

Министерство образования Самарской области
Государственное бюджетное образовательное учреждение средняя
общеобразовательная школа № 4 имени Героя Советского Союза Д.П.
Левина городского округа Сызрань Самарской области

Принята
на заседании
методического объединения
учителей естественно-научного
цикла
Протокол № 1 от 31.07.2024 г.

Утверждаю
Директор ГБОУ СОШ № 4
г.о.Сызрань
_____ С.Ю.Титова

Приказ от 31.07.2024 г. №856-ОД

**Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«IT-программирование»**

Направленность: техническая

Возраст обучающихся: 12-17 лет

Срок реализации: 1 год (108 часов)

Автор-составитель:

Левин М.Ю.,

педагог дополнительного образования

Сызрань, 2024 г.

Краткая аннотация

Дополнительная общеобразовательная общеразвивающая программа «IT-программирование» предназначена для учащихся в возрасте от 12 до 17 лет, проявляющих интерес к техническому творчеству. В результате обучения дети овладеют фундаментальными знаниями, умениями и навыками необходимыми для программирования на различных языках, познакомятся с основами логики и всеми базовыми конструкциями, разработают свои первые полноценные приложения, игры и сайты, а также познакомятся с системами, применяемыми в рамках проектирования и реализации интернета вещей и умного дома на основе микроконтроллеров.

Оглавление

Пояснительная записка	4
Учебно-тематический план	10
Содержание программы	11
Методическое обеспечение	23
Материально-техническое обеспечение.....	24
Список литературы.....	25
ПРИЛОЖЕНИЯ	27

Пояснительная записка

Информационные технологии с каждым годом все более прочно занимают передовые позиции в любых сферах человеческой деятельности. Объясняется это многими факторами, например, стремлением людей к эволюции, жадой упростить собственную жизнь или свой быт, познать окружающий мир, двигаться в ногу со временем и быть несравнимо более социальным существом, для которого нет ни расстояний, ни границ. Менее полувека назад электронно-лучевой прибор под названием «телевизор» был роскошью, а уже сегодня это обыденность. Технологии шагнули далеко вперед и не только они, но и сама человеческая мысль, а самое главное, что процесс этот не остановить. Сферу образования, целью которой является воспитание знаний, умений и навыков всесторонне развитой личности также необходимо совершенствовать и актуализировать. Ведь в наше время никому не интересно становиться каким-нибудь дворником и это подходит только лишь для родительской мотивации детей к плодотворной учебе при помощи запугивания, разумеется, они ни в коем случае не желают такого будущего своему ребенку. С детства многие связали свою жизнь с передовыми технологиями – дети нового поколения. Пусть оказываемое ими влияние спорно и не до конца изучено, но настоящим мейнстримом и квинтэссенцией всего выше сказанного оказывается человек, не выпускающий любимый электронный гаджет из рук и связывающий с технологиями не только свое свободное время, но и профессию, и даже всю свою жизнь. Разумеется, людям нужны таланты способные двигать всю индустрию и человечество вперед, но как их найти или создать? Пользоваться благом одно, а управлять и совершенствовать качественно иной уровень доступный не каждому, но желаемый многими в соответствии с популярностью и современной доступностью. Над разработками в области информационных технологий и программного обеспечения трудятся крупнейшие и богатейшие корпорации всего мира, а их специалисты занимающиеся программированием очень ценны и востребованы.

Общеразвивающая образовательная программа дополнительного образования детей «IT-программирование» имеет **техническую направленность**. Программа является модифицированной. Составлена на основе экспериментальной дополнительной общеразвивающей программы «Информационные технологии: от «умного» чайника до автоматизированного производства» АНО ДО «Детский технопарк «Кванториум», Булахов Н.Г., учебно-методического практикума «Основы алгоритмизации и программирования» Коврижных А.Ю..

Программа предназначена для обучающихся 12-17 лет.

Сроки реализации программы:

Программа рассчитана на 1 год обучения: 108 часов в год.

Новизна данной дополнительной образовательной программы заключается в том, что **по форме организации образовательного процесса она является модульной** и учитывает новейшие технологические уклады, а в образовательном процессе применяются различные элементы педагогических технологий, например, коллективное взаимообучение, игровые технологии и ориентированность на проектную деятельность.

Уникальность программы обусловлена использованием широкого спектра оборудования для приобретения практических навыков работы с ультрасовременными технологиями, такими как Internet of Things (IoT). Это технологическая концепция, согласно которой физические объекты и приборы оснащаются устройствами для обмена данными между собой и внешней средой. Считается, что такой подход открывает принципиально новые возможности в идентификации, измерении, сборе и обработке данных, невозможные без применения данной технологии.

Работа по программе заключается в использовании ноутбуков с современным программным обеспечением, что повышает мотивацию учащихся к обучению, при этом требуются знания практически из всех учебных дисциплин в особенности из математики, английского языка, естественных наук. Занятия опираются на интерес к систематизации знаний, умений и навыков в процессе изучения различных языковых конструкций. Изучение основ алгоритмизации и программирования дает школьникам фундаментальные и основополагающие знания в области IT.

Обучение по данной программе направлено на приобретение учащимися знаний и привлечение их к современным технологиям телекоммуникаций, программирования, работы с микропроцессорными системами.

Работа с различными языковыми конструкциями позволит школьникам развивать логическое и структурное мышление, комплексный подход при выполнении проектов и декомпозицию задач. Программирование это в первую очередь творчество, позволяющее с нуля и абсолютно самостоятельно реализовывать идеи.

Очень важным представляется тренировка работы в коллективе и развитие самостоятельного технического творчества и фантазии.

В ходе курса школьники изучают основы программирования, выполняя учебные проекты на компьютерах с возможностью использования современных микроконтроллерных плат, являющихся основой для построения простых систем автоматики и робототехники. Учащиеся программируют логику работы микропроцессорных устройств для выполнения периферийными устройствами

практических задач. Формируется общее представление и понимание логики работы операционных систем и их возможностей по управлению периферийными устройствами.

Актуальность данной программы обусловлена современной потребностью рынка в специалистах в области информационных технологий. Учитывается и междисциплинарность ИТ, что способствует интегрированию преподавания информатики, математики, физики, черчения, естественных наук с развитием инженерного мышления, через техническое творчество. Техническое творчество — мощный инструмент синтеза знаний, закладывающий прочные основы системного мышления. Предусмотрено приобретение навыков в области применения информационных технологий в робототехнике, виртуальной реальности, дизайне, геоинформационных системах, аэрокосмических технологиях и тд.

Педагогическая целесообразность программы состоит в том, что введение в дополнительное образование образовательной программы «ИТ-программирование» с использованием таких методов, как командная работа, поиск проблем и их практическое решение, анализ и обобщение опыта, подготовка исследовательских проектов и их защита, элементы соревнований и т.д., неизбежно изменит картину восприятия учащимися технических дисциплин, переводя их из разряда умозрительных в разряд прикладных. Применение детьми на практике теоретических знаний, полученных из области математики, физики, информатики, ведет к более глубокому пониманию основ, закрепляет полученные навыки, формируя образование в его наилучшем смысле. И с другой стороны, практическая работа с самым современным оборудованием данной области позволит учащимся в дальнейшем самостоятельно следовать тенденциям развития средств вычислительной техники, телекоммуникаций и веб-технологий. Таким образом, новое поколение теоретически окажется способным к совершению инновационного прорыва в современной науке и технике. Сформируется проектный подход и развивается командная работа юных «специалистов». Учениками приобретаются надпредметные компетенции: коммуникабельность, предсказательная аналитика и другие, что предоставит возможность в будущем стать успешными специалистами в любой области технологических разработок.

Цель – развитие интереса к информационным и телекоммуникационным технологиям, а также реализация творческих идей в области программирования в виде проектов различного уровня сложности.

Задачи:

Обучающие:

- ознакомить с методами программирования на языках, применяемых в современной вычислительной технике;
- дать первоначальные знания по работе в интегрированных средах разработки;
- навыкам конструирования сложных систем, управляемых микроконтроллерами и миникомпьютерами;
- сформировать общенаучные и технологические навыки программирования и проектирования.

Воспитывающие:

- воспитывать нравственные качества личности: настойчивость в достижении целей и продуктивно законченных результатов проектной деятельности, ответственность, дисциплинированность, трудолюбие;
- воспитывать коммуникативные качества;
- формировать творческое отношение к выполняемой работе;
- мотивировать учащихся к созданию собственных программных реализаций и электронных устройств.

Развивающие:

- развивать образное, техническое мышление;
- развивать умение работать в команде по предложенным инструкциям;
- развивать творческое мышление и воображение, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска;
- развивать навыки программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развивать внимательность, аккуратность и изобретательность при работе с техническими устройствами, создании электронных устройств и выполнении учебных проектов.

Основной отличительной особенностью данной программы является вариативность содержания: в зависимости от предпочтений обучающегося в проектной деятельности возможен выбор образовательных модулей в профильном блоке.

Возрастные особенности детей: в 12-17 лет для детей важно разработать систему мотивации участия во всем, например, рейтинговая система (дети очень любят соревноваться - кто больше). Они уже не маленькие, поэтому многое понимают, и готовы

во всем вам помогать. В этом возрасте у них особенно развито желание лидерства, а также дети с удовольствием участвуют во всевозможных конкурсах и соревнованиях. Они понимают правила игры или идеи выступления, но могут быть обременены подростковыми комплексами и не сформированы окончательно. Этот возраст характеризует также стремление к сплочённости. Этим детям очень нравится быть командой, быть лучше всех.

Группы формируются из расчета 10-12 человек. Система набора в группы осуществляется по собственному желанию ребенка.

Формы и режим занятий: занятия строятся с учётом индивидуальных особенностей воспитанников, что позволяет заинтересовать, увлечь каждого ребёнка, раскрыть его творческие способности. При изучении тем программа предусматривает использование фронтальной, индивидуальной и групповой формы учебной работы обучающихся:

- фронтальная форма - для изучения нового материала, информация подаётся всей группе из 10-12 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 (3-5 человека).

Занятия групп проводятся 2 раза в неделю по 2 и 1 часу, т.е. 3 часа в неделю (108 часов в год).

Данная программа подходит для реализации в рамках дистанционного обучения в случае отсутствия возможности проведения очных занятий или исходя из санитарно-гигиенических требований.

Прогнозируемые результаты образовательной деятельности.

По окончании обучения обучающиеся **должны знать:**

- решение ряда задач с использованием образовательных технических и программных средств;
- как реализовать и защитить собственный научно-технический проект;
- специфики различных языковых конструкций;
- основы алгоритмизации и программирования микроконтроллеров.

Должны уметь:

- принимать или намечать учебную задачу, ее конечную цель;
- решать поставленные задачи самостоятельно;

- прогнозировать результаты работы;
- планировать ход выполнения задания;
- рационально выполнять задание;
- руководить работой группы или коллектива;
- высказываться устно в виде сообщения, рецензии или доклада;
- совершенствовать различные ранее известные модели и алгоритмы, создавать творческие проекты;
- представлять одну и ту же информацию различными способами.

Фонд оценочных средств, методики и формы оценки учебных достижений:

- в течение курса предполагаются регулярные практики, на которых решение поставленной заранее известной задачи принимается в свободной форме (не обязательно предложенной преподавателем);
- по окончании курса учащиеся защищают творческий проект, требующий проявить знания и навыки по ключевым темам;
- полученные знания и навыки могут быть проверены на открытых конференциях, конкурсах и состязаниях, куда направляются наиболее успешные ученики.

Оценка достижения планируемых результатов освоения программы осуществляется по трем уровням: высокий (от 80 до 100% освоения программного материала), средний (от 51 до 79% освоения программного материала), низкий (менее 50% освоения программного материала). Оценочные материалы — пакет диагностических методик, позволяющих определить достижение учащимися планируемых результатов представлен в Приложении 2.

Учебно-тематический план

№	Наименование тем	Кол-во часов	Теория	Практика
1	Модуль «Основы программирования и алгоритмизации»	36	20	16
1.1	РАЗДЕЛ Введение в программирование	21	12	9
1.2	РАЗДЕЛ Программирование микроконтроллеров	15	8	7
2	Модуль «Программирование и разработка игр»	36	14	22
2.1	РАЗДЕЛ Разработка игр на Unity	12	6	6
2.2	РАЗДЕЛ Проектная деятельность	24	8	16
3	Модуль «Веб-технологии и веб-разработка»	36	19	17
3.1	РАЗДЕЛ язык HTML и каскадная таблица стилей	23	15	8
3.2	РАЗДЕЛ Проектная деятельность	13	4	9
	Итого	108	53	55

Содержание программы

Модуль 1. «Основы программирования и алгоритмизации» (всего часов – 36)

В данном модуле обучающиеся изучают неотъемлемые основы программирования. Учатся обращаться с передовыми информационными технологиями и программным обеспечением. Закладывают фундамент для дальнейшей успешной деятельности в сфере ИТ. Обучающиеся познакомятся с программированием микроконтроллеров и сформируют навыки управления ими.

Цель модуля: знакомство с традиционными языковыми конструкциями, как в лекционной, так и в игровой формах, а также освоение функционирования и программирования интернет вещей.

Задачи модуля:

- изучить основы алгоритмизации;
- закрепить знания, умения и навыки известные по школьному курсу информатики;
- сформировать навыки программирования простейших программ;
- на основе полученных знаний о микроконтроллерах спроектировать подобие «умного дома».

Учебно-тематический план модуля «Основы программирования и алгоритмизации»

№	Наименование тем	Кол-во часов	Теория	Практика	Формы аттестации/ контроля
1	Модуль «Основы программирования и алгоритмизации»	36	20	16	
1.1.	РАЗДЕЛ 1. Введение в программирование	21	12	9	Опрос
1.	Вводное занятие. Инструктаж по ОТ и ТБ. Алгоритм и его формальное исполнение	1	1		Наблюдение, беседа
2.	Принципы разработки алгоритмов и программ	1	1	-	Беседа
3.	Изучение и прохождение Kodu game lab	1	-	1	Тренинг, игра
4.	Кодирование основных типов алгоритмических структур	1	1	-	Наблюдение
5.	Линейный алгоритм, организация ветвления и алгоритмическая структура циклов	1	1		Наблюдение
6.	Блок-схемы и графы	1	1	-	Беседа

7.	Языки программирования	2	2	-	Беседа
8.	Оператор ветвления	1	1	-	Наблюдение
9.	Операторы повторений	2	1	1	Наблюдение
10.	Изучение и прохождение CodeCombat	1	-	1	Тренинг, игра
11.	Функции и процедуры	2	1	1	Упражнение
12.	Массивы	1	1	-	Наблюдение
13.	Одномерные массивы	2	-	2	Упражнение
14.	Двумерные массивы	1	-	1	Упражнение
15.	Классы	2	1	1	Беседа
16.	Запись алгоритмов на языках программирования и решение задач	1	-	1	Тестирование
1.2	РАЗДЕЛ 2. Программирование микроконтроллеров	15	8	7	Опрос
1.	Что такое микроконтроллер?	2	1	1	Беседа
2.	Обзор языка программирования Arduino	1	1	-	Беседа
3.	Простейшие компоненты: кнопка, светодиод. Управление яркостью	2	1	1	Наблюдение
4.	Сенсоры. Протоколы обмена данными, получение информации с сенсоров	1	1	-	Наблюдение
5.	Вывод информации. LCD - дисплей. Соединение с компьютером	2	1	1	Наблюдение
6.	Как сделать свой дом умным?	1	1	-	Беседа
7.	Центры управления	2	1	1	Наблюдение
8.	Датчики	1	-	1	Наблюдение
9.	Системы видеонаблюдения	2	-	2	Упражнение
10.	Готовые наборы, умная техника	1	1	-	Тестирование

Раздел 1. Введение в программирование.

Тема 1. Вводное занятие. Инструктаж по ОТ и ТБ. Алгоритм и его формальное исполнение.

Теория (1ч.): введение в программу. Инструктаж по технике безопасности. Понятие алгоритма.

Понимание и чтение алгоритмов. Усвоение свойств алгоритмов.

Приведение примеров.

Тема 2. Принципы разработки алгоритмов и программ.

Теория (1ч.): виды алгоритмов. Примеры линейных алгоритмов и работа с ними. Связь понятий алгоритм и программа.

Тема 3. Изучение и прохождение Kodu game lab.

Практика (1ч.): Kodu game lab представляет собой интегрированную среду разработки для обучения основам программирования и алгоритмизации.

Тема 4. Кодирование основных типов алгоритмических структур.

Теория (1ч.): типы алгоритмических структур. Знакомство с ветвлением,

повторением и подпрограммами.

Тема 5. Линейный алгоритм, организация ветвления и алгоритмическая структура циклов.

Теория (1ч.): отличие линейных алгоритмов от разветвляющихся и повторяющихся.

Переход от линейного алгоритма к другим алгоритмическим структурам.

Практика (1ч.): обучающиеся составляют простейшие линейные алгоритмы, а затем модифицируют их.

Тема 6. Блок-схемы и графы.

Теория (1ч.): наглядное представление алгоритмов. Изучение принципов визуализации алгоритма в блок-схему или граф.

Тема 7. Языки программирования.

Теория (2ч.): что такое язык программирования? Какими бывают языки программирования? Для чего служат различные языковые системы? Примеры.

Тема 8. Оператор ветвления.

Теория (1ч.): оператор ветвления IF (ЕСЛИ), его языковая конструкция на различных языках. Применение ELSE (ИНАЧЕ). Примеры.

Тема 9. Операторы повторений.

Теория (1ч.): циклы WHILE (ПОКА), FOR (ДЛЯ). Применение.

Практика (1ч.): написание программы с использованием ветвления и повторения на различных языках (Pascal, Python, C++).

Тема 10. Изучение и прохождение CodeCombat.

Практика (1ч.): CodeCombat самый захватывающий и бесплатный способ в игровой форме научиться реальному программированию. Управляя персонажем путем написания кода на нескольких возможных языках программирования, вы совершенствуетесь и постепенно сталкиваетесь с все более запутанными заданиями и тем самым познаете основные принципы программирования.

Тема 11. Функции и процедуры.

Теория (1ч.): понятия функций и процедур. Написание подпрограмм на различных языках. Применение. Дополнительные библиотеки и их подключение.

Практика (1ч.): написание кода с использованием подпрограмм и подключением дополнительных библиотек.

Тема 12. Массивы.

Теория (1ч.): понятие массива. Виды массивов. Примеры.

Тема 13. Одномерные массивы.

Практика (2ч.): работа с одномерными массивами данных на практике. Составление программ с их применением.

Тема 14. Двумерные массивы.

Практика (1ч.): работа с двумерными массивами данных на практике. Составление программ с их применением.

Тема 15. Классы.

Теория (1ч.): понятие класс. Что можно описать с его помощью и как использовать в процессе программирования. Объектно-ориентированное программирование.

Практика (1ч.): работа посредством объектно-ориентированного программирования. Создание классов.

Тема 16. Запись алгоритмов на языках программирования и решение задач.

Практика (1ч.): решение задач с применением ранее изученного материала.

Раздел 2. Программирование микроконтроллеров.

Тема 1. Что такое микроконтроллер?

Теория (1ч.): понятие микроконтроллер. Для чего используется Arduino. Демонстрация возможностей наборов Матрешка Z.

Практика (1ч.): сборка различных проектов согласно инструкции.

Тема 2. Обзор языка программирования Arduino.

Теория (1ч.): Си-подобный язык программирования Arduino, представление простейших программ для управления микроконтроллером.

Тема 3. Простейшие компоненты: кнопка, светодиод. Управление яркостью.

Теория (1ч.): что такое кнопка и диод, их представление в электротехнике.

Практика (1ч.): сборка проекта «Светильник» при помощи набора Матрешка Z.

Тема 4. Сенсоры. Протоколы обмена данными, получение информации с сенсоров.

Теория (1ч.): работа различных сенсоров. Примеры использования.

Тема 5. Вывод информации. LCD - дисплей. Соединение с компьютером.

Теория (1ч.): портативный дисплей и вывод информации на него при помощи Arduino Uno и набора Матрешка Z.

Практика (1ч.): реализация заданного проекта согласно инструкции Матрешка Z.

Тема 6. Как сделать свой дом умным?

Теория (1ч.): как управлять температурой, влажностью, видеть что происходит у вас дома в ваше отсутствие и многое другое.

Тема 7. Центры управления.

Теория (1ч.): что может являться центром управления умным домом. Обзор портативного компьютера Raspberry Pi из набора Малина.

Практика (1ч.): работа с Raspberry pi из набора Малина.

Тема 8. Датчики.

Практика (1ч.): работа с датчиками из набора Матрешка Z на базе Arduino UNO.

Тема 9. Системы видеонаблюдения.

Практика (2ч.): установка системы видеонаблюдения и настройка управления через смартфон.

Тема 10. Готовые наборы, умная техника

Теория (1ч.): изучение готовых наборов умного дома, а также технический обзор на тему: «Какую технику в наше время можно назвать умной».

Модуль 2. «Программирование и разработка игр» (всего часов – 36)

Обучающиеся сформируют фундаментальные знания о процессе создания игр в современном мире. Спроектируют свои собственные игровые оболочки с помощью ранее полученных умений.

Цель модуля: освоить работу и программирование самостоятельных игровых систем, а также изучить необходимое программное обеспечение в процессе создания игр.

Задачи модуля:

- научиться планировать деятельность и реализовать в командах простейшие игровые проекты;
- научиться программировать 2D игры, изучить основы искусственного интеллекта, физики, анимации, текстур и многого другого.

Учебно-тематический план модуля «Программирование и разработка игр»

№	Наименование тем	Кол-во часов	Теория	Практика	Формы аттестации/контроля
2	Модуль «Программирование и разработка игр»	36	14	22	
2.1	РАЗДЕЛ 1. Разработка игр на Unity	12	6	6	Опрос
1.	2D и 3D моделирование	1		-	Беседа
2.	3D моделирование в Blender	1	-	1	Упражнение
3.	Работа с анимацией и текстурами	2	1	1	Беседа
4.	Создание текстур при помощи Gimp	1	-	1	Упражнение
5.	Звук и озвучивание	2	2	-	Наблюдение
6.	Работа со звуком в Audacity	1	-	1	Упражнение
7.	Редакторы кода и скрипты	1	1	-	Наблюдение
8.	MonoDevelop, Microsoft Visual Studio, Notepad ++	1	-	1	Беседа
9.	Межплатформенная среда разработки компьютерных игр Unity	2	1	1	Тестирование

2.2	РАЗДЕЛ 2. Проектная деятельность	24	8	16	Опрос
1.	Графика и обработка пошаговых событий	1	1	-	Беседа
2.	Проект: «Крестики-нолики»	2	-	2	Упражнение
3.	Проект: «Четыре в ряд»	1	-	1	Упражнение
4.	Синхронизация, движение, столкновение и анимация в реальном времени	2	2	-	Наблюдение
5.	Проект: «Space Invaders»	1	-	1	Упражнение
6.	Проект: «Тетрис»	2	-	2	Упражнение
7.	Переинженеринг	1	1	-	Беседа
8.	Планирование, анализ столкновений, физика, искусственный интеллект	2	2	-	Наблюдение
9.	Проект: «Бильярд»	1	-	1	Упражнение
10.	Проект: «Pac Man»	2	-	2	Упражнение
11.	Платформеры, Action/Adventure, RPG	1	1	-	Беседа
12.	Проект: «LIMBO»	2	-	2	Упражнение
13.	Проект: «Ori»	1	-	1	Упражнение
14.	Проектная деятельность и защита	2	-	2	Тренинг
15.	Шоу проектов	1	-	1	Тренинг
16.	3D игры	1	1	-	Беседа
17.	Проект: «Cubium Dreams»	1	-	1	Тестирование

Раздел 1. Разработка игр на Unity.

Тема 1. 2D и 3D моделирование.

Теория (1ч.): моделирование. Различия 2D и 3D моделирования. Бесплатные программные средства.

Тема 2. 3D моделирование в Blender.

Практика (1ч.): Blender – профессиональное свободное и открытое программное обеспечение для создания трёхмерной компьютерной графики, включающее в себя средства моделирования, скульптинга, анимации, симуляции, рендеринга, постобработки и монтажа видео со звуком, компоновки с помощью «узлов», а также создания 2D анимаций. Знакомство с интерфейсом программы. Создание простейших моделей.

Тема 3. Работа с анимацией и текстурами.

Теория (1ч.): создание анимации и добавление готовых текстур инструментами Blender.

Практика (1ч.): моделирование 3D-модели, добавление текстуры и анимирование по ключевым кадрам.

Тема 4. Создание текстур при помощи Gimp.

Практика (1ч.): создание бесшовных текстур при помощи Gimp. GNU Image Manipulation Program или GIMP — свободно распространяемый растровый графический редактор, программа для создания и обработки растровой графики и частичной поддержкой работы с векторной графикой.

Тема 5. Звук и озвучивание.

Теория (2ч.): откуда рождается звук? Бесплатные программные средства для создания и редактирования звуков.

Тема 6. Работа со звуком в Audacity.

Практика (1ч.): Audacity — свободный многоплатформенный аудиоредактор звуковых файлов, ориентированный на работу с несколькими дорожками. Изучение программного интерфейса.

Тема 7. Редакторы кода и скрипты.

Теория (1ч.): что такое скрипт и почему необходимы редакторы кода.

Тема 8. MonoDevelop, Microsoft Visual Studio, Notepad ++.

Практика (1ч.): изучение на практике различных программных средств и оболочек для редактирования кода.

Тема 9. Межплатформенная среда разработки компьютерных игр Unity.

Теория (1ч.): Unity. Как создавать игры для различных платформ с его помощью и почему для этого необходимы ранее изученные приложения в лице Blender, Gimp, Audacity и Notepad++.

Практика (1ч.): интерфейс Unity. Добавление объектов и создание игровой сцены.

Раздел 2. Проектная деятельность.

Тема 1. Графика и обработка пошаговых событий.

Теория (1ч.): обработка пошаговых игровых событий и типы игровой графики.

Тема 2. Проект: «Крестики-нолики».

Практика (2ч.): разработка проекта под кодовым названием «Крестики-нолики». В данном задании требуется воссоздать собственное видение или подобие известной игры.

Тема 3. Проект: «Четыре в ряд».

Практика (1ч.): разработка проекта под кодовым названием «Четыре в ряд». В данном задании требуется воссоздать собственное видение или подобие известной игры.

Тема 4. Синхронизация, движение, столкновение и анимация в реальном времени.

Теория (2ч.): как реализовать многопоточность, реагировать на столкновения игровых объектов, запускать анимации в реальном времени.

Тема 5. Проект: «Space Invaders».

Практика (1ч.): разработка проекта под кодовым названием «Space Invaders». В данном задании требуется воссоздать собственное видение или подобие известной игры.

Тема 6. Проект: «Тетрис».

Практика (2ч.): разработка проекта под кодовым названием «Тетрис». В данном задании требуется воссоздать собственное видение или подобие известной игры.

Тема 7. Переинженеринг.

Теория (1ч.): понятие переинженеринга. Как его избежать и чем оно опасно в рамках программирования и игростроения.

Тема 8. Планирование, анализ столкновений, физика, искусственный интеллект.

Теория (2ч.): добавление искусственного интеллекта, физических свойств игровым объектам.

Тема 9. Проект: «Бильярд».

Практика (1ч.): разработка проекта под кодовым названием «Бильярд». В данном задании требуется воссоздать собственное видение или подобие известной игры.

Тема 10. Проект: «Pac Man».

Практика (2ч.): разработка проекта под кодовым названием «Pac Man». В данном задании требуется воссоздать собственное видение или подобие известной игры.

Тема 11. Платформеры, Action/Adventure, RPG.

Теория (1ч.): изучение различных и распространенных игровых жанров, над элементами которых придется работать в процессе обучения.

Тема 12. Проект: «LIMBO».

Практика (2ч.): разработка проекта под кодовым названием «LIMBO». В данном практическом задании требуется воссоздать собственное видение или подобие одной выбранной механики из известной игры на выбор учащегося.

Тема 13. Проект: «Ori».

Практика (1ч.): разработка проекта под кодовым названием «Ori». В данном практическом задании требуется воссоздать собственное видение или подобие одной выбранной механики из известной игры на выбор учащегося.

Тема 14. Проектная деятельность и защита.

Практика (2ч.): дополнительное время на полноценную реализацию выбранной игровой механики. Создание паспорта проекта и презентации на проектную защиту. Подготовка выступления. Как продать свой проект?

Тема 15. Шоу проектов.

Практика (1ч.): анализ проектной деятельности учащихся. Демонстрация и защита лучших работ и представление игровых решений.

Тема 16. 3D игры.

Теория (1ч.): обзор различий геймдизайна и внутреннего игростроения в 2D- и 3D-форматах.

Тема 17. Проект: «Cubium Dreams».

Практика (1ч.): разработка проекта под кодовым названием «Cubium Dreams». В

данном задании требуется воссоздать собственное видение или подобие малоизвестной игры в 3D. Или аналог любой выбранной игровой механики в 3D пространстве.

Модуль 3. «Веб-технологии и веб-разработка» (всего часов – 36)

В данном модуле обучающиеся получают необходимые знания, умения и навыки для создания собственных интернет ресурсов и целостное представление о сущности всемирной паутины.

Цель модуля: знакомство с языком гипертекстовой разметки HTML и каскадной таблицей стилей CSS.

Задачи модуля:

- научиться конструировать и верстать простейшие структуры сайтов самостоятельно.
- познакомиться с программным обеспечением для работы с интернет ресурсами.

Учебно-тематический план модуля «Веб-технологии и веб-разработка»

№	Наименование тем	Кол-во часов	Теория	Практика	Формы аттестации/ контроля
3	Модуль «Веб-технологии и веб-разработка»	36	19	17	
3.1	РАЗДЕЛ 1. Язык HTML и каскадная таблица стилей CSS	23	15	8	Опрос
1.	Передача информации	2	2	-	Беседа
2.	Базовые понятия компьютерных сетей	1	1	-	Беседа
3.	Локальные и глобальные сети. Конфигурирование компьютеров в локальной сети	2	1	1	Беседа
4.	Доменная система и протоколы передачи данных	1	1	-	Беседа
5.	Всемирная глобальная паутина. Язык разметки гипертекстовых страниц HTML: обзор возможностей	2	1	1	Беседа
6.	Структура HTML-документа	1	1	-	Наблюдение
7.	Модульность и виды подходов к верстке	2	1	1	Наблюдение
8.	Табличная верстка	1	1	-	Упражнение
9.	Верстка слоями	2	1	1	Упражнение
10.	Блочная верстка	1	1	-	Упражнение
11.	Верстка фреймами	2	1	1	Упражнение
12.	Адаптивная верстка	1	1	-	Беседа
13.	Возможности и методы HTML. Создание собственных веб-страниц и размещение их на сервере	2	-	2	Упражнение

14.	Валидность и кроссбраузерность	1	1	-	Упражнение
15.	Понятие каскадной таблицы стилей CSS, применение и написание	2	1	1	Тестирование
3.2	РАЗДЕЛ 2. Проектная деятельность	13	4	9	Опрос
1.	Целеполагание, целевая аудитория и функционал сайта	1	1	-	Беседа
2.	Содержание и структура сайта	2	1	1	Беседа
3.	Дизайнерские идеи и решения	1	1	-	Наблюдение
4.	Создание персонального сайта	2	-	2	Упражнение
5.	Создание личного блога	1	-	1	Упражнение
6.	Создание тематического сайта	2	-	2	Упражнение
7.	Создание новостного сайта	1	-	1	Упражнение
8.	Создание интернет-магазина	2	-	2	Упражнение
9.	Размещение сайта в интернете	1	1	-	Тестирование

Раздел 1. Язык HTML и каскадная таблица стилей CSS.

Тема 1. Передача информации.

Теория (2ч.): передача информации на расстояния посредством компьютерных сетей.

Тема 2. Базовые понятия компьютерных сетей.

Теория (1ч.): топология сети, протоколы передачи данных, адресация.

Тема 3. Локальные и глобальные сети. Конфигурирование компьютеров в локальной сети.

Теория (1ч.): как объединить компьютеры в локальную сеть. Отличие локальных сетей от глобальных.

Практика (1ч.): создание локальной сети на практике.

Тема 4. Доменная система и протоколы передачи данных.

Теория (1ч.): протоколы TCP/IP и доменная система. Что означают всемирно известные .com, .ru, .edu и многое другое.

Тема 5. Всемирная глобальная паутина. Язык разметки гипертекстовых страниц HTML: обзор возможностей.

Теория (1ч.): всемирная сеть интернет и язык гипертекстовой разметки веб-страниц HTML, каскадная таблица стилей CSS. Программное обеспечение для создания веб-страниц. Конструктор WIX.

Практика (1ч.): работа с заголовками и добавление текста на HTML-страницу.

Тема 6. Структура HTML-документа.

Теория (1ч.): рассмотрение структуры веб-страниц. Head, body, footer.

Тема 7. Модульность и виды подходов к верстке.

Теория (1ч.): таблицы, слои, блоки, фреймы, модули и комбинирование различных

подходов к верстке веб-страниц.

Практика (1ч.): рукописная заготовка проекта своего сайта.

Тема 8. Табличная верстка.

Теория (1ч.): обзор метода верстки веб-документов, при котором в качестве структурной основы для расположения текстовых и графических элементов документа используются таблицы.

Тема 9. Верстка слоями.

Теория (1ч.): обзор метода верстки веб-документов, при котором в качестве структурной основы для расположения текстовых и графических элементов документа используются слои.

Практика (1ч.): проектирование своего сайта.

Тема 10. Блочная верстка.

Теория (1ч.): обзор метода верстки веб-документов, при котором в качестве структурной основы для расположения текстовых и графических элементов документа используются блоки.

Тема 11. Верстка фреймами.

Теория (1ч.): обзор метода верстки веб-документов, при котором в качестве структурной основы для расположения текстовых и графических элементов документа используются фреймы.

Практика (1ч.): создание своего сайта иными методами верстки и анализ полученных результатов.

Тема 12. Адаптивная верстка.

Теория (1ч.): обзор метода верстки веб-документов, при котором в качестве структурной основы для расположения текстовых и графических элементов документа используются сочетания и различные комбинации традиционных подходов к верстке веб-страниц.

Тема 13. Возможности и методы HTML. Создание собственных веб-страниц и размещение их на сервере.

Практика (2ч.): зная содержание и компоновку своего сайта применить полученные знания и сверстать его, используя комбинированный стиль.

Тема 14. Валидность и кроссбраузерность.

Теория (1ч.): понятие валидности и кроссбраузерности. Соответствие сайтов определенным правилам и работа в разных браузерах и версиях.

Тема 15. Понятие каскадной таблицы стилей CSS, применение и написание.

Теория (1ч.): CSS. Почему принято интегрировать на веб-страницу различные

стили оформления из отдельного файла.

Практика (1ч.): работа с каскадной таблицей стилей. Изменение оформления ранее созданного сайта.

Раздел 2. Проектная деятельность.

Тема 1. Целеполагание, целевая аудитория и функционал сайта.

Теория (1ч.): с чего лучше начать разработку своего нового проекта.

Тема 2. Содержание и структура сайта.

Теория (1ч.): как определиться со структурой сайта и его наполнением.

Практика (1ч.): рукописные заготовки новых идей.

Тема 3. Дизайнерские идеи и решения.

Теория (1ч.): как оформить разные типы сайтов.

Тема 4. Создание персонального сайта.

Практика (2ч.): верстка своей личной страницы в интернете.

Тема 5. Создание личного блога.

Практика (1ч.): верстка своего сайта-блога.

Тема 6. Создание тематического сайта.

Практика (2ч.): верстка сайта соответствующего определенной тематике.

Тема 7. Создание новостного сайта.

Практика (1ч.): верстка проекта «новостной сайт».

Тема 8. Создание интернет-магазина.

Практика (2ч.): верстка проекта «интернет-магазин»

Тема 9. Размещение сайта в интернете.

Теория (1ч.): Как разметить свою страницу в интернете, чтобы она стала достоянием общественности, и любой желающий мог ее посетить. Как продвигать свой сайт в поисковых системах.

Методическое обеспечение

№	Раздел	Методические виды продукции (разработки игр, походов, экскурсий, конкурсов, бесед, конференций и т.д.)	Рекомендации по проведению лабораторных и практических работ, по постановке опытов или экспериментов и т.д.	Дидактический и лекционный материалы, тематика (или методики) опытной или исследовательской работы и т.д.
Модуль «Основы программирования и алгоритмизации»				
1	Введение в программирование	The Official Guide to Creating Your Own Video Games	Инструкция по технике безопасности в процессе работы за ноутбуками	Ноутбук – 10шт., ПО
Модуль «Программирование микроконтроллеров и разработка игр»				
1	Программирование микроконтроллеров	The Internet of Things for Education: A Brief Guide	Инструкция по технике безопасности в процессе работы за ноутбуками	ПО Arduino IDE, The Internet of Things 5 шт, набор «Матрешка Z» - 5 шт, Ноутбук – 10шт
2	Разработка игр на Unity	Обучающий курс https://learn.unity.com/	Инструкция по технике безопасности в процессе работы за ноутбуками	ПО Unity 3D, Ноутбук – 10шт
3	Проектная деятельность	Обучающий курс https://learn.unity.com/	Инструкция по технике безопасности в процессе работы за ноутбуками	Ноутбук – 10шт, ПО
Модуль «Веб-технологии и веб-разработка»				
1	язык HTML и каскадная таблица стилей	Обучающий курс Learn to Code HTML & CSS	Инструкция по технике безопасности в процессе работы за ноутбуками	Ноутбук – 10шт, ПО
2	Проектная деятельность	Обучающий курс Learn to Code HTML & CSS	Инструкция по технике безопасности в процессе работы за ноутбуками	Ноутбук – 10шт, ПО

Материально-техническое обеспечение

Занятия по программе проводятся на базе СП «ЦВР» ГБОУ СОШ № 4 г.о. Сызрань, а именно в «Детском мини-технопарке «Квантум» созданном при поддержке Национального проекта – «Образование». Занятия организуются в специально отведенных кабинетах, соответствующих требованиям СанПиН и техники безопасности. В кабинетах имеется все необходимое оборудование для полноценной реализации программы.

Список литературы

Литература, используемая педагогом дополнительного образования:

1. Периферийные устройства вычислительной техники: учеб. пособие / под ред. Партыка Т. Л., Попов И. И. - 3-е изд., испр. и доп. — М. : ФОРУМ, 2016. — 432 с.
2. Таненбаум Э., Бос Х. Современные операционные системы [пер. с англ.]. 4-е изд. — СПб.: Питер, 2015. — 1120 с.
3. Новиков Ю.В. Основы локальных сетей/ Новиков Ю.В., Кондратенко С.В. — М.: Национальный Открытый университет "Интуит", 2016. — 407с.
4. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 [пер. с англ.]. — СПб.: Питер, 2015. — 688с.
5. Олифер В.Г., Олифер Н.А Компьютерные сети. Принципы, технологии, протоколы. Учебник для вузов [пер. с англ.]. 5-ое изд. — СПб.: Питер, 2016. — 992 с.
6. Колисниченко Д.Н. PHP и MySQL. Разработка Web-приложений. — 5е изд., перераб. и доп.- СПб.: БХВ-Петербург, 2015. —592с.
7. Робачевский А.М., Немнюгин С.А., Стесик О.Л. Операционная система UNIX. — 2-е изд., перераб. и доп. — СПб.. БХВ-Петербург, 2016. — 656 с.
8. Хоровиц П., Хилл У. Искусство схемотехники [пер. с англ.]. — 7-е изд, пер. — М.: Бином, 2014. — 704 с.
9. Джонс М. Х. Электроника — практический курс [пер. с англ.]. — М.: Техносфера, 2016. — 512 с.
10. Соммер У. Программирование микроконтроллерных плат Arduino/Freduino [пер. с нем.]. — СПб.: БХВ-Петербург, 2012. —244 с.
11. Блум Д. Изучаем Arduino: инструменты и методы технического [пер. с англ.]. — СПб.: БХВ-Петербург, 2015. — 336 с.
12. Монк С. Програмируем Arduino. Основы работы со скетчами [пер. с англ.]. — СПб.: Питер, 2016. — 176 с.
13. Язык программирования C++. Базовый курс, 5-е изд. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2014. – 1120 с. : ил. – Парал. тит. англ.

Литература, рекомендованная для чтения учащимся:

1. Основы HTML и CSS (<https://www.coursera.org/learn/snovy-html-i-css>).
2. Вычислительные системы, сети и телекоммуникации. Учебное пособие / под ред. А.П. Пятибратов, Л.П. Гудыно, А.А. Кириченко - 2016 - books.google.com (Дата обращения: 07.11.2016).

3. Введение в практическую электронику (<https://universarium.org/course/738>).
4. Строим роботов и другие устройства на Arduino. От светофора до 3Dпринтера (<https://www.coursera.org/learn/roboty-arduino>).
5. Знакомство с цифровой электроникой (<https://universarium.org/course/496>)
6. Введение в программирование (C++) (<https://stepik.org/course/363/>).
7. Джереми Блум Изучаем Arduino. <http://radiohata.ru/arduino/162-dzheremiblum-izuchaem-arduino-instrumenty-i-metody-tehnicheskogo-volshebstva.html>
8. Энциклопедический словарь юного техника. - М., «Педагогика», 2015.
9. Таненбаум Э.С. Архитектура компьютера[пер. с англ.] — 2015 — books.google.com (Дата обращения: 07.11.2016).

Дополнительные интернет-ресурсы:

1. www.school.edu.ru/int
2. <http://www.int-edu.ru>
3. <http://www.samsung.com/ru/iotacademy/>
4. <http://www.samsung.com/ru/iotacademy/equip/>
5. <http://radiohata.ru/arduino/162-dzheremi-blumizuchaem-arduino-instrumenty-i-metody-tehnicheskogo-volshebstva.html>
6. <http://avidreaders.ru/book/arduino-i-raspberry-pi-v-proektah.html>

ПРИЛОЖЕНИЯ

Календарно-тематический план

№	Наименование тем	Кол-во часов	Теория	Практика	Формы аттестации/контроля
1	Модуль «Основы программирования и алгоритмизации»	36	20	16	
1.1.	РАЗДЕЛ 1. Введение в программирование	21	12	9	Опрос
1.	Вводное занятие. Инструктаж по ОТ и ТБ. Алгоритм и его формальное исполнение	1	1		Наблюдение, беседа
2.	Принципы разработки алгоритмов и программ	1	1	-	Беседа
3.	Изучение и прохождение Kodu game lab	1	-	1	Тренинг, игра
4.	Кодирование основных типов алгоритмических структур	1	1	-	Наблюдение
5.	Линейный алгоритм, организация ветвления и алгоритмическая структура циклов	2	1	1	Наблюдение
6.	Блок-схемы и графы	1	1	-	Беседа
7.	Языки программирования	2	2	-	Беседа
8.	Оператор ветвления	1	1	-	Наблюдение
9.	Операторы повторений	2	1	1	Наблюдение
10.	Изучение и прохождение CodeCombat	1	-	1	Тренинг, игра
11.	Функции и процедуры	2	1	1	Упражнение
12.	Массивы	1	1	-	Наблюдение
13.	Одномерные массивы	2	-	2	Упражнение
14.	Двумерные массивы	1	-	1	Упражнение
15.	Классы	2	1	1	Беседа
16.	Запись алгоритмов на языках программирования и решение задач	1	-	1	Тестирование
1.2.	РАЗДЕЛ 2. Программирование микроконтроллеров	15	8	7	Опрос
1.	Что такое микроконтроллер?	2	1	1	Беседа
2.	Обзор языка программирования Arduino	1	1	-	Беседа
3.	Простейшие компоненты: кнопка, светодиод. Управление яркостью.	2	1	1	Наблюдение
4.	Сенсоры. Протоколы обмена данными, получение информации с сенсоров	1	1	-	Наблюдение
5.	Вывод информации. LCD - дисплей. Соединение с компьютером	2	1	1	Наблюдение
6.	Как сделать свой дом умным?	1	1	-	Беседа
7.	Центры управления	2	1	1	Наблюдение
8.	Датчики	1	-	1	Наблюдение
9.	Системы видеонаблюдения	2	-	2	Упражнение
10.	Готовые наборы, умная техника	1	1	-	Тестирование
2	Модуль «Программирование и разработка игр»	36	14	22	

2.1	РАЗДЕЛ 1. Разработка игр на Unity	12	6	6	Опрос
1.	2D и 3D моделирование	1	1	-	Беседа
2.	3D моделирование в Blender	1	-	1	Упражнение
3.	Работа с анимацией и текстурами	2	1	1	Беседа
4.	Создание текстур при помощи Gimp	1	-	1	Упражнение
5.	Звук и озвучивание	2	2	-	Наблюдение
6.	Работа со звуком в Audacity	1	-	1	Упражнение
7.	Редакторы кода и скрипты	1	1	-	Наблюдение
8.	MonoDevelop, Microsoft Visual Studio, Notepad ++	1	-	1	Беседа
9.	Межплатформенная среда разработки компьютерных игр Unity	2	1	1	Тестирование
2.2	РАЗДЕЛ 2. Проектная деятельность	24	8	16	Опрос
1.	Графика и обработка пошаговых событий	1	1	-	Беседа
2.	Проект: «Крестики-нолики»	2	-	2	Упражнение
3.	Проект: «Четыре в ряд»	1	-	1	Упражнение
4.	Синхронизация, движение, столкновение и анимация в реальном времени	2	2	-	Наблюдение
5.	Проект: «Space Invaders»	1	-	1	Упражнение
6.	Проект: «Тетрис»	2	-	2	Упражнение
7.	Переинженеринг	1	1	-	Беседа
8.	Планирование, анализ столкновений, физика, искусственный интеллект	2	2	-	Наблюдение
9.	Проект: «Бильярд»	1	-	1	Упражнение
10.	Проект: «Pac Man»	2	-	2	Упражнение
11.	Платформеры, Action/Adventure, RPG	1	1	-	Беседа
12.	Проект: «Never Alone»	2	-	2	Упражнение
13.	Проект: «Ori»	1	-	1	Упражнение
14.	Проект: «LIMBO»	2	-	2	Упражнение
15.	Проект: «Child of Light»	1	-	1	Упражнение
16.	3D игры	1	1	-	Беседа
17.	Проект: «Cubium Dreams»	1	-	1	Тестирование
3	Модуль «Веб-технологии и веб-разработка»	36	19	17	
3.1	РАЗДЕЛ 1. Язык HTML и каскадная таблица стилей CSS	23	15	8	Опрос
1.	Передача информации	2	2	-	Беседа
2.	Базовые понятия компьютерных сетей	1	1	-	Беседа
3.	Локальные и глобальные сети. Конфигурирование компьютеров в локальной сети	2	1	1	Беседа
4.	Доменная система и протоколы передачи данных	1	1	-	Беседа
5.	Всемирная глобальная паутина. Язык разметки гипертекстовых страниц HTML: обзор возможностей	2	1	1	Беседа
6.	Структура HTML-документа	1	1	-	Наблюдение
7.	Модульность и виды подходов к верстке	2	1	1	Наблюдение
8.	Табличная верстка	1	1	-	Упражнение
9.	Верстка слоями	2	1	1	Упражнение
10.	Блочная верстка	1	1	-	Упражнение

11.	Верстка фреймами	2	1	1	Упражнение
12.	Адаптивная верстка	1	1	-	Беседа
13.	Возможности и методы HTML. Создание собственных веб-страниц и размещение их на сервере	2	-	2	Упражнение
14.	Валидность и кроссбраузерность	1	1	-	Упражнение
15.	Понятие каскадной таблицы стилей CSS, применение и написание	2	1	1	Тестирование
3.2	РАЗДЕЛ 2. Проектная деятельность	13	4	9	Опрос
1.	Целеполагание, целевая аудитория и функционал сайта	1	1	-	Беседа
2.	Содержание и структура сайта	2	1	1	Беседа
3.	Дизайнерские идеи и решения	1	1	-	Наблюдение
4.	Создание персонального сайта	2	-	2	Упражнение
5.	Создание личного блога	1	-	1	Упражнение
6.	Создание тематического сайта	2	-	2	Упражнение
7.	Создание новостного сайта	1	-	1	Упражнение
8.	Создание интернет-магазина	2	-	2	Упражнение
9.	Размещение сайта в интернете	1	1	-	Тестирование
	Итого	108	53	55	

Контрольно-измерительные материалы по программе «IT-программирование»

Модуль 1: общеобразовательный модуль, в котором даются основные вводные понятия в индустрию IT-технологий. Обсуждаются такие темы как: этапы развития электроники; обсуждаются темы промышленной автоматизации. Таким образом, в модуле №1 обучающимся раскрывается траектория развития всех основных этапов развития информационных технологий и дается видение как это с течением времени отражалось в различных областях, таких как: автомобилестроение, электроника, медицина, исследование космоса, и постепенное стирание границ между этими, некогда самостоятельными областями (направлениями), в новое, пока еще формирующуюся область «Internet of Things» – «интернет вещей».

Тест №1

Учащимся будет предложено выбрать один или несколько вариантов ответа в тесте.

1. Зачем изобрели компьютер?
2. Что значит бинарный (двоичный) код?
3. Какую роль в истории ЭВМ сыграло изобретение транзистора?
4. Какую функцию в компьютере выполняет процессор?
5. Что такое «Интернет вещей»?

Тест №2

Проверка усвоения материала на тему: алгоритмы и постановка и решение задач. Учащимся предлагается выполнить (сформировать и описать) последовательность действий для достижения определенного результата.

1. Что такое алгоритм и зачем их постоянно придумывают и усложняют?
2. Как бы вы описали свои действия, если бы вам предложили организовать мероприятие, например: хакатон или выставку?
3. Какие способы записи алгоритмов вы знаете?
4. Для решения, каких задач, трудно подобрать алгоритм либо они вообще не применимы?
5. Из каких операторов состоит алгоритм работы программа на Arduino?

Модуль 2: базовый модуль, целью которого является знакомство учащихся с образовательной платформой Arduino. На занятия данного модуля учащиеся получают навыки основ программирования, общие теоретические знания устройства

микроконтроллеров их виды и различия. Приобретут практические навыки основ программирования микроконтроллеров. Системы на кристалле (СнК) – миниатюризация электронной компонентной базы, снижение тех. процесса в производстве микропроцессорной техники, повлияло на окружающий нас цифровой мир. Смартфоны, планшеты, ноутбуки, Smart-часы, фитнес браслеты, системы виртуальной реальности и микрокомпьютеры. Появившись относительно недавно и получив широкое распространение СнК уже прочно вошли в нашу повседневную жизнь. Их нельзя назвать процессором или микроконтроллером – это новый этап в развитии электроники, т.к. они включают в себя и высокопроизводительный процессор способный выполнять задачи привычной нам, операционной системы (Android, Windows, Linux), средства беспроводной связи Bluetooth и Wi-Fi, а также располагают средствами микроконтроллеров (порты управления, интерфейсы (UART, I2C, SPI) PWM, таймеры и все это в чипе размером с сим карту. На примере микрокомпьютера Raspberry Pi и ESP8266 мы познакомимся ближе с устройством и архитектурой СнК, а также напишем ряд программ для выполнения определенных задач.

Тест №1

1. Что такое микроконтроллер и чем он отличается от процессора в ПК?
2. Какие виды микроконтроллеров существуют и где они используются чаще всего?
3. Что представляет собой платформа Arduino и какая её главная цель?
4. Какие типы данных существуют в языке Си и в чем их различие?
5. Что такое оператор цикла, оператор условия, массив?

Тест №2

1. Что такое интерфейс? Зачем они нужны?
2. Какие языки программирования можно использовать при работе с СнК?
3. Что такое СнК? В чем отличие СнК от микроконтроллера и процессора в ПК?
4. Какие существуют способы передачи информации между Arduino и ПК?
5. Какой оператор в Arduino способен выводить текст на экран последовательного порта?

Модуль 3: познакомившись поближе с СнК и их многообразием, Вы удивитесь такой гибкости, обилию средств разработки. Занятия данного (заключительного) модуля, помогут объединить весь накопившийся опыт работы в программировании, проектировании несложных систем и выйти на финишную прямую где поднимем такие важные темы как безопасность IoT, web программирование для управления электроникой,

познакомимся с Back-end и Front-end разработкой, а также кратко разберем такую важную тему как: управление проектом с момента появления идеи и до получения первого результата. Обсудим чем сейчас пользуются менеджеры проектов, какие роли есть в команде, как планировать и выделять первоочередные задачи.

Тест №1

1. Какие средства защиты информации вам известны? Криптография, протоколы шифрование: WEP/WPA2, SSL. Где мы можем это увидеть и как применить в своих задачах?

2. Какие основные этапы работы при создании сайта? Например нужно разработать сайт для доставки пиццы. С чего вы начнете и какие технологии будете использовать? HTML, CSS, PHP, JavaScript, Python.

3. В чем преимущества JavaScript и чем он отличается от PHP? Динамические и статические страницы в чем разница?

4. Back-end и Front-end разработка: какое между ними различие? Какие инструменты?

5. Фреймворки и CMS (система управления контентом) Bitrix, WordPress, Joomla, Drupal, что это такое и какие их виды существуют (блог, форму, персональная страница), в чем их плюсы и минусы.

Тест №2

1. Операционная система Linux: ядро операционной системы, сборка, дистрибутив, Debian, Suse, Ubuntu, RedHat, ALT-Linux, Kniprix. Скажите: чем они отличаются и что их объединяет?

2. Язык программирования Python: где он используется, в решении каких задач можно им воспользоваться? Приведите пример.

3. Как вы понимаете процесс управления проектом с момента появления идеи и до получения первого результата? Ответ составьте в виде алгоритма или блок-схемы.

4. Что такое SCRUM и Agile? Какие системы управления проектом (ПО) вы знаете, в чем их преимущество?

5. Python + Raspberry Pi, какие библиотеки можно использовать для работы с портами GPIO на Raspberry Pi?. Как работать с камерой и сенсорным дисплеем? Опишите процесс, в виде алгоритма или блок-схемы.

Уровни усвоения	Результат
Высокий уровень освоения программы	Учащиеся демонстрируют высокую заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают отличное знание теоретического материала (выше 80%), практическое применение знаний воплощается в качественный проектный продукт представленный в рамках итоговой защиты.
Средний уровень освоения программы	Учащиеся демонстрируют достаточную заинтересованность в учебной, познавательной и творческой деятельности, составляющей содержание Программы. На итоговом тестировании показывают хорошее знание теоретического материала (от 51% до 79%), практическое применение знаний воплощается в проектный продукт, требующий незначительной доработки, но, тем не менее, успешно представленный в рамках итоговой защиты проектов.
Низкий уровень освоения программы	Учащиеся демонстрируют низкий уровень заинтересованности в учебной, познавательной и творческой деятельности, составляющей содержание программы. На итоговом тестировании показывают недостаточное знание теоретического материала (ниже 50%), практическая работа и итоговый проект не соответствует требованиям.