбору дальнейшего направления обучения.

1.2 Отличительные особенности

Данная программа разработана в соответствии с нормативными правовыми актами в области образования:

Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;

Указом Президента Российской Федерации «О национальных целях развития Российской Федерации на период до 2030 года»;

Концепцией развития дополнительного образования до 2030 года (утверждена распоряжением Правительства РФ от 31.03.2022 № 678-р);

Стратегией развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства Российской Федерации от 29.05.2015 № 996-р);

Планом мероприятий по реализации в 2021 - 2025 годах Стратегии развития воспитания в Российской Федерации на период до 2025 года (утвержден распоряжением Правительства Российской Федерации от 12.11.2020 № 2945-р);

Приказом Министерства образования и науки Российской Федерации от 23.08.2017 № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;

Приказом Министерства просвещения России от 09.11.2018 № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;

Приказом Министерства просвещения РФ от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;

Приказом Министерства Просвещения Российской Федерации от 30.09.2020 № 533 «О внесении изменений в порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденный приказом Министерства Просвещения Российской Федерации от 09.11.2018 № 196»;

Постановлением Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;

Письмом Министерства образования и науки РФ от 18.11.2015 № 09-3242 «О направлении информации» (с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы))».

Программа охватывает основы языка программирования C# и работы в среде разработки Unity. Обучение начинается с изучения базовых конструкций языка C#, затем изучаются основы работы со средой Unity. Кроме того, в рамках изучения программы, обучающиеся приобретут базовые навыки

ведения проектной деятельности, а также навыки командной работы и представления результатов своей работы.

1.3 Адресат программы, требования к обучающимся, возрастные особенности группы

Программа «Разработка программного обеспечения с использованием Unity» адресована обучающимся успешно освоившим ДООП «Основы программирования».

Набор на Программу осуществляется в соответствии с Порядком организации образовательной деятельности по дополнительным общеобразовательным общеразвивающим программам и Правилами приема и отчисления обучающихся автономной некоммерческой организации «Красноярский детский технопарк «Кванториум».

Программа «Разработка программного обеспечения с использованием Unity» рассчитана на обучающихся 13-18 лет.

1.4 Педагогическая целесообразность

Выбор своего направления в IT-сфере - это очень сложная задача и поэтому программа построена таким образом, чтобы увлечь обучающихся программированием, найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни и понять в каком направлении им хотелось бы развиваться дальше. Многие люди стремятся заняться информационными технологиями, не предполагая о всех возможностях этой области.

Изучение основ программирования связано с целым рядом умений и навыков, которые могут помочь в дальнейшем не только будущим инженерам и программистам, но и тем, чья жизнь не будет напрямую связана с написанием программ.

Изучая программирование, обучающиеся получают глубокое понимание принципов работы компьютера, организации ввода, вывода и хранения информации, принципов построения диалоговых приложений, познают азы профессии программиста.

В рамках данной программы обучающийся познакомится с языком программирования C#, после чего увидит, как создаются мобильные и десктопные приложения и сами попробуют силы в разработке.

Содержание программы раскрывает процесс достижения результатов обучения по всем видам деятельности обучающихся: развитие интеллектуальных способностей и познавательных интересов обучающихся через овладение технологиями создания программных продуктов.

Процесс обучения выстроен с учетом используемых современных образовательных технологий: информационные технологии обучения, технология проектного обучения, что отражено в дидактических принципах

обучения (индивидуальность, доступность, преемственность, результативность и др.). Наряду с технологичностью обучения в образовательном процессе решаются задачи организации и управления. В соответствии с учебным планом программы на каждом этапе обучения разработаны занятия, наиболее эффективные для развития познавательного потенциала обучающихся. Виды деятельности обучающихся отражены в предлагаемых формах организации обучения, что и определяет продолжительность по времени, динамичную структуру занятия, способы организации процесса обучения и его методическое оснащение. В техническом творчестве и проектной технологии ведущее место занимают интерактивные и активные методы обучения, обеспечивающие эффективное достижение поставленной цели обучения (в данной предметной области) и отслеживание результативности обучения на всех этапах.

1.5 Объем/срок освоения программы

Программа рассчитана на 1 год обучения. Годовая нагрузка на обучающегося составляет 144 часа.

1.6 Форма обучения, виды занятий и режим занятий

Учебные занятия проходят по очной форме обучения. Режим занятий – 2 раза в неделю по 2 академических часа (1 академический час 40 минут) с обязательным перерывом.

При проведении занятий используются комбинированные занятия – изложение нового материала, проверка пройденного материала, закрепление полученных знаний, самостоятельная работа.

При проведении занятий используются три формы работы:

- демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном или экранами компьютеров на ученических рабочих местах;
- фронтальная, когда обучающиеся синхронно работают под управлением педагога;
- самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия.

Повторение и усвоение пройденного материала осуществляется через контрольные и проверочные работы, анализ полученных результатов.

Закрепление знаний, умений и навыков через постановку задачи и самостоятельную работу обучающегося под руководством педагога.

Применение полученных знаний и навыков через прикладную работу обучающегося, использующего на практике приобретенные компетенции.

В качестве основного метода обучения используется проектный метод.

1.7 Цель и задачи программы

Цель: создание мобильного или десктопного приложения при помощи языка программирования C# в среде разработки Unity.

Задачи:

- сформировать навыки программирования на языке C#;
- сформировать навыки работы в среде разработки Unity;
- сформировать навыки ведения проектной деятельности.

1.8 Планируемые результаты

Образовательные результаты

По результатам обучения, обучающиеся овладевают основами технологий разработки программного обеспечения на языке C# с использованием среды разработки Unity.

Обучающиеся будут знать:

- основы языка программирования C#;
- принципы организации проектной деятельности;
- принципы объектно-ориентированного программирования;
- основные алгоритмы и структуры данных.

Обучающиеся будут уметь:

- создавать приложения с использованием языка программирования C#;
- создавать приложения с использованием среды разработки Unity;
- осуществлять проектную деятельность;
- применять изученные алгоритмы и структуры данных.

Сформированные навыки:

- программирования на языке C#;
- применения различных алгоритмов и структур данных;
- использования среды разработки Unity;
- ведения проектной деятельности.

Личностные результаты

Обучающиеся приобретут навыки самостоятельной организации своей деятельности; саморазвития и самовоспитания.

У обучающихся формируется готовность и способность к самостоятельной, творческой деятельности, к образованию, в том числе самообразованию; готовность к осознанному выбору будущей профессии.

Метапредметные результаты

Умение планировать; умение анализировать; умение алгоритмизировать.

1.9 Формы подведения итогов

По итогам каждого ключевого раздела проводится промежуточная аттестация в форме проверочной работы.

Аттестация по итогам освоения программы проводится в конце года и представляет собой защиту индивидуального или группового проекта.

2. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ п/п	Название раздела, модуля	Количество академических часов			Форма контроля
		всего	теория	практика	
1	Вводный урок. Правила и техника безопасности при работе с компьютером.	2	2	0	
2	Раздел 1. Язык программирования С#	26	10	16	
2.1	Переменные. Типы данных	6	2	4	
2.2	Операторы. Условия	6	2	4	
2.3	Операторы. Циклы	6	2	4	
2.4	Объектно-ориентированное программирование. Классы, поля и методы	8	4	4	Проверочная работа
3	Раздел 2. Разработка на Unity	38	20	18	
3.1	Знакомство со средой разработки	2	2	0	
3.2	Создание проекта. Знакомство с интерфейсом	4	3	1	
3.3	Работа с объектами в сцене	4	2	2	
3.4	Свойства объектов	4	2	2	
3.5	Префабы	2	1	1	
3.6	Управление объектами через скрипты	6	2	4	
3.7	Триггеры и коллайдеры	4	2	2	
3.8	Функции для работы со временем	4	2	2	
3.9	Управление сценами	2	1	1	
3.10	Корутины	4	2	2	
3.11	Сборка проекта	2	1	1	Проверочная работа
4	Раздел 3. Проектная деятельность	78	12	66	
4.1	Методология проектной деятельности	4	4	0	
4.2	Выбор темы проекта. Формулирование проблемы	4	0	4	
4.3	Реализация проекта	64	8	56	
4.4	Подготовка к защите проекта	4	0	4	
4.5	Аттестация по итогам освоения программы	2	0	2	Защита проекта

3. СОДЕРЖАНИЕ

1. Соблюдение правил техники безопасности и санитарно-гигиенических норм при работе с электрооборудованием и учебно-методическими материалами. Общие правила безопасности в образовательном учреждении. Основы техники безопасности при работе с электрическими приборами.

2. Раздел 1. Язык программирования C#

2.1 Тема: Переменные. Типы данных

Теория: понятие переменной. Понятие типа данных. Какие типы данных существуют.

Практика: объявление и инициализация переменных. Действия над переменными. Представление переменных в памяти компьютера.

2.2 Тема: Операторы. Условия

Теория: определение ветвления. Полное ветвление. Неполное ветвление. Условный оператор if-else. Базовые логические операции И, ИЛИ, НЕ. Оператор множественного выбора switch-case.

Практика: формирование условий. Применение условного оператора и оператора множественного выбора при решении задач.

2.3 Тема: Операторы. Циклы

Теория: определение цикла. Понятие итерации. Виды циклов: цикл с параметром, цикл с условием. Циклические операторы for, while.

Практика: применение циклических операторов при решении задач.

2.4 Тема: Объектно-ориентированное программирование. Классы, поля и методы

Теория: понятие объектно-ориентированного программирования. Понятие класса, полей класса, методов класса. Модификаторы доступа для полей и методов класса. Что такое наследование.

Практика: применение объектно-ориентированного подхода при решении задач.

3. Раздел 2. Разработка на Unity

3.1 Тема: Знакомство со средой разработки

Теория: среда разработки Unity. История. Возможности среды. Демонстрация примеров применения.

3.2 Тема: Создание проекта. Знакомство с интерфейсом

Теория: как создать проект. Выбор шаблона при создании проекта. Отличия шаблонов. Окно редактора. Области Inspector, Console, Hierarchy, Scene, Game. Основные меню редактора.

Практика: самостоятельное ознакомление с интерфейсом среды разработки Unity.

3.3 Тема: Работа с объектами в сцене.

Теория: определение сцены, понятие объекта. Создание и добавление объектов в сцену. Изменение параметров объектов с помощью инструментов редактора (масштабирование, перемещение, вращение)

Практика: самостоятельное добавление объектов в сцену. Изменение их параметров.

3.4 Тема: Свойства объектов

Теория: понятие свойств(компонентов) объектов. Добавление, удаление и изменение компонентов объекта. Какие компоненты бывают.

Практика: самостоятельное изменение свойств объектов.

3.5 Тема: Префабы

Теория: понятие префаба. Создание и редактирование префабов.

Практика: самостоятельное создание и редактирование префаба.

3.6 Тема: Управление объектами через скрипты

Теория: как управлять объектами через скрипты на языке C#. Какие свойства и как можно изменять.

Практика: решение задач на изменение параметров объектов с помощью скриптов.

3.7 Тема: Триггеры и коллайдеры

Теория: понятия триггера и коллайдера. Взаимодействие объектов. Обработка столкновений объектов.

Практика: решение задач с описанием взаимодействия объектов.

3.8 Тема: Функции для работы со временем

Теория: класс Time. Поля и методы класса Time. Приостановка выполнения скриптов.

Практика: применение класса Time для решения задач, связанных с обработкой поведения физических объектов. Реализация таймеров.

3.9 Тема: Управление сценами

Теория: переход между сценами. Класс SceneManager. Методы класса SceneManager.

Практика: применение методов класса SceneManager при решении задач. Создание меню.

3.10 Тема: Корутины

Теория: понятие корутины. способ запускать функции, которые должны работать параллельно в течение некоторого времени.

Практика: применение корутин при решении задач

3.11 Тема: Сборка проекта

Теория: конфигурация проекта перед финальной сборкой. Выбор целевой платформы. Финальная сборка.

Практика: финальная сборка проекта для целевой платформы.

4. Раздел 3. Проектная деятельность

4.1 Тема: Методология проектной деятельности

Теория: введение в проектную деятельность. Понятие проекта. Что такое проблема, цель и задачи. Важность выполнения обзора аналогов. Методологии SCRUM и Kanban.

4.2 Тема: Выбор темы проекта. Формулирование проблемы

Практика: формулирование проблемы и формулирование темы проекта для реализации. Предложение решения проблемы. Обзор аналогов.

4.3 Тема: Реализация проекта

Теория: определение целевой аудитории проекта. Основы экономического планирования.

Практика: Реализация проекта. Подготовка технической документации по проекту.

4.4 Тема: Подготовка к защите проекта

Практика: подготовка презентации проекта.

4.5 Тема: Аттестация по итогам освоения программы.

Практика: Презентация и защита проекта.

4. СПИСОК ЛИТЕРАТУРЫ

Джозеф Албахари, Бен Албахари. С# 6.0. Справочник. Полное описание языка = C# 6.0 in a Nutshell: The Definitive Reference. — М.: «Вильямс», 2018. — 1040 с. — ISBN 978-5-8459-2087-4. — ISBN 978-1-491-92706-9.

Хокинг, Джозеф. Unity — в действии. Мультиплатформенная разработка на С#: [рус.]. — 2. — СПб : Питер, 2016. — 336 с. — ISBN 978-1617292323.

Unity Learn [Электронный ресурс] — Режим доступа: <https://learn.unity.com/>

Unity - Manual: Unity User Manual [Электронный ресурс] — Режим доступа: <https://docs.unity3d.com/Manual/index.html>

5. ИНФОРМАЦИОННОЕ И МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОСНАЩЕНИЕ ДООП

Информационное оснащение:

Торн А. Искусство создания сценариев в Unity / пер. с англ. Р. Н. Рагимова – М.: ДМК Пресс, 2016. – 360 с.: ил. (Электронная книга)

Хокинг Дж. Unity в действии. Мультиплатформенная разработка на C# / Пер. с англ. И. Рузмайкиной. — СПб.: Питер, 2016. — 336 с.: ил. — (Серия «Для профессионалов»). (Электронная книга)

Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. – СПб.: Питер, 2017. – 288 с.: ил. – (серия «Библиотека программиста»). (Электронная книга)

Материально-техническое оснащение:

Помещения для лабораторных занятий и самостоятельной работы обучающихся должны быть оснащены персональными компьютерами с установленной средой разработки программного обеспечения на языке объектно-ориентированного программирования Microsoft Visual Studio Community 2022 (и выше), Microsoft Office 2016 (и выше), Unity 2022.3(LTS), с возможностью подключения к сети Интернет и доска с маркером или интерактивная доска.

Помещение для занятий лекционного типа должно быть оснащено мультимедийным проектором и персональными компьютерами с установленной средой разработки программного обеспечения на языке объектно-ориентированного программирования Microsoft Visual Studio Community 2022 (и выше), Microsoft Office 2016 (и выше), Unity 2022.3(LTS), с возможностью подключения к сети Интернет, доска с маркером или интерактивная доска.

6. Оценочные материалы

Критерии оценивания промежуточных проверочных работ:

Критерий	Оценка
Программа выполняет требуемую функцию.	Зачет
Программа не выполняет требуемую функцию	Незачет

Критерии оценивания проекта:

№ п/п	Объект оценки	Критерии	Баллы
1.	Оценка изготовленного изделия / разработанного цифрового продукта	Новизна и актуальность темы проекта	от 0 до 10
		Привлекательность и оригинальность внешнего вида (изготовленного изделия, разработанного цифрового продукта)	от 0 до 10
		Работоспособность и (или) возможность для внедрения (изготовленного изделия, разработанного цифрового продукта)	от 0 до 10
		Качество (изготовленного изделия, разработанного цифрового продукта)	от 0 до 10
		Перспективность и конкурентоспособность (изготовленного изделия, разработанного цифрового продукта))	от 0 до 10
2.	Оценка описания проекта	Точность формулировки концепции	от 0 до 7
		Формулировка цели и задач проекта	от 0 до 7
		Полнота описания изделия или цифрового продукта	от 0 до 7
		Демонстрация основных функций продукта	от 0 до 7
		Соответствие результата поставленной цели	от 0 до 7
		Дальнейшее развитие проекта	от 0 до 7
		Рефлексия и оценка работы команды	от 0 до 7
3.	Оценка защиты проекта	Соблюдение регламента публичной защиты	от 0 до 5
		Качество подачи материала и представления	от 0 до 5
		Понимание сути задаваемых вопросов и аргументированность ответов докладчика	от 0 до 5
		Качество презентации и презентационных материалов	от 0 до 5

Программа считается освоенной, если итоговый балл по результатам защиты проекта более 59.

7. ПРИЛОЖЕНИЯ

Приложение 1

Примеры заданий для практического решения по темам:

2. Раздел 1. Язык программирования C#

2.1 Тема: Переменные. Типы данных

Объявить 2 переменные целочисленного типа, вычислить их сумму, разность, произведение и частное. Результаты вычислений вывести в консоль.

2.2 Тема: Операторы. Условия

Объявить 3 переменные целочисленного типа, найти среди них наибольшее и наименьшее значения. Вывести найденные значения в консоль.

2.3 Тема: Операторы. Циклы

Вывести в столбик в консоль N случайных чисел.

2.4 Тема: Объектно-ориентированное программирование. Классы, поля и методы

Объявить класс «Фигура», содержащий поля, в которых хранится периметр и площадь фигуры. Создать 3 дочерних класса «Прямоугольник», «Треугольник» и «Окружность». В классе «Прямоугольник» должны быть поля, содержащие длину и ширину, а также в классе «Треугольник» должны быть поля, содержащие длины сторон. В классе «Окружность» должно быть поле, содержащее радиус окружности. Для всех дочерних классов должны быть реализованы методы, вычисляющие площадь и периметр.

3. Раздел 2. Разработка на Unity

3.2 Тема: Создание проекта. Знакомство с интерфейсом

Практика: самостоятельное ознакомление с интерфейсом среды разработки Unity.

Продемонстрировать местоположение окон «Иерархия», «Инспектор», «Сцена», «Игра», «Консоль», «Проект».

3.3 Тема: Работа с объектами в сцене.

Средствами редактора создать «пирамидку» из 5 объектов «Cube», в которой размеры нижнего куба 10*1*10, длина и ширина кубов, стоящих выше должны отличаться на 2.

3.4 Тема: Свойства объектов

Средствами редактора создать объект «Cube», добавить на него компонент «Rigidbody», в свойствах этого компонента запретить объекту вращение по всем осям и запретить перемещение по осям X и Z

3.5 Тема: Префабы

Создать объект «Sphere», добавить на него компонент «Rigidbody» и создать префаб этого объекта.

3.6 Тема: Управление объектами через скрипты

Создать объект «Cube», создать скрипт, который будет вращать этот объект вокруг оси Y.

3.7 Тема: Триггеры и коллайдеры

Создать 2 объекта «Cube», создать скрипт, который будет при столкновении этих объектов выводить в консоль сообщение «Столкновение».

3.8 Тема: Функции для работы со временем

Реализовать скрипт, который будет каждые 10 секунд изменять состояние объекта «Активно»/ «Неактивно».

3.9 Тема: Управление сценами

Создать две сцены, в каждой создать одну кнопку. Реализовать скрипт, который при нажатии на кнопку на одной сцене будет запускать вторую сцену.

3.10 Тема: Корутины

Реализовать эффект затухания экрана при помощи корутины.

3.11 Тема: Сборка проекта

Настроить конфигурацию сборки для платформы Windows, добавить в сборку сцену, произвести сборку проекта.

4. Раздел 3. Проектная деятельность

4.2 Тема: Выбор темы проекта. Формулирование проблемы

Сформулировать тему проекта, возможные темы проектов представлены в приложении 4.

4.3 Тема: Реализация проекта

Реализация проекта.

4.4 Тема: Подготовка к защите проекта

Подготовка презентации проекта.

4.5 Тема: Аттестация по итогам освоения программы.

Презентация и защита проекта.

Приложение 1

Примеры заданий и тестовые данные для проведения промежуточного контроля успеваемости по разделу 1:

Задача:

Напишите программу, которая в последовательности натуральных чисел определяет сумму чисел, оканчивающихся на 4. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, оканчивающееся на 4. Количество чисел не превышает 1000. Введённые числа не превышают 30 000. Программа должна вывести одно число — сумму чисел, оканчивающихся на 4.

Пример тестовых данных:

Входные данные	Выходные данные
3 14 25 24	38

Задача:

Напишите программу, которая в последовательности натуральных чисел определяет количество чисел, кратных 6. Программа получает на вход количество чисел в последовательности, а затем сами числа. В последовательности всегда имеется число, кратное 6.

Пример тестовых данных:

Входные данные	Выходные данные
3 14 25 24	2

Задача:

Напишите программу, которая в последовательности натуральных чисел определяет количество чисел, кратных 6 и оканчивающихся на 4. Программа получает на вход количество чисел в последовательности, а затем сами числа. Количество чисел не превышает 1000. Введённые числа по модулю не превышают 30 000. Программа должна вывести одно число: количество чисел, кратных 6 и оканчивающихся на 4.

Пример тестовых данных:

Входные данные	Выходные данные
3 14 25 24	2

Примеры заданий для проведения промежуточного контроля успеваемости по разделу 2:

Задача:

Реализовать перемещение игрового объекта по сцене при помощи клавиш со стрелками. Объект должен перемещаться в соответствии с нажатой клавишей: стрелка вверх – объект двигается вперед (от камеры), стрелка вниз – назад (к камере), стрелка влево – влево, стрелка вправо – вправо.

Задача:

Реализовать перемещение трех объектов в случайном направлении в ограниченной области (область ограничивается координатами: $X [-10;10]$, $Y [0;0]$, $Z [-10;10]$). При столкновении объектов вывести в консоль сообщение: «Столкновение!».

Задача:

Реализовать меню с тремя кнопками, обеспечивающими выбор сцены, на которую должен осуществиться переход. В сценах должны находиться различные игровые объекты.

Возможные темы проектов:

Мобильное приложение для изучения физики в игровой форме.

Приложение дополненной реальности для изучения устройства автомобиля.

Разработка компьютерной игры в жанре Roguelike.

Интерактивная азбука.

Интерактивная карта Красноярского края. Приложение дополненной реальности.