

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ  
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ  
«ЦЕНТР ДЛЯ ОДАРЕННЫХ ДЕТЕЙ «ПОИСК»

РЕКОМЕНДОВАНА

педагогическим советом

Протокол от «\_\_» \_\_\_\_\_ 2020

УТВЕРЖДАЮ

Зав. филиалом

Т.В. Ларина

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

**«ИТ-КВАНТУМ»**

Возраст обучающихся: 11-18 лет

Срок реализации: 2 года

**Составители программы:**

Бондаренко Ангелина  
Викторовна,  
педагог дополнительного  
образования

Кривенко Игорь Вячеславович,  
педагог дополнительного  
образования

Ставрополь,

2020

## ОГЛАВЛЕНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	3
УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН	18
СОДЕРЖАНИЕ КУРСА «IT-КВАНТУМ»	24
МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММЫ «IT-КВАНТУМ»	46
СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ	51

## **ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

В настоящее время процесс информатизации проявляется во всех сферах человеческой деятельности. Использование современных информационных технологий является необходимым условием успешного развития как отдельных отраслей, так государства в целом. Создание, внедрение, эксплуатация, а также совершенствование информационных технологий немислимо без участия квалифицированных и увлеченных специалистов. Стремительный рост информационных технологий ставит новые задачи перед образованием и наукой, изучение классических дисциплин недостаточно для решения таких задач. В связи с этим актуальной задачей является подготовка специалистов сферы информационных технологий в соответствии с профессиональными требованиями динамично развивающихся отраслей. При этом требуется постоянная актуализации знаний, приобретения новых компетенций, формирование нового типа мышления. В этом смысле важнейшую роль играет процесс изучения базовых основ информационных технологий еще в школьном возрасте.

Подготовка национально-ориентированного кадрового резерва для наукоемких и высокотехнологичных отраслей экономики вызвана запросом прямых работодателей. Система научно-технического просвещения через привлечение детей к изучению и практическому применению наукоемких технологий формирует компетенции эффективного управления проектной деятельностью, которое в современном мире становится наиболее актуальной метапредметной задачей образования.

### **Направленность программы**

Программа имеет техническую направленность. Однако, для

многостороннего развития личности, в ней отражены следующие аспекты изучения:

1. Технологический. Содержание программы рассматривается как средство формирования образовательного потенциала, позволяющего развивать наиболее передовые на сегодняшний день технологии — информационные, интегрирующие в себе науку, технологию, инженерное дело.

2. Социально-психологический. Содержание программы рассматривается как средство формирования навыков эффективной деятельности в проекте, успешной работы в команде, развития стрессоустойчивости, эмпатических способностей, умению распределять приоритеты и пользоваться инструментами планирования, а также креативного и инженерно-технического мышления.

### **Актуальность программы**

Актуальность данной программы состоит в том, что она составлена с учётом современных потребностей рынка в специалистах в области информационных технологий. Учитывается и междисциплинарность информационных технологий. Предусмотрено приобретение навыков в области применения информационных технологий в биологии, робототехнике, дизайне.

Данная программа дает возможность детям творчески мыслить, находить самостоятельные индивидуальные решения, а полученные умения и навыки применять в жизни. Развитие творческих способностей помогает также в профессиональной ориентации подростков.

Современное информационное общество требует постоянного обновления и расширения профессиональных компетенций. Необходимо улавливать самые перспективные тенденции развития мировой конъюнктуры, шагать в ногу со временем. В процессе реализации данной

программы формируются и развиваются умения и навыки в области информационных технологий, новые компетенции, которые необходимы всем для успешности в будущем.

### **Новизна программы**

Новизна программы состоит в том, что она учитывает новые технологические уклады, которые требуют иных способов мышления и тесного взаимодействия при постоянном повышении уровня междисциплинарности проектов, а также использует принципы вытягивающей модели обучения.

Введение в дополнительное образование общеобразовательной и общеразвивающей программы «IT–квантум» с использованием таких методов, как командная работа, поиск проблем и их практическое решение, анализ и обобщение опыта, подготовка исследовательских и инженерно-технических проектов и их защита, элементы соревнований, неизбежно изменит картину восприятия учащимися технических дисциплин, переводя их из разряда умозрительных в разряд прикладных.

### **Цели программы:**

- привлечь обучающихся к исследовательской и изобретательской деятельности в сфере IT-технологий;
- развить интерес обучающихся к информационным технологиям;
- помочь реализовать творческие идеи обучающихся в области программирования, электроники или веб-проектирования в виде проектов различного уровня сложности.

### **Задачи:**

#### **Образовательные:**

- дать представление о значении информационных технологий в развитии общества и в изменении характера труда человека;
- познакомить с основными понятиями информатики непосредственно в процессе создания информационного продукта;

- выработать навыки применения средства ИТ в повседневной жизни, при выполнении индивидуальных и коллективных проектов, при дальнейшем освоении будущей профессии;
- познакомить с базовой частью математического аппарата, применяемого в программировании современных электронных вычислительных машин и микропроцессорной техники;
- обучить методам программирования на языках, применяемых в современной вычислительной технике, и работе в интегрированных средах разработки;
- обучить навыкам конструирования сложных систем, управляемых микроконтроллерами и миникомпьютерами;
- сформировать навыки проектирования игровых приложений, создания программ и их отладки;
- научить проектировать, осуществлять верстку и программировать сайты разного уровня сложности;
- научить проектировать, настраивать локальную сеть и монтировать оборудование;
- формировать и развивать навыки публичного выступления.

***Воспитательные:***

- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций и электронных устройств;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- привить информационную культуру: ответственное отношение к информации с учетом правовых и этических аспектов её распространения, избирательного отношения к полученной информации;
- формировать правильное восприятие системы ценностей, принципов, правил информационного общества;

- формировать потребность в самостоятельном приобретении и применении знаний, потребность к постоянному саморазвитию;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность.

***Развивающие:***

- способствовать развитию творческих способностей учащихся, познавательных интересов, развитию индивидуальности и самореализации;
- расширять технологические навыки при подготовке различных информационных материалов;
- развивать познавательные способности ребенка, память, внимание, пространственное мышление, аккуратность и изобретательность при работе с техническими устройствами, создании электронных устройств и выполнении учебных проектов;
- формировать творческий подход к поставленной задаче;
- развивать навыки инженерного мышления, умения работать как по предложенным инструкциям, так и находить свои собственные пути решения поставленных задач;
- развивать навыки эффективной деятельности в проекте, успешной работы в команде;
- развивать стрессоустойчивость;
- развивать способности к самоанализу, самопознанию;
- Развить общекультурные компетенции у обучающихся через активное использование ресурсов организаций культуры, искусства и истории;
- формировать навыки рефлексивной деятельности.

## **Отличительные особенности программы**

Особенностью данной программы является использование современных методов и технологий в обучении, а именно кейс-метода и командная проектная деятельность.

Кейс представляет собой описание конкретной реальной ситуации, подготовленное по определенному формату и предназначенное для обучения учащихся анализу разных видов информации, ее обобщению, навыкам формулирования проблемы и выработки возможных вариантов ее решения в соответствии с установленными критериями. Кейсовая технология (метод) обучения – это обучение действием. Суть кейс-метода состоит в том, что усвоение знаний и формирование умений и навыков есть результат активной самостоятельной деятельности учащихся по разрешению противоречий, в результате чего и происходит творческое овладение профессиональными знаниями, навыками, умениями и развитие мыслительных способностей.

Эта техника обучения использует описание реальных экономических, социальных и бизнес-ситуаций. Кейсы основываются на реальном фактическом материале или же приближены к реальной ситуации. Кейс технология объединяет в себе одновременно и ролевые игры, и метод проектов, и ситуативный анализ.

Занятия строятся с учётом индивидуальных особенностей воспитанников, что позволяет заинтересовать, увлечь каждого ребенка, раскрыть его творческие способности.

Проектная деятельность включает в себя познавательную, учебную, исследовательскую и творческую деятельность, в результате которой появляется решение задачи, которое представлено в виде проекта. Такой вид работ направлен на решение интересной проблемы, сформулированной самими учащимися. Результат этой деятельности — найденный способ



решения проблемы — носит практический характер и значим для самих открывателей.

### **Категория обучающихся**

Программа предназначена для детей, проявляющих интерес к программированию, конструированию, информационным технологиям в целом, стремящимся к саморазвитию, профессиональному самоопределению.

**Возраст обучающихся:** 11 — 18 лет.

**Наполняемость группы:** 15 человек.

**Состав группы:** разновозрастной.

### **Условия приема детей**

На курсы программы зачисляются все желающие при наличии свободных мест.

**Срок реализации программы:** 2 года.

**Форма реализации программы** — очная с использованием электронного обучения. Под электронным образованием понимается реализация образовательных программ с использованием информационно - образовательных ресурсов, информационно-коммуникационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу информационно-образовательных ресурсов и взаимодействие участников образовательного пространства.

Дистанционный формат образования рассматривается как пространство самоактуализации человека и в корне отличается от традиционной системы обучения. Использование в процессе учёбы ультрасовременных компьютерных технологий положительно влияет на степень восприятия информации человеком и улучшает его когнитивные функции. Общение с преподавателем в удалённом формате полностью

меняет стандартные роли обоих, позиционируя обучающегося как «координатора знаний», тогда как преподаватель выступает в роли «интерпретатора знаний».

Преимуществами удалённого формата обучения являются следующие факторы:

1) возможность получить образование вне зависимости от геолокации;

2) охват узкой предметной области;

3) возможность организации дискуссий, групповых работ, совместных проектов;

4) невысокая стоимость (по сравнению с традиционным заочным обучением);

5) возможность совмещения обучения с основной деятельностью;

6) высокое качество учебных материалов;

7) использование новейших компьютерных технологий;

8) повышение интеллектуального и творческого потенциала;

9) высокая скорость обучения;

10) проявление самоорганизации и самостоятельности;

11) быстрое освоение современных компьютерных технологий, «гаджетов»;

12) возможность обучаться у высококвалифицированных преподавателей.

Неоспоримым преимуществом дистанционного обучения является массовость. Система позволяет одновременно охватывать многочисленные аудитории, предоставляя необходимый объем информации вне зависимости от количества слушателей.

## **Формы организации деятельности обучающихся**

При изучении тем программа предусматривает использование фронтальной, индивидуальной и групповой формы учебной работы обучающихся:

✓ фронтальная форма - для изучения нового материала, информация подаётся всей группе до 15 человек;

✓ индивидуальная форма - самостоятельная работа учащихся, педагог может направлять процесс в нужную сторону;

✓ групповая форма помогает педагогу сплотить группу общим делом, способствует качественному выполнению задания, для реализации проектной деятельности в малых группах (3-7 человек).

## **Методы обучения**

Будут реализованы активные методы обучения такие, как:

- метод проектов;
- кейс метод;
- тренинги по формированию гибких методов управления проектами;
- дистанционные занятия.

По способу организации занятий — словесные, наглядные, практические.

**Типы занятий:** теоретические, практические, комбинированные.

## **Режим занятий**

Определяется учебным планом.

## **Ожидаемые результаты**

Основным результатом обучения является достижение высокой информационно-коммуникационной компетентности учащегося.

В результате освоения программы обучающийся должен приобрести

следующие знания, умения и навыки:

***знать:***

- правила работы с компьютером и технику безопасности;
- назначение и функции используемых информационных технологий;
- назначение и основные возможности текстовых и графических редакторов;
- виды компьютерной графики и их особенности;
- правила создания и представления мультимедийной презентации;
- основные свойства алгоритма, типы алгоритмических конструкций: следование, ветвление, цикл; понятие вспомогательного алгоритма;
- основные функции и принцип работы микроконтроллера;
- особенности работы с интегрированной средой разработки для программирования микроконтроллеров Arduino, Iskra JS, Raspberry Pi;
- активные электронные компоненты и способы их подключения;
- базовые и сложные конструкции, способы организации процедур и функций в языках программирования C++, C#, JavaScript, Python 3;
- язык гипертекстовой разметки HTML и основы применения CSS;
- знание гибких техники ведения проектной деятельности и принципов тайм-менеджмента;
- знание этапов и структурных компонентов проекта.

***уметь:***

- создавать информационные объекты, в том числе:
- создавать и использовать различные формы представления информации: формулы, графики, диаграммы, таблицы (в том числе динамические, электронные, в частности – в практических задачах), переходить от одного представления данных к другому;

- создавать рисунки, чертежи, графические представления реального объекта, в частности, в процессе проектирования с использованием основных операций графических редакторов;

- искать информацию с применением правил поиска в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным темам;

- отстаивать свою точку зрения в восприятии элементов общекультурных ценностей;

- отличить традиционные ценности от новых течений в культурном пространстве;

- пользоваться персональным компьютером и его периферийным оборудованием (принтером, сканером, модемом, мультимедийным проектором, цифровой камерой, цифровым датчиком);

- следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;

- использовать приобретенные знания и умения в практической деятельности и повседневной жизни для:

- создания простейших моделей объектов и процессов в виде изображений и чертежей,

- проведения компьютерных экспериментов с использованием готовых моделей объектов и процессов;

- создания информационных объектов, в том числе для оформления результатов учебной работы;

- организации индивидуального информационного пространства, создания личных коллекций информационных объектов;

- передачи информации по телекоммуникационным каналам в учебной и личной переписке, использования информационных ресурсов

общества с соблюдением соответствующих правовых и этических норм;

- эффективно использовать интегрированную среду разработки;
- отбирать методы, приемы и средства организации проектной деятельности;

- разрабатывать программные проекты на основе использования разных технологий программирования;

- разрабатывать и собирать программируемые электронные устройства;

- подключать и программировать работу аналоговых и цифровых датчиков с различными микроконтроллерами;

- проектировать мобильных приложений, создавать программы и выполнять их отладку на мобильных устройствах;

- проектировать и создавать сайты при помощи HTML и CSS;

- писать код программы на языках C++, C#, JavaScript, Python 3;

- работать с локальным сервером;

- работать с CMS WordPress;

- работать с Corona SDK;

- настраивать локальную сеть и оборудование;

- проектировать сеть и монтировать оборудование;

- устанавливать и настраивать UNIX подобные операционные системы;

- формулировать цели, ставить задачи для её достижения в ходе решения проблемных ситуаций;

- эффективно работать в команде;

- презентовать себя, свой продукт, свою команду;

- мыслить творчески, придумывать и воплощать в жизнь свои идеи.

***обладать навыками:***

- исследовательской, проектной и социальной деятельности, строить логическое доказательство;
- использования, создания и преобразования различных символьных записей, схем и моделей для решения познавательных и учебных задач в различных предметных областях, исследовательской и проектной деятельности;
- проектирования, разработки, документирования и представления собственных проектов в составе команды;
- самообразования - периодической оценкой своих успехов и собственной работы самими обучающимися;
- коммуникации - сотрудничество и работа в команде, успешное распределение ролей;
- навыками командной работы;
- навыками применения современных методик и технологий организации проектной деятельности;
- навыками анализа на предмет культурной ценности для общества;
- навыками действий в нестандартных ситуациях в ходе проектной деятельности.

### **Способы определения результативности**

Основным критерием освоения программы является активное участие в проектно-исследовательской деятельности. Программа считается успешно освоенной при условии защиты промежуточных и итоговых проектов разных уровней ограничений группой (3-7 человек) обучающихся.

Уровень сложности задач в кейсах и соответственно их принадлежность к тому или иному модулю определяется уровнем «ограничений». Всего 4 уровня ограничений.

Первый ограничений	уровень	- научить искать информацию; - провести анализ информации; - провести небольшое исследование.
Второй ограничений	уровень	- воплотить в жизнь что-либо известное; - провести углубленное исследование; - выполнить прикладную задачу; - получить мини-артефакт.
Третий ограничений	уровень	- частичная смарт-компонента; - реальные задачи; - глубокий уровень; - практическая реализация; - широкий диапазон направлений; - «полное» отсутствие ограничений.
Четвертый ограничений	уровень	- возможность проведения соревнований; - высокая неопределенность и вариативность итога — результата — устройства; - четкие и ясные рамки и границы; - узкая и сложная прикладная задача.

**Виды контроля:** промежуточный, итоговый.

### **Формы подведения итогов реализации программы**

По окончании 1-го обучения (базового модуля) проводится промежуточная аттестация в форме публичной защиты проектов второго уровня ограничений соответственно. Документальной формой подтверждения итогов промежуточной аттестации является Оценочный лист установленного Центром «Поиск» образца.

По окончании 2-го года обучения (углубленного модуля) проводится итоговая аттестация в форме публичной защиты проектов третьего уровня ограничений соответственно. Документальной формой подтверждения



итогах промежуточной аттестации является документ об образовании установленного Центром «Поиск» образца.

После второго года проводится итоговая аттестация в форме публичной защиты проектов четвертого уровня ограничений. Документальной формой подтверждения итогов итоговой аттестации является документ об образовании установленного Центром «Поиск» образца.

## УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

### Базовый модуль

№	Наименование кейса, темы	Количество часов		
		Теория	Практика	Всего
<b>Раздел 1. Визуальное (блочное) программирование</b>				
1	<b>Командообразование.</b>	<b>1</b>	<b>1</b>	<b>2</b>
2	Тема 1. Командообразование и методы групповой работы.	1	0	1
3	Тема 2. Тренинговое занятие по командообразованию.	0	1	1
4	<b>Кейс 1. Создание игр и анимационных мультфильмов на платформе Scratch</b>	<b>13</b>	<b>18</b>	<b>31</b>
5	Тема 1.1. Основные определения по теме «Алгоритмизация и программирование». Знакомство со средой программирования Scratch.	5	3	8
6	Тема 1.2. Блоки команд «Переменные», «Считывание», «Звук» и «Операторы» на примере создания простой игры.	5	3	8
7	<b>Основы ведения проектной деятельности</b>	<b>1</b>	<b>0</b>	<b>1</b>
8	Тема 1. Основы ведения проектной деятельности.	1		1
9	Тема 1.3. Постановка проблемной ситуации, поиск путей решения. Организация командной работы.	1	1	2
10	Тема 1.4. Работа над проектом.		7	7
11	Тема 1.5. Подготовка к публичной демонстрации и защите результатов кейса.	2	3	5
12	Защита проектов. Рефлексия		1	1
<b>Раздел 2. Электронные устройства на основе микропроцессора Arduino</b>				
13	<b>Тайм-менеджмент.</b>	<b>1</b>	<b>0</b>	<b>1</b>
14	Тема 1. Тайм-менеджмент. Технология управления временем.	1	0	1

15	<b>Кейс 2. Создание элементов умного дома на базе микроконтроллера Arduino</b>	<b>24</b>	<b>25</b>	<b>49</b>
16	Тема 2.1. Постановка проблемной ситуации и поиск путей решения. Организация командной работы.	2	1	3
17	Тема 2.2. Введение в Arduino. Знакомство с микроконтроллером. Основные модули.	2	1	3
18	Тема 2.3. Основные конструкции языка. Переменные и константы.	3	2	5
19	Тема 2.4. Сборка простых схем. Маячок, светильник с управляемой яркостью.	2	1	3
20	Тема 2.5. Сборка простых схем, data scouting. Мерзкое пианино с реальными нотами	1	1	2
21	Тема 2.6. Основные конструкции языка. Условные операторы if..else. Циклы for и while.	3	2	5
22	Тема 2.7. Сборка и программирование мини-игры “Кнопочные ковбой”	1	1	2
23	Тема 2.8. Подключение LCD-экрана. Сборка схемы “Тестер батареек”	2	2	4
24	Тема 2.9. Сборка схемы “RGB-подсветка”	2	1	3
25	Тема 2.10. Сборка робота “Заварщик чая”.	4	4	8
26	Тема 2.11. Работа над проектом: сбор собственного электронного устройства и его программирование.		7	7
27	Тема 2.12. Подготовка к публичной демонстрации и защите результатов кейса.	2	1	3
28	Защита проектов.Рефлексия.		1	1
<b>Раздел 3. Основы Web-проектирования (html, CSS)</b>				
29	<b>Стрессоустойчивость.</b>	<b>1</b>	<b>0</b>	<b>1</b>
30	Тема 1. Методы психорегуляции.	1	0	1
31	<b>Креативность.</b>	<b>0</b>	<b>1</b>	<b>1</b>
32	Тема 1. Развитие компонентов творческой личности, инженерно-технического мышления.	0	1	1

33	<b>Кейс 3. Создание сайта-визитки</b>	<b>16</b>	<b>25</b>	<b>41</b>
34	Тема 3.1. Постановка проблемной ситуации и поиск путей решения. Организация работы в командах.	2	1	3
35	Тема 3.2. Техническое задание для будущего сайта.	2	1	3
36	Тема 3.3. Язык гипертекстовой разметки html и его инструментарий. Структуры html-кода.	2	1	3
37	Тема 3.4. Теги: типы, правила применения, атрибуты.	2	1	3
38	Тема 3.5. Особенности текста в html.	2	1	3
39	Тема 3.6. Ссылки: типы, правила вложения, атрибуты.	1	1	2
40	Тема 3.7. Добавление изображений на web-страницу.	1	2	3
41	Тема 3.8. Создание списков на web-странице.	1	2	3
42	Тема 3.9. Создание таблиц.	1	3	4
43	Тема 3.10. Фреймы: создание, границы, изменение размеров.	1	2	3
44	Тема 3.11. Таблица каскадных стилей CSS. Изучение базового синтаксиса CSS.	1	3	4
45	Тема 3.14. Работа над проектом: подготовка материала и наполнение сайта.		6	6
46	Защита проектов.Рефлексия.		1	1
47	Работа над итоговым проектом		8	8
48	Защита итогового проекта		1	1
	<b>Итого</b>	<b>54</b>	<b>82</b>	<b>136</b>

## Углубленный модуль.

№	Наименование кейса, темы	Количество часов		
		Теория	Практика	Всего
<b>Раздел 1. Язык программирования Python. Raspberry Pi 3.</b>				
1	<b>Кейс 1. Создание элементов умного дома на базе микрокомпьютера Raspberry Pi 3.</b>	<b>13</b>	<b>20</b>	<b>33</b>
2	<b>Сплочение коллектива.</b>	<b>0</b>	<b>2</b>	<b>2</b>
3	Тема 1. Тренинговое занятие по сплочению группы.		2	2
4	Тема 1.1. Постановка проблемной ситуации и поиск путей решения.	1	2	3
5	Тема 1.2. Основные конструкции языка. Переменные.	2	1	3
6	Тема 1.3. Основные конструкции языка. Условные конструкции.	2	1	3
7	Тема 1.4. Основные конструкции языка. Циклы.	2	1	3
8	Тема 1.5. Основные конструкции языка. Методы и функции.	2	1	3
9	Тема 1.6. Объектно-ориентированное программирование.	2	1	3
10	Тема 1.7. Работа над проектом.		8	8
11	Тема 1.8. Подготовка к публичной демонстрации и защите результатов кейса.	2	2	4
12	Защита проектов. Рефлексия.		2	2
<b>Раздел 2. Язык программирования C#</b>				
13	<b>Кейс 2. Разработка собственного Windows-приложения на платформе Visual Studio</b>	<b>16</b>	<b>26</b>	<b>42</b>
14	<b>Scrum-метод.</b>	<b>1</b>	<b>0</b>	<b>1</b>
15	Тема 1. Scrum-метод управления проектами.	1	0	1

16	Тема 2.1. Постановка проблемной ситуации и поиск путей решения. Знакомство со средой программирования Visual Studio	2	1	3
17	Тема 2.2. Основные конструкции языка. Переменные.	3	3	6
18	Тема 2.3. Основные конструкции языка. Условные конструкции.	3	3	6
19	Тема 2.4. Основные конструкции языка. Циклы.	3	3	6
20	Тема 2.5. Основные конструкции языка. Методы, функции и классы.	3	3	6
21	Тема 2.6. Объектно-ориентированное программирование.	2	3	5
22	Тема 2.7. Работа над проектом.		8	8
23	Тема 2.8. Подготовка к публичной демонстрации и защите результатов кейса.		1	1
24	Защита проектов. Рефлексия.		1	1
<b>Раздел 3. Создание игровых приложений на платформе Unity 3D</b>				
25	<b>Scrum-метод.</b>	<b>1</b>	<b>0</b>	<b>1</b>
26	Тема 1. Scrum-метод управления проектами.	1	0	1
27	<b>Кейс 3. Разработка собственной игры для Windows и Android</b>	<b>21</b>	<b>22</b>	<b>43</b>
28	Тема 3.1. Постановка проблемной ситуации и поиск путей решения. Знакомство с интерфейсом платформы Unity 3D	3	1	4
29	Тема 3.2. Основные компоненты. Сцена, объект, спрайт, скрипт.	4	3	7
30	<b>Стрессоустойчивость.</b>	<b>0</b>	<b>1</b>	<b>1</b>
31	Тема 1. Экспресс-методы по преодолению психологической напряженности. Формирование позитивной установки.	0	1	1

32	Тема 3.3. Основы анимации.	4	3	7
33	Тема 3.4. 2D игра. Создание персонажа, передвижение, прыжки.	3	3	6
34	Тема 3.5. 2D игра. Создание врагов, смена уровней.	5	2	7
35	Тема 3.6. UI интерфейс. Основные компоненты, методы создания и масштабирования.	2	2	4
36	<b>Креативность.</b>	<b>0</b>	<b>2</b>	<b>2</b>
37	Тема 1. Приёмы активизации мыслительной деятельности.	0	2	2
38	Тема 3.7. Работа над проектом.		6	6
39	<b>Эмоциональный интеллект.</b>	<b>1</b>	<b>1</b>	<b>2</b>
40	Тема 1. Эмоциональный интеллект. Эмпатические способности.	1	1	2
41	Тема 3.8. Подготовка к публичной демонстрации и защите результатов кейса.		1	1
42	Защита проектов. Рефлексия.		1	1
43	Работа над итоговым проектом.		8	8
44	Защита итогового проекта		1	1
	<b>Итого</b>	<b>53</b>	<b>83</b>	<b>136</b>

## СОДЕРЖАНИЕ КУРСА «IT-КВАНТУМ»

### Базовый модуль.

#### Технические навыки (hard skills).

#### Раздел № 1 «Визуальное (блочное) программирование».

Данный раздел имеет прикладную направленность. В процессе работы по данному кейсу учащиеся познакомятся с программированием в среде Scratch. Они изучат основы блочного (визуального) программирования.

В результате учащиеся, работая в команде, должны будут создать свою игру или викторину и провести их презентацию.

*Учащиеся должны знать:*

- понятия «исполнитель» (спрайт), «алгоритм», «переменная», «условные операторы», «циклы», «функция», «подпрограмма»;
- правила составления программ в среде программирования Scratch.

*Учащиеся должны уметь:*

- генерировать идеи;
- слушать и слышать собеседника;
- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- умение работать в команде;
- создавать игру\викторину на Scratch;
- пользоваться инструментами для совместной работы над проектом на сайте <https://scratch.mit.edu/>;
- работать с программами по созданию презентаций (MS PowerPoint, prezi.com);
- объективно оценивать результаты своей работы.

*Формы занятий, используемые при изучении данного кейса:*



- лекционная;
- групповая (командная) работа;
- групповые консультации;
- защита проектов.

**Тема 1.1.** Основные определения по теме «Алгоритмизация и программирование». Знакомство со средой программирования Scratch.

*Теория.* Понятия «алгоритм», «исполнитель», «программа», «команда», «программирование», «язык программирования». Изучение основных понятий среды разработки Scratch: «скрипт», «спрайт», «костюм», «сцена».

*Практика.* Знакомство с интерфейсом и главным меню сайта <https://scratch.mit.edu/>, окном создания программ. Группы блоков «События», «Движения», «Контроль» и «Внешность».

*Форма подведения итогов:* Публикация учебного проекта.

**Тема 1.2.** Блоки команд «Переменные», «Считывание», «Звук» и «Операторы» на примере создания простой игры.

*Теория.* Блоки команд «Считывание», «Звук» и «Операторы». Переменные.

*Практика.* Создание простой игры-погони с использованием переменных.

*Форма подведения итогов:* публикация учебного проекта.

**Тема 1.3.** Постановка проблемной ситуации, поиск путей решения. Организация командной работы.

*Теория.* Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения конечного результата. Деление на проектные команды, выбор темы проекта, определение цели и задач, распределение ролей.

*Практика.* Использование инструментов для совместной работы (студия, рюкзак, ремикс) над проектом на сайте <https://scratch.mit.edu/>

#### **Тема 1.4. Работа над проектом**

*Практика.* Создание командного проекта игры\викторины при помощи широких возможностей Scratch.

**Тема 1.5.** Подготовка к публичной демонстрации и защите результатов кейса.

*Теория.* Принципы успешной презентации проекта. Основы ораторского искусства.

*Практика.* Подготовка речи выступления и структуры презентации по итогам работы над кейсом. Создание презентации. (MS Power Point, prezi.com).

*Форма подведения итогов:* защита проектов.

#### **Раздел № 2 «Основы микроэлектроники».**

Данный раздел имеет инженерно-техническую направленность. В процессе работы по данному кейсу учащиеся овладеют основными навыками работы с аппаратной платформой Arduino и совместимыми электронными компонентами, изучат среду разработки Arduino IDE и научатся составлять программные коды на языке программирования C++.

В результате учащиеся должны собрать интеллектуальную кормушку для рыб, запрограммировать её и провести презентацию.

*Учащиеся должны знать:*

- основные понятия микроэлектроники;
- запись арифметических операций на языке C++;
- правила записи и применения логических и условных выражений, операторов цикла;
- технику безопасности при работе с электроникой.

*Учащиеся должны уметь:*

- генерировать идеи;
- слушать и слышать собеседника;

- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- комбинировать, видоизменять и улучшать идеи;
- грамотно письменно формулировать свои мысли;
- работать в команде;
- устанавливать и работать в среде разработки Arduino IDE;
- программировать микроконтроллерные платформы на языке C++;
- подключать электронные компоненты к микроконтроллерной платформе Arduino;
- получать и обрабатывать показания цифровых и аналоговых датчиков;
- управлять работой сервопривода;
- собирать конструкции с использованием винтовых и невинтовых соединений;
- работать с программами по созданию презентаций (MS PowerPoint, prezi.com);
- объективно оценивать результаты своей работы.

*Формы занятий*, используемые при изучении данного кейса:

- лекционная;
- групповая (командная) работа;
- групповые консультации;
- защита проектов.

**Тема 2.1.** Постановка проблемной ситуации и поиск путей решения.

Организация командной работы.

*Теория.* Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения идеального конечного продукта. Распределение учащихся по командам. Распределение задач

между членами команды.

**Тема 2.2.** Сборка интеллектуальной кормушки для рыб на основе микроконтроллера Arduino Uno.

*Теория.* Знакомство с имеющимися деталями дозатора и корпуса, аппаратной платформой и электронными компонентами. Измерение расстояния. Расчет объема геометрической фигуры.

*Практика.* Сборка конструкций с использованием винтовых и невинтовых соединений. Сборка дозатора корма и подключение электронных компонентов к микроконтроллерной платформе Arduino.

*Форма подведения итогов:* защита проектов.

**Тема 2.3.** Программа для управления интеллектуальной кормушкой для рыб на языке программирования C++.

*Теория.* Составление алгоритма программы и его формализация в виде блок-схемы. Среда разработки Arduino IDE. Написание и отладка программного кода.

*Практика.* Разработка алгоритма работы дозатора. Программирование микроконтроллерных платформ на языке C++ в Arduino IDE. Применение инфракрасных датчиков для определения расстояния. Получение и обработка показаний цифровых и аналоговых датчиков. Использование приводов с отрицательной обратной связью. Тестирование работы дозатора в различных режимах работы.

*Форма подведения итогов:* защита проектов.

**Тема 2.4.** Подготовка к публичной демонстрации и защите результатов кейса

*Теория.* Принципы успешной презентации проекта. Основы ораторского искусства.

*Практика.* Подготовка речи выступления и структуры презентации по итогам работы над кейсом. Создание презентации. (MS Power Point, prezi.com).

*Форма подведения итогов: защита проектов.*

### **Раздел № 3 «Основы Web-проектирования (html, CSS)».**

Данный раздел имеет прикладную направленность. В процессе работы по данному кейсу учащиеся овладеют навыками создания сайта на языке гипертекстовой разметки html с использованием CSS. Также повторяют принципы работы с графическими редакторами.

В результате учащиеся в команде должны спроектировать и создать сайт и провести его презентацию.

*Учащиеся должны знать:*

- инструментарий html;
- теги верхнего уровня и заголовка документа;
- различные виды тегов (блочные, строчные), правила их применения;
- особенности представления и оформления текстовой информации в html;
- теги для относительных и абсолютных ссылок, атрибуты ссылок и якоря;
- виды списков и соответствующие им теги: маркированный, нумерованный;
- теги для создания таблиц на web-странице;
- понятие «фрейм», достоинства и недостатки использования фреймов;
- понятие «каскадные таблицы стилей» (CSS), преимущества их применения;
- значения стилевых свойств.

*Учащиеся должны уметь:*

- генерировать идеи;
- слушать и слышать собеседника;

- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- комбинировать, видоизменять и улучшать идеи;
- грамотно письменно формулировать свои мысли;
- работать в команде;
- разрабатывать техническое задание проекта;
- создавать простую web-страницу при помощи программ «Блокнот», «Notepad++»;
- применять блочные, строчные теги;
- использовать атрибуты тегов;
- правильно оформлять ссылки в html коде;
- правильно добавить изображение на web-страницу: отредактировать в графическом редакторе; использовать тег <img> и его атрибуты;
- оформлять списки (маркированные, нумерованные);
- создавать таблицы;
- создавать фреймы с помощью тега <frameset>;
- добавлять стили на страницу разными способами и правильно применять CSS;
- работать с программами по созданию презентаций (MS PowerPoint, prezi.com);
- объективно оценивать результаты своей работы.

*Формы занятий*, используемые при изучении данного кейса:

- лекционная;
- групповая (командная) работа;
- групповые консультации;
- защита проектов.

**Тема 3.1.** Постановка проблемной ситуации и поиск путей решения.

Организация работы в командах.

*Теория.* Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения конечного результата. Обзор программных средств (HTML-редакторы, визуальные редакторы формата WYSIWYG) и интернет ресурсов для создания сайтов. Этапы проектирования сайтов. Распределение учащихся по командам.

**Тема 3.2.** Техническое задание для будущего сайта.

*Теория.* Формирование будущей структуры, визуальное оформление сайта. План предстоящих работ. Распределение задач между членами команды.

*Форма подведения итогов:* согласование технического задания.

**Тема 3.3.** Язык гипертекстовой разметки html и его инструментарий. Структуры html-кода.

*Теория.* Язык html и его инструментарий: блокнот (Notepad++) для написания кода, браузеры для отображения страниц. Структура html-кода.

*Практика.* Создание первой страницы, разбор строк html кода: теги верхнего уровня и заголовка документа.

**Тема 3.4.** Теги: типы, правила применения, атрибуты.

*Теория.* Различные виды тегов, правила их применения.

*Практика.* Применение блочных и строчных тегов. Использование атрибутов тегов (цвет, размер, адрес) при написании кода.

**Тема 3.5.** Особенности текста в html.

*Теория.* Особенности представления и оформления текстовой информации в html.

*Практика.* Представление текста на web-странице при помощи html.

**Тема 3.6.** Ссылки: типы, правила вложения, атрибуты.

*Теория.* Теги для относительных и абсолютных ссылок, правила вложения для тега <a>, атрибуты ссылок и якоря.

*Практика.* Добавление ссылок на web-страницы.

**Тема 3.7.** Добавление изображений на web-страницу.

*Теория.* Правила добавления изображений на web-страницу.

*Практика.* Работа с графическим редактором. Тег `<img>` и его атрибуты.

**Тема 3.8.** Создание списков на web-странице.

*Теория.* Виды списков и соответствующие им теги.

*Практика.* Маркированные и нумерованные списки в html.

**Тема 3.9.** Создание таблиц.

*Теория.* Таблицы на web-странице.

*Практика.* Теги `<table>`, `<td>`, `<tr>` для создания таблиц на web-странице.

**Тема 3.10.** Фреймы: создание, границы, изменение размеров.

*Теория.* Определение понятия «фрейм». Достоинства и недостатки использования фреймов.

*Практика.* Создание фреймов с помощью тега `<frameset>`.

**Тема 3.11.** Таблица каскадных стилей CSS. Изучение базового синтаксиса CSS.

*Теория.* Понятие «каскадные таблицы стилей» (CSS), преимущества их применения. Правила применения CSS.

*Практика.* Способы добавления стилей на web-страницу. Значения стилевых свойств (размер, цвет и пр.).

**Тема 3.12.** Работа над проектом: подготовка материала и наполнение сайта.

*Практика.* Сбор и редактирование необходимой информации (фото, текст) для наполнения сайта. Размещение подготовленной информации на созданном сайте. Редактирование сайта с помощью CSS.

*Форма подведения итогов:* защита проектов.

**Тема 3.13.** Подготовка к презентации результата командной работы.



*Теория.* Принципы успешной презентации проекта. Основы ораторского искусства.

*Практика.* Подготовка речи выступления и структуры презентации по итогам работы над кейсом. Создание презентации. (MS Power Point, prezi.com).

*Форма подведения итогов:* защита проектов.

### **Базовый модуль.**

#### **Социальные навыки (soft skills).**

#### **Командообразование.**

В процессе реализации данной темы, учащиеся получают позитивные навыки работы в команде, повысят лояльность к команде, улучшат коммуникации внутри команды, научатся распределению обязанностей и делегированию полномочий в команде, а также получат навыки эффективной работы в команде.

#### **Тема 1.** Командообразование и методы групповой работы.

*Теория.* Мини-лекция: «Этапы формирования команды». Рассматриваются базовые модели и практические навыки проведения групповой работы.

#### **Тема 2.** Тренинговое занятие по командообразованию.

*Практика.* Деловые игры, тренинговые упражнения.

*Форма подведения итогов:* рефлексия.

### **Основы ведения проектной деятельности.**

В ходе реализации данной темы, учащиеся получают навыки практического применения проектной деятельности.

#### **Тема 1.** Основы ведения проектной деятельности.

*Теория.* Что такое проект. Виды проектов. Этапы проектной деятельности. Что такое проблема. Понятие о гипотезе. Выбор идеи

проекта. Постановка целей и задач. Определение формы взаимодействия при работе над проектом. Определение предмета и методов исследования в работе над проектом. Составление плана работы над проектами. Подготовка к защите. Защита проектов.

*Форма подведения итогов:* рефлексия.

### **Тайм-менеджмент.**

В процессе реализации темы, учащиеся сформируют навыки планирования, распределения и расходования времени, освоят техники постановки целей, получат умения распределять приоритеты, пользоваться инструментами планирования и грамотным их применением.

**Тема 1.** Тайм-менеджмент. Технология управления временем.

*Теория.* Мини-лекция «Тайм-менеджмент». Теория управления временем Франклина.

*Форма подведения итогов:* рефлексия.

### **Стрессоустойчивость.**

Данная тема формирует психологическую готовность учащихся к участию в ответственных мероприятиях.

**Тема 1.** Методы психорегуляции.

*Теория.* Метод отвлечения. Метод самоприказа. Метод регуляции дыхания. Метод сосредоточения внимания. Метод мышечного расслабления. Создание комфортного психоэмоционального состояния.

*Практика.* Тренинговое занятие. Практическое применение рассмотренных методов психорегуляции.

*Форма подведения итогов:* рефлексия.

### **Креативность.**

В процессе реализации данной темы, учащиеся сформируют навыки

и умения управления креативным процессом; раздел способствует развитию гибкости и оригинальности мышления, развитию воображения, нестандартного и творческого мышления.

**Тема 1.** Развитие компонентов творческой личности, инженерно-технического мышления.

*Практика.* Практические упражнения по развитию креативности, творческого мышления, инженерно-технического мышления.

*Форма подведения итогов:* рефлексия.

**Углубленный модуль.**

**Технические навыки (hard skills).**

**Раздел 1. «Электронные устройства на основе одноплатного компьютера Raspberry Pi 3».**

Данный раздел имеет прикладную направленность. В процессе работы по данному кейсу учащиеся овладеют основными навыками работы с одноплатным компьютером Raspberry Pi 3 и совместимыми электронными компонентами. Изучат среду разработки Python IDE и научатся составлять программные коды на языке программирования Python 3.

В результате учащиеся должны собрать какое-либо свое электронное устройство, запрограммировать его и провести презентацию.

*Учащиеся должны знать:*

- основные понятия микроэлектроники;
- строение платы Raspberry Pi 3 и области её применения;
- типы переменных, их запись на языке Python 3;
- различные операторы (присваивания, сравнения, арифметические и логические) на языке Python 3;
- условные инструкции if-else и if-elif-else на языке Python 3;
- правила записи и применение функций на языке Python 3;

- разные виды циклов на языке Python 3: for, while;
- списки (массивы), индексы и срезы, кортежи, словари, множества в языке Python 3;
- стандартные библиотеки для Python 3;
- технику безопасности при работе с электроникой.

*Учащиеся должны уметь:*

- генерировать идеи;
- слушать и слышать собеседника;
- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- комбинировать, видоизменять и улучшать идеи;
- грамотно письменно формулировать свои мысли;
- работать в команде;
- настраивать и использовать среду разработки Python IDE;
- программировать Raspberry Pi 3 на языке Python 3;
- подключать различные компоненты к микроконтроллерной платформе Raspberry Pi 3;
- получать и обрабатывать показания цифровых и аналоговых датчиков;
- работать с программами по созданию презентаций (MS PowerPoint, prezi.com);
- объективно оценивать результаты своей работы.

*Формы занятий, используемые при изучении данного кейса:*

- лекционная;
- групповая (командная) работа;
- групповые консультации;
- защита проектов.

**Тема 1.1.** Постановка проблемной ситуации и поиск путей решения.

Знакомство с аппаратной платформой Raspberry Pi.

Подключение и настройка платформы.

*Теория.* Устройство и назначения основных частей микроконтроллерной платформы Raspberry Pi и областей ее применения.

Операционные системы для Raspberry Pi.

*Практика.* Подключение периферийных устройств (монитор, клавиатура, компьютерная мышь) к Raspberry Pi. Установка и настройка операционной системы Raspbian.

**Тема 1.2.** Выводы общего назначения (GPIO). Подключение и работа с электронными компонентами.

*Теория.* Схема выводов Raspberry Pi. Возможности GPIO. Работа с GPIO на Python.

*Практика.* Подключение простых электронных компонентов и датчиков к Raspberry Pi. Сбор и программирование электронных устройств.

**Тема 1.3.** Язык программирования Python 3 для Raspberry Pi 3.

*Теория.* Переменные в языке Python 3. Операторы (присваивания, сравнения, арифметические и логические) в языке Python 3. Условные инструкции if-else и if-elif-else в языке Python 3. Функции в языке Python 3. Циклы for, while в языке Python 3. Списки (массивы), индексы и срезы, кортежи, словари, множества в языке Python 3.

*Практика.* Сбор и программирование электронных устройств.

**Тема 1.4.** Работа над проектом: сбор собственного электронного устройства и его программирование.

*Теория.* Составление технического задания проекта.

*Практика.* Проектирование и сбор электронного устройства. Отладка программного кода для собранного устройства.

*Форма подведения итогов:* защита проектов.

**Тема 1.5** Подготовка к публичной демонстрации и защите

результатов кейса.

*Теория.* Принципы успешной презентации проекта. Основы ораторского искусства.

*Практика.* Подготовка речи выступления и структуры презентации по итогам работы над кейсом. Создание презентации. (MS Power Point, prezi.com).

*Форма подведения итогов:* защита проектов.

## **Раздел 2. «Язык программирования C#».**

Данный раздел имеет прикладную направленность. В процессе работы по данному кейсу учащиеся овладеют основными навыками работы с языком программирования C# и платформой Visual Studio.

В результате учащиеся должны собрать какое-либо свое электронное устройство, запрограммировать его и провести презентацию.

*Учащиеся должны знать:*

– основные конструкции си-подобных языков программирования;

– типы переменных, их запись на языке C#;

– различные операторы (присваивания, сравнения, арифметические и логические) на языке C#;

– условные инструкции if...else и switch на языке C#;

– правила записи и применение функций на языке C#;

– разные виды циклов на языке C#: for, do...while, while;

– понятие «объект», правила записи объектов и их свойств на языке C#;

– технику безопасности при работе с компьютером.

*Учащиеся должны уметь:*

– генерировать идеи;

– слушать и слышать собеседника;

- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- комбинировать, видоизменять и улучшать идеи;
- грамотно письменно формулировать свои мысли;
- работать в команде;
- настраивать и использовать среду разработки Visual Studio;
- работать с программами по созданию презентаций (MS PowerPoint, prezi.com);
- объективно оценивать результаты своей работы.

*Формы занятий*, используемые при изучении данного кейса:

- лекционная;
- групповая (командная) работа;
- групповые консультации;
- защита проектов.

**Тема 2.1.** Постановка проблемной ситуации и поиск путей решения.

Организация командной работы.

*Теория.* Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения конечного результата. Деление на проектные команды, выбор темы проекта, определение цели и задач, распределение ролей.

**Тема 2.2.** Основные конструкции языка. Переменные.

*Теория.* Правила именования переменных. Типы данных.

*Практика.* Обзор типов данных, пример создания простых переменных.

**Тема 2.3.** Основные конструкции языка. Условные конструкции.

*Теория.* Условные конструкции if...else, switch...case.

*Практика.* Применение условных конструкций в калькуляторе.

**Тема 2.4.** Основные конструкции языка. Циклы.

*Теория.* Циклические конструкции языка программирования C#.

*Практика.* Заполнение массивов при помощи циклов do...while, for и foreach.

**Тема 2.5.** Основные конструкции языка. Методы, функции и классы.

*Теория.* Правила создания методов, написания собственных функций и классов

*Практика.* Создание своего типа данных, манипулирующего цифрами (продвинутый калькулятор).

**Тема 2.6.** Объектно-ориентированное программирование

*Теория.* ООП. Инкапсуляция, наследование и полиморфизм. Термины и законы.

*Практика.* Применение ООП на практике.

**Тема 2.7.** Работа над проектом.

*Теория.* Составление технического задания проекта.

*Практика.* Проектирование и создание собственного приложения, использующего все преимущества ООП. Отладка программного кода и создание дизайна.

*Форма подведения итогов:* защита проектов.

**Тема 2.8.** Подготовка к публичной демонстрации и защите результатов кейса.

*Теория.* Принципы успешной презентации проекта. Основы ораторского искусства.

*Практика.* Подготовка речи выступления и структуры презентации по итогам работы над кейсом. Создание презентации. (MS Power Point, prezi.com).

*Форма подведения итогов:* защита проектов.

**Раздел 3.** «Создание игровых приложение на движке Unity».

Данный кейс имеет прикладную направленность. В процессе работы



по данному кейсу учащиеся овладеют навыками создания игровых приложений на базе движка Unity и языка программирования C#.

В результате учащиеся в команде должны спроектировать и создать игру и провести его презентацию.

*Учащиеся должны знать:*

- инструментарий unity;
- создавать различные объекты на сцене;
- пользоваться методами отладки;
- основы создания текстур;
- методы создания скриптов для объектов;
- способы разработки своих анимаций;
- основы 2D-разработки;
- основы 3D-разработки;
- методы создания своих спрайтов

*Учащиеся должны уметь:*

- генерировать идеи;
- слушать и слышать собеседника;
- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- комбинировать, видоизменять и улучшать идеи;
- грамотно письменно формулировать свои мысли;
- работать в команде;
- разрабатывать техническое задание проекта;
- создавать простую игру на платформе Unity 3D;
- создавать анимации объектов;
- писать свои скрипты;
- правильно оформлять сцены в игре;
- правильно добавить свои анимации;

- оформлять визуальный контент приложения
- создавать объекты;
- создавать текстуры;
- добавлять ассеты;
- работать с программами по созданию презентаций (MS PowerPoint, prezi.com);
- объективно оценивать результаты своей работы.

*Формы занятий*, используемые при изучении данного кейса:

- лекционная;
- групповая (командная) работа;
- групповые консультации;
- защита проектов.

**Тема 3.1.** Постановка проблемной ситуации и поиск путей решения.

Знакомство с интерфейсом платформы Unity3D.

*Теория.* Анализ проблемной ситуации, генерация и обсуждение методов ее решения и возможности достижения конечного результата. Обзор программных средств и интернет ресурсов для создания игр. Этапы проектирования сайтов. Распределение учащихся по командам.

**Тема 3.2.** Основные компоненты. Сцена, объект, спрайт, скрипт.

*Теория.* Знакомство с основными элементами интерфейса платформы Unity3D.

*Практика.* Создание сцены и добавление на неё различных элементов.

**Тема 3.3.** Основы анимации.

*Теория.* Анимация и связь между различными состояниями объекта.

*Практика.* Создание первой анимации главного героя.

**Тема 3.4.** 2D игра. Создание персонажа. Передвижение, прыжки.

*Теория.* Скрипты – как основа передвижения персонажей.

*Практика.* Написание скрипта передвижения для главного героя.

### **Тема 3.5.** 2D игра. Создание врагов. Смена уровня

*Теория.* Триггеры в играх. Значение и необходимость использования триггеров.

*Практика.* Создание врагов. Зоны агрессии и триггеры перехода между уровнями.

**Тема 3.6.** UI интерфейс. Основные компоненты, методы создания и масштабирования.

*Теория.* Что такое UI и как правильно его создавать.

*Практика.* Создание главного меню игры.

### **Тема 3.7.** Добавление изображений на web-страницу.

*Теория.* Правила добавления изображений на web-страницу.

*Практика.* Работа с графическим редактором. Тег `<img>` и его атрибуты.

**Тема 3.8.** Работа над проектом: подготовка материала и создание игры.

*Практика.* Сбор и редактирование необходимой информации для создания игры.

*Форма подведения итогов:* защита проектов.

### **Тема 3.9.** Подготовка к презентации результата командной работы.

*Теория.* Принципы успешной презентации проекта. Основы ораторского искусства.

*Практика.* Подготовка речи выступления и структуры презентации по итогам работы над кейсом. Создание презентации. (MS Power Point, prezi.com).

*Форма подведения итогов:* защита проектов.

**Углубленный модуль.**

**Социальные навыки (soft компетенции).**

**Сплочение коллектива.**

В процессе реализации, учащиеся получают навыки сплочения группы и построения эффективного командного взаимодействия, формируют благоприятный психологический климат в группе, разовьют умение работать в команде, получают навыки анализа групповой работы, управления процессами коммуникаций в группе.

**Тема 1.** Тренинговое занятие по сплочению группы.

*Практика.* Тренинговые упражнения на сплочение группы.

Форма подведения итогов: рефлексия.

### **Scrum-метод.**

Учащиеся получают практические навыки ведения проектов с помощью использования Scrum-метода.

**Тема 1.** Scrum-метод управления проектами.

*Теория.* Основные определения в Scrum-методе. Концепция Scrum-методологии. Роли в Scrum. Ритуалы и артефакты Scrum.

*Практика.* Создание мини-проекта с использованием Scrum-метода.

Форма подведения итогов: рефлексия.

### **Тайм-менеджмент.**

Учащиеся сформируют навыки управления временем и достижению максимально положительного результата.

**Тема 1.** Тайм-менеджмент. Упражнения по управлению временем.

*Практика.* Деловые игры, практические задания, выполнение индивидуальных и групповых упражнений по управлению временем.

Форма подведения итогов: рефлексия.

### **Стрессоустойчивость.**

Данная тема формирует психологическую готовность учащихся к участию в ответственных мероприятиях.

**Тема 1.** Экспресс-методы по преодолению психологической напряженности. Формирование позитивной установки.

*Практика.* Тренинговые упражнения на преодоление психологической напряженности.

*Форма подведения итогов:* рефлексия.

### **Креативность.**

В ходе реализации данной темы, учащиеся научатся основным приемам активизации мыслительной деятельности и получают рекомендации для самостоятельного развития творческих компонентов личности.

**Тема 1.** Приёмы активизации мыслительной деятельности.

*Практика.* Применение метода фокальных объектов. Упражнения для активизации мыслительной деятельности и креативного мышления.

*Форма подведения итогов:* рефлексия.

### **Эмоциональный интеллект.**

Данная тема способствует созданию условий для личностного роста учащихся, развитию и совершенствованию способности понимать собственный эмоциональный мир, способности к сочувствию и сопереживанию окружающим.

**Тема 1.** Эмоциональный интеллект. Эмпатические способности.

*Теория.* Интерактивная лекция «Эмоциональный интеллект». Модель способностей. Смешанная модель.

*Практика.* Тренинг по развитию эмпатических способностей.

*Форма подведения итогов:* рефлексия.

## МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММЫ «ИТ-КВАНТУМ»

Тема кейса	Форма занятий	Приемы и методы организации образовательного процесса	Дидактический материал. Электронные источники	Техническое оснащение и расходный материал	Форма подведения итогов
<i>Базовый модуль</i>					
Кейс 1. Визуальное (блочное) программирование.	Комбинированная	Кейс метод. Метод проектов.	<ul style="list-style-type: none"> <li>● <a href="https://scratch.mit.edu/">https://scratch.mit.edu/</a> - официальный сайт среды разработки Scratch с руководствами и примерами проектов;</li> <li>● <a href="https://csfirst.withgoogle.com/s/en/home">https://csfirst.withgoogle.com/s/en/home</a> — англоязычный сайт с методическими разработками для обучения детей работе в Scratch.</li> </ul>	<ul style="list-style-type: none"> <li>● Компьютеры (ноутбуки) с монитором, клавиатурой и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7) пакет офисных программ MS Office;</li> <li>● Презентационное оборудование.</li> </ul>	Защита проектов

<p>Кейс 2. Электронные устройства на основе микропроцессора Arduino</p>	<p>Комбинированная</p>	<p>Кейс метод. Метод проектов.</p>	<ul style="list-style-type: none"> <li>● <a href="http://wiki.amperka.ru/">http://wiki.amperka.ru/</a> - сайт Амперка, где содержатся материалы, которые помогут освоить Arduino, основы схемотехники и программирования</li> <li>● <a href="https://www.arduino.cc/">https://www.arduino.cc/</a> - официальный сайт Arduino</li> <li>● <a href="https://arduinomaster.ru/">https://arduinomaster.ru/</a> - сайт с инструкциями по работе с микроконтроллерами Arduino</li> <li>● <a href="https://all-arduino.ru/">https://all-arduino.ru/</a> - сайт с разными уроками, схемами подключения, библиотеками Arduino.</li> </ul>	<ul style="list-style-type: none"> <li>● Компьютеры (ноутбуки) с монитором, клавиатурой и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7); среда разработки Arduino IDE (версия не ниже 1.6.10); пакет офисных программ MS Office;</li> <li>● Презентационное оборудование.</li> <li>● Комплекты деталей для кейса «Интеллектуальная кормушка для рыб» версия 1.0;</li> <li>● Плоскогубцы;</li> <li>● Отвертки крестовые;</li> <li>● Инструменты режущие (ножницы, кусачки);</li> <li>● Большие картонные коробки (30 x 20 см)</li> </ul>	<p>Защита проектов</p>
---	------------------------	--	---	---	------------------------

				<ul style="list-style-type: none"> <li>или аквариум с прямыми стенками;</li> <li>• Корм для рыб в виде мелких и крупных гранул;</li> <li>• Емкости с водой, глубиной не менее 15 сантиметров.</li> </ul>	
Кейс 3. Основы Web-проектирования (html, CSS).	Комбинированная	Кейс метод. Метод проектов.	<ul style="list-style-type: none"> <li>• <a href="http://www.aklychin.ru/">http://www.aklychin.ru/</a> - блог «Основы сайтостроения и веб дизайна»;</li> <li>• <a href="http://htmlbook.ru/samhtml">http://htmlbook.ru/samhtml</a> - сайт по азам создания сайтов, включающий самоучитель и справочник по html и CSS;</li> <li>• <a href="http://www.webremeslo.ru/index.html">http://www.webremeslo.ru/index.html</a> - сайт, на котором содержится электронный учебник по курсу html и учебник по CSS.</li> </ul>	<ul style="list-style-type: none"> <li>• Компьютеры (ноутбуки) с монитором, клавиатурой и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7); графический редактор (Adobe Photoshop), пакет офисных программ MS Office;</li> <li>• Презентационное оборудование.</li> </ul>	Защита проектов
<b>Углубленный модуль</b>					
Кейс 1. Язык программирования	Комбинированная	Кейс метод. Метод проектов.	<ul style="list-style-type: none"> <li>• <a href="http://wiki.amperka.ru/">http://wiki.amperka.ru/</a> - сайт Амперка, где содержатся</li> </ul>	<ul style="list-style-type: none"> <li>• Компьютеры (ноутбуки) с</li> </ul>	Защита проектов



Python. Raspberry Pi 3.

материалы, которые помогут освоить Arduino, основы схемотехники и

программирования;

- <https://www.raspberrypi.org/> - официальный сайт с документацией по одноплатному компьютеру Raspberry Pi;
- <http://raspberrypi.ru/> - сайт с уроками и справочной документацией по Raspberry Pi;
- <https://www.python.org/> - официальный сайт языка программирования Python;
- <https://pythonworld.ru/> - сайт с уроками по Python 3 на русском языке.

монитором, клавиатурой и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7), пакет офисных программ MS Office;

- Презентационное оборудование;
- периферийные устройства: монитор, клавиатура, компьютерная мышь;
- Наборы «Малина», в состав которых входят: одноплатный компьютер Raspberry Pi 3 Model B+,

				<p>microSD-карта с операционной системой, Raspbian Linux, плата GPIO Cloud, корпус, блок питания с USB-выходом, кабель USB (A — Micro USB), кабель HDMI, патч-корд витой пары.</p>	
<p>Кейс 2. Язык программирования C#</p>	<p>Комбинированная</p>	<p>Кейс метод. Метод проектов.</p>	<ul style="list-style-type: none"> <li>● <a href="https://www.microsoft.com/ru-ru">https://www.microsoft.com/ru-ru</a> - Официальный сайт корпорации microsoft</li> <li>● <a href="https://professorweb.ru/">https://professorweb.ru/</a> - Уроки по C# и платформе .NET Framework</li> <li>● <a href="https://metanit.com/">https://metanit.com/</a> - Данный сайт посвящен различным языкам и технологиям программирования, компьютерам, мобильным платформам и ИТ-технологиям. Здесь будут выкладываться различные руководства и учебные материалы, статьи и примеры.</li> </ul>	<ul style="list-style-type: none"> <li>● Компьютеры (ноутбуки) с монитором, клавиатурой и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7), пакет офисных программ MS Office;</li> <li>● Презентационное оборудование;</li> <li>● VisualStudio</li> </ul>	<p>Защита проектов</p>

<p>Кейс 3. Создание игровых приложений на платформе Unity 3D</p>	<p>Комбинированная</p>	<p>Кейс метод. Метод проектов.</p>	<ul style="list-style-type: none"> <li>● <a href="https://www.microsoft.com/ru-ru">https://www.microsoft.com/ru-ru</a> - Официальный сайт корпорации microsoft</li> <li>● <a href="https://professorweb.ru/">https://professorweb.ru/</a> - Уроки по С# и платформе .NET Framework</li> <li>● <a href="https://metanit.com/">https://metanit.com/</a> - Данный сайт посвящен различным языкам и технологиям программирования, компьютерам, мобильным платформам и ИТ-технологиям. Здесь будут выкладываться различные руководства и учебные материалы, статьи и примеры.</li> <li>● <a href="https://unity.com/ru">https://unity.com/ru</a> - Официальный сайт с уроками и методичками</li> </ul>	<ul style="list-style-type: none"> <li>● Компьютеры (ноутбуки) с монитором, клавиатурой и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: операционная система Windows (версия не ниже 7), пакет офисных программ MS Office;</li> <li>● Презентационное оборудование;</li> <li>● VisualStudio</li> <li>● Unity Hub</li> <li>● Unity 3D</li> </ul>	<p>Защита проектов</p>
--	------------------------	--	---	---	------------------------

## **СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ**

### **использованных при написании программы:**

1. Войков В. «АЙТИ Квантум тулкит». - Базовая серия «Методический инструментарий тьютора», 2017.

### **рекомендованных обучающимся:**

1. Изучаем Python, 4-е издание – Марк Лутц;
2. Программирование на Python 3. Подробное руководство – Марк Саммерфилд;
3. PHP 7 – Дмитрий Котеров, Игорь Симдянов;
4. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 – Робин Никсон;
5. PHP. Быстрый старт – Каллум Хопкинс;
6. Занимательная электроника – Ревич Юрий;
7. Делаем сенсоры. Проекты сенсорных устройств на базе Arduino и Raspberry Pi – Торо Карвинен, Киммо Карвинен, Вилле Валтокари;
8. Компьютерные сети. Принципы, технологии, протоколы – Виктор Олифер, Наталия Олифер;
9. <https://stepik.org/> – ресурс для самообразования, образовательная платформа и конструктор онлайн-курсов;
10. <https://scratch.mit.edu/> – официальный сайт среды разработки Scratch с руководствами и примерами проектов;
11. <http://appinventor.mit.edu/explore/> – официальный сайт MIT App Inventor;
12. <http://wiki.amperka.ru/> – сайт Амперка, где содержатся материалы, которые помогут освоить Arduino, основы схемотехники и программирования;
13. <https://www.arduino.cc/> – официальный сайт Arduino;

14. <https://arduinomaster.ru/> – сайт с инструкциями по работе с микроконтроллерами Arduino;
15. <https://all-arduino.ru/> – сайт с разными уроками, схемами подключения, библиотеками Arduino;
16. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Guide> – руководство по изучению Java Script;
17. <http://htmlbook.ru/samhtml> – сайт по азам создания сайтов, включающий самоучитель и справочник по html и CSS;
18. <http://www.webremeslo.ru/index.html> – сайт, на котором содержится электронный учебник по курсу html и учебник по CSS;
19. <https://serveradmin.ru/> – сайт со справочным материалом по сетевому администрированию;
20. <https://www.virtualbox.org/> – официальный сайт VirtualBox – программного продукта виртуализации для операционных систем;
21. <https://ru.wordpress.org/> – официальный сайт CMS WordPress с документацией по системе;
22. <http://php.net/> – сайт с документацией по языку программирования PHP;
23. <https://www.mysql.com/> – официальный сайт с документацией по базам данных MySQL;
24. <http://www.lua.org/> – официальный сайт языка программирования Lua;
25. <https://coronalabs.com/> – официальный сайт фреймворка Corona SDK;  
<https://www.raspberrypi.org/> – официальный сайт с документацией по одноплатному компьютеру Raspberry Pi;
26. <http://raspberrypi.ru/> – сайт с уроками и справочной документацией по Raspberry Pi;
27. <https://www.python.org/> – официальный сайт языка программирования Python;
28. <https://pythonworld.ru/> – сайт с уроками по Python 3 на русском языке.

29.<https://www.centos.org/> – официальный сайт операционной системы CentOS, на котором можно скачать дистрибутив операционной системы;

30.<https://openvpn.net/> – официальный сайт OpenVPN.