

Структурное подразделение дополнительного образования
«Дом детского и юношеского творчества «Успех» город Жигулевск» государственного
бюджетного общеобразовательного учреждения Самарской области средней
общеобразовательной школы №14 имени полного кавалера ордена Славы Николая
Георгиевича Касьянова города Жигулевск

Утверждаю: Руководитель
СП ДО «ДДЮТ «Успех»:
_____ О.Д. Трошенкова
20 июня 2025 г.

Принята на заседании
педагогического совета
от 20 июня 2025г.
Протокол №9

Дополнительная общеобразовательная общеразвивающая программа технической

направленности

«IT-квантум»

Возраст обучающихся: 6-18 лет

Срок реализации: 3 года

Разработчик программы:

Старший методист,
педагог дополнительного образования-
Костина Екатерина Петровна,
педагог дополнительного образования-
Новичков Максим Александрович

г. Жигулевск,
2025 года.

Оглавление

Краткая аннотация	4
Актуальность и новизна	5
Педагогическая целесообразность	6
Формы обучения	7
Цели и задачи программы	8
Воспитательная работа	8
Методы обучения	10
Принципы обучения	10
Ожидаемые результаты освоения программы.....	11
Критерии и способы определения результативности	12
Формы контроля	12
Содержание программы	14
1 год обучения: Базовый	15
<i>Модуль базового уровня сложности «Основы алгоритмизации и программирования в программе Kodu game lab»</i>	15
Учебно-тематический план модуля базового уровня сложности «Основы алгоритмизации и программирования в программе Kodu game lab»	17
Содержание модуля базового уровня сложности «Основы алгоритмизации и программирования в программе Kodu game lab».....	18
Планируемые результаты освоения модуля продвинутого уровня сложности «Основы алгоритмизации и программирования в программе Kodu game lab»	20
<i>Модуль базового уровня сложности модуля углубленного уровня сложности.....</i>	21
Учебно-тематический план модуля углубленного уровня сложности «Основы алгоритмизации и программирования в программе Scratch»	22
Содержание модуля углубленного уровня сложности «Основы алгоритмизации и программирования в программе Scratch»	23
Планируемые результаты модуля углубленного уровня сложности «Основы алгоритмизации и программирования в программе Scratch»	24
<i>Модуль продвинутого уровня сложности «Кодинг с нуля».....</i>	25
Учебно-тематический план модуля продвинутого уровня сложности «Кодинг с нуля»	26
Содержание модуля продвинутого уровня сложности.....	27
«Кодинг с нуля».....	27
Планируемые результаты освоения модуля продвинутого уровня сложности	28
«Кодинг с нуля».....	28
2 год обучения: Углубленный	29
<i>Модуль базового уровня сложности «Основы веб-разработки: HTML и CSS»</i>	29
Учебно-тематический план модуля базового уровня сложности «Основы веб-разработки: HTML и CSS»	30
Содержание модуля базового уровня сложности «Основы веб-разработки: HTML и CSS»...31	31
Планируемые результаты освоения модуля базового уровня сложности «Основы веб-разработки: HTML и CSS»	32
<i>Модуль углубленного уровня сложности «Знакомство с Django».....</i>	33
Учебно-тематический план модуля углубленного уровня сложности «Знакомство с Django»	34
Содержание модуля углубленного уровня сложности «Знакомство с Django»	35
Планируемые результаты модуля углубленного уровня сложности «Знакомство с Django»	37
<i>Модуль продвинутого уровня сложности «Программирование микроконтроллеров Arduino».....</i>	38
Учебно-тематический план модуля продвинутого уровня сложности «Программирование	

микроконтроллеров Arduino».....	39
Содержание модуля продвинутого уровня сложности «Программирование микроконтроллеров Arduino».....	40
Планируемые результаты модуля продвинутого уровня сложности «Программирование микроконтроллеров Arduino».....	42
3 год обучения: Продвинутый.	43
<i>Модуль базового уровня сложности «Python для продвинутых»</i>	43
Учебно-тематический план модуля базового уровня сложности «Python для продвинутых»	45
Содержание модуля базового уровня сложности «Python для продвинутых»	45
Планируемые результаты модуля базового уровня сложности «Python для продвинутых»	47
<i>Модуль углубленного уровня сложности «Объектно-ориентированное программирование»</i>	48
Учебно-тематический план модуля углубленного уровня сложности «Объектно- ориентированное программирование».....	49
Содержание модуля углубленного уровня сложности «Объектно-ориентированное программирование».....	50
Планируемые результаты модуля углубленного уровня сложности «Объектно- ориентированное программирование».....	51
<i>Модуль продвинутого уровня сложности «Умные устройства»</i>	52
Содержание модуля продвинутого уровня сложности «Умные устройства»	54
Планируемые результаты модуля углубленного уровня сложности «Умные устройства».	56
Материально-техническое обеспечение программы	57
Организационно-педагогические условия реализации образовательной программы	58
Календарный учебный график 1 года обучения	60
Календарный учебный график 2 года обучения	64
Календарный учебный график 3 года обучения	67
План воспитательной работы.....	71
Список литературы.....	74
<i>Приложение 1 «Входной контроль».....</i>	78
<i>Приложение 2 «Текущий контроль 1»</i>	81
<i>Приложение 3 «Текущий контроль 2»</i>	82
<i>Приложение 4. «Промежуточный контроль».....</i>	83
<i>Приложение 5. «Промежуточный контроль».....</i>	85
<i>Приложение 6. «Итоговый контроль».....</i>	86
<i>Приложение 7. «Итоговый контроль».....</i>	87
<i>Приложение 8. «Контроль воспитательной работы».....</i>	88
<i>Приложение 9. Анкета для родителей (обратная связь)</i>	90
<i>Приложение 10 Шаблон портфолио обучающегося</i>	92
<i>Приложение 11. Лист самооценки обучающегося</i>	93

Краткая аннотация

Дополнительная общеобразовательная общеразвивающая программа технической направленности «IT-квантум» является разноуровневой и предусматривает изучение модулей различного уровня сложности. Программа ориентирована на развитие технических и творческих способностей и умений обучающихся, организацию научно-исследовательской деятельности, профессионального самоопределения обучающихся.

Программа разработана с учетом следующих законодательных нормативно-правовых документов:

- ✓ Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- ✓ Указ Президента Российской Федерации от 02.07.2021 г. № 400 «О Стратегии национальной безопасности Российской Федерации»;
- ✓ Указ Президента Российской Федерации от 09.11.2022 г. № 809 «Об утверждении основ государственной политики по сохранению и укреплению традиционных российских духовно-нравственных ценностей»;
- ✓ Указ Президента Российской Федерации от 7.05.2024 г. № 309 «О национальных целях развития Российской Федерации на период до 2030 года и на перспективу до 2036 года»;
- ✓ Концепция развития дополнительного образования до 2030 года (утверждена распоряжением Правительства РФ от 31.03.2022 № 678-р);
- ✓ ИЗМЕНЕНИЯ, которые вносятся в распоряжение Правительства РФ от 31.03.2022 №678-р (утверждены распоряжением Правительства РФ от 15.05.2023 №1230-р);
- ✓ Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена распоряжением Правительства Российской Федерации от 29.05.2015 № 996-р);
- ✓ Постановление Правительства РФ от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- ✓ Приказ Министерства просвещения Российской Федерации от 27 июля 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- ✓ Приказ Министерства просвещения РФ от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;
- ✓ Приказ Министерства просвещения РФ от 21.04.2023 № 302 «О внесении изменений в Целевую модель развития региональных систем дополнительного образования детей, утвержденную приказом Министерства просвещения РФ от 3.09.2019 г. № 467»;
- ✓ Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям

- воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- ✓ Стратегия социально-экономического развития Самарской области на период до 2030 года (утверждена распоряжением Правительства Самарской области от 12.07.2017 № 441);
 - ✓ Письмо Министерства образования и науки РФ от 18.11.2015 № 09-3242 «О направлении информации» (с «Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»);
 - ✓ Письмо министерства образования и науки Самарской области от 30.03.2020 № МО-16-09-01/434-ТУ (с «Методическими рекомендациями по подготовке дополнительных общеобразовательных общеразвивающих программ к прохождению процедуры экспертизы (добровольной сертификации) для последующего включения в реестр образовательных программ, включенных в систему ПФДО»).
 - ✓ Программа предусматривает развитие творческих способностей детей, формирование начальных технических ЗУНов, а также, овладение soft и hard компетенциями и рассчитана на два года обучения по данному направлению.

Актуальность и новизна

Информационные технологии или ИТ (от англ. *Information Technologies*, сокр. *IT*) — методы, способы, приемы и процессы обработки информации с применением СВТ (средств вычислительной техники или программных и технических средств).

В широком понимании ИТ охватывают все области создания, передачи, хранения и восприятия информации, не ограничиваясь только компьютерными технологиями. Мы живем в век информатизации общества. Информационные технологии проникают в нашу жизнь с разных сторон. В этих условиях, умение создавать, хранить и изменять информацию, становится жизненно необходимым. В современном образовании детей большое внимание уделяется информатизации. Современные компьютерные технологии с их достаточно простым пользовательским интерфейсом способствуют нормированию «потребительского» отношения к ним – современное поколение детей и подростков может стать поколением «продвинутых пользователей», поэтому все больше и больше вызывают интерес информационные технологии.

Психологическая готовность ребёнка к жизни в информационном обществе должна формироваться с первых лет обучения, что предполагает овладение компьютерной грамотностью. Не менее важно формировать у обучающегося навыки алгоритмического мышления и умения логически мыслить. Психологи утверждают, что основные логические структуры мышления формируются в возрасте 5-11 лет и что запоздалое формирование этих структур протекает с большими трудностями и часто остается не завершенным. Умение для любой предметной области выделить систему понятий, представить их в виде совокупности значимых признаков, описать алгоритмы типичных действий улучшает ориентацию человека в

этой предметной области и свидетельствует о его развитом логическом мышлении. Следовательно, обучать детей в этом направлении целесообразно с начальной школы.

Педагогическая целесообразность

Педагогическая целесообразность заключается в создании условий для проявления учащимися творчества, инициативы, формирования навыков самоорганизации, самообслуживания, коммуникации, получении новых и закреплении имеющихся знаний в области информационно - коммуникационных технологий. Обучаясь по программе, дети решают серию кейсов, конечным результатом которой является созданные индивидуальные и коллективные проекты. Акцент обучения переносится не на овладение готовым знанием, а на его выработку, на створчество детей и педагога.

Изучение ИТ технологий с младшего школьного возраста, дает возможность учащимся создавать инновации своими руками, и заложить основы успешного освоения профессии инженера в будущем. В нашем информационном обществе, ИТ-сфера становится одной из ведущих сторон жизни, поэтому необходимо с детства учиться сбору, обработке и хранению информации, а также основам алгоритмизации и программирования. На что и направлена данная программа.

Отличительные особенности. Основной принцип определения содержания программы состоит в отборе учебного материала, опирающегося на современное состояние развития информационных технологий, и позволяющего организовать обучение в разновозрастных группах школьников. Программа является разноуровневой и имеет модульную структуру, позволяющую выстроить индивидуальную траекторию обучения, когда школьник выбирает всю программу, либо ее часть.

Отличительной особенностью от других программ, является использование в образовательном процессе промышленных средств программирования, передовых технологий в области электроники и программирования. В результате освоения программы, обучающиеся освоят практические навыки передовых технологий их применения, научатся понимать принципы работы, возможностей и ограничений технических устройств, предназначенных для автоматизированного поиска и обработки информации; развитие лидерских качеств и аналитического мышления.

Важным направлением в реализации целей и задач курса является интегрирование профессиональных, личностных и межличностных компетенций (командных компетенций, навыков ведение проекта, критическое мышление).

Новизна программы обусловлена использованием в образовательном процессе большого количества современных технических устройств, что позволяет сделать процесс обучения не только интереснее, но и нагляднее и информативнее. Использование при обучении “открытого” программного обеспечения позволяет учащимся свободно использовать его на

своих домашних устройствах, что позволяет им достаточно легко продолжить обучение и развиваться самостоятельно в этом направлении.

Важным направлением в реализации целей и задач курса является интегрирование в профессиональных, личностных и межличностных компетенций (командных компетенций, навыков ведение проекта, критическое мышление).

Формы обучения

Формы обучения определены образовательным учреждением СП ДО «ДДЮТ «Успех» г.Жигулёвск» ГБОУ СОШ №14 г.Жигулёвск на основании:

- Приказа Министерства просвещения Российской Федерации от 27 июля 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», регулирующий организацию и осуществление образовательной деятельности по дополнительным общеобразовательным программам;
- Постановления Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

В организации образовательно-воспитательного процесса по программе «ИТ квантум» предусмотрены следующие формы обучения: очное/заочное/дистанционное в виде урока, занятия, лекции, консультации, собеседования; обучение по индивидуальному плану.

Содержание программы учитывает возрастные психологические особенности детей школьного возраста, которые определяют выбор форм проведения занятий с обучающимися.

Форма обучения – очное, очно-заочное, заочное по образовательной программе, с применением дистанционных технологий и/или электронного обучения в виде практических занятий, занятий-соревнований, экскурсий. Workshop (рабочая мастерская-групповая работа, где все участники активны и самостоятельны); консультаций, метода проектов.

Адресат программы – Программа подразумевает изучение модулей различного уровня сложности и ориентирована на дополнительное образование обучающихся 6-18 лет разного уровня подготовки, проявляющих интерес к ИТ технологиям и программированию.

Наполняемость группы -12- 15 человек, группы могут быть разновозрастными.

Объем и срок освоения программы, режим занятий. Программа рассчитана на 3 года обучения, 324 часа (каждый год по 108 часов): занятия проходят 2 раза в неделю по 2 и 1 академических часа.

Цели и задачи программы

Цель: создание условий для формирования у обучающихся теоретических знаний и практических навыков компьютерной грамотности; сформировать у обучающихся базовые представления о языках программирования; организовать проектную научно-познавательную деятельность творческого характера.

Задачи:

Обучающие:

- ✓ Освоить основы программирования и алгоритмики.
- ✓ Изучить основы веб-разработки и разработки мобильных приложений.
- ✓ Познакомиться с микроконтроллерами и основами «интернета вещей» (IoT).
- ✓ Овладеть языком Python и основами объектно-ориентированного программирования.

Развивающие:

- ✓ Развить навыки критического и логического мышления.
- ✓ Стимулировать творческое мышление и самостоятельность.
- ✓ Формировать навыки работы в команде и публичных выступлений.
- ✓ Способствовать развитию способностей к проектированию и решению комплексных задач.

Воспитательные:

- ✓ Воспитать интерес к науке и технике, программированию и информационным технологиям.
- ✓ Развить личные качества, такие как упорство, дисциплинированность и ответственность.
- ✓ Сформировать уважительное отношение к современному цифровому обществу и важным технологическим изменениям.

Воспитательная работа

Дополнительно к образовательной цели и задачам, программа ставит перед собой важнейшую воспитательную цель — формирование гармонично развитой личности с высокой степенью социальной адаптации и гражданской зрелости, готовой жить и работать в современном информационном обществе.

Цель воспитания: формирование активной гражданской позиции, патриотизма, уважительного отношения к отечественным научным достижениям и традиционной русской культуре.

Задачи воспитания:

- Воспитать в обучающихся интерес к научно-техническому творчеству, инженерному делу и информтехнологиям.
- Сформировать у детей понимание значимости научного знания и технических изобретений для процветания России.
- Развить у школьников культуру безопасного обращения с компьютерами и современными устройствами.

- Воспитать осознанное отношение к личным данным и ответственности за их защиту.
- Стимулировать интерес к общественно полезным действиям и активному образу жизни.

Ожидаемые результаты воспитания:

- У обучающихся формируется уважение к истории и достижениям отечественных учёных и инженеров.
- Выпускники готовы применять свои знания и навыки для пользы общества и экономического благополучия страны.
- Дети чувствуют свою причастность к судьбе Родины и принимают активное участие в сохранении исторического наследия.

Формы воспитания:

- Кружковая работа, проводимая в формате исследований и проектов.
- Встречи с представителями компаний, работающими в сфере ИТ.
- Соревнования и хакатоны по программированию и разработке мобильных приложений.
- Участие в конкурсах, олимпиадах и соревнованиях разного уровня.
- Посещения музеев науки и техники, научных центров и предприятий.

Методы воспитания:

- Личный пример педагогов и успешных представителей ИТ-сообщества.
- Игровой подход, позволяющий каждому ребёнку проявить инициативу и творчество.
- Работа в группах, где каждый участник учится сотрудничеству и умению выслушивать мнения других.
- Самоанализ и рефлексия, помогающая осознать достигнутые результаты и поставить новые цели.

Диагностика результатов воспитания:

- Анализ динамики успеваемости и интереса к выбранной области.
- Наблюдение за поведением детей во время коллективных мероприятий.
- Оценка качества выполнения проектов и успехов в конкурсных мероприятиях.
- Обратная связь от родителей и партнёров образовательного учреждения.

Организационный раздел

Педагогический коллектив совместно с экспертами в области информационных технологий создаёт атмосферу, располагающую к творчеству и открытию новых горизонтов. Особое внимание уделяется сохранению здоровья детей и профилактике цифрового стресса.

Программа реализуется в течение трёх лет, постепенно усложняясь и расширяя спектр формируемых компетенций. Благодаря такому подходу достигается максимальная эффективность воспитательно-образовательного процесса, удовлетворяется потребность детей в творчестве и познании.

Таким образом, воспитательный компонент программы дополняет образовательную составляющую, создавая благоприятные условия для раскрытия потенциала и становления гармонично развитой личности.

Работа с родителями или законными представителями осуществляется в форме:

- родительских собраний;
- открытых занятий для родителей;
- консультаций в групповом чате студии;

Воспитательная работа осуществляется на основной учебной базе структурного подразделения дополнительного образования «Дом детского и юношеского творчества «Успех» г. Жигулевск», государственного бюджетного общеобразовательного учреждения Самарской области средней общеобразовательной школы № 14 имени полного кавалера ордена Славы Николая Георгиевича Касьянова города Жигулевска городского округа Жигулевск Самарской области (СП ДО «ДДЮТ «Успех» г.Жигулевск» ГБОУ СОШ № 14 г. Жигулевск), а также на базе школ города Жигулевска в рамках сетевого взаимодействия.

Методы обучения

- ✓ **Объяснительно-иллюстративный** метод обучения - обучающиеся получают знания в ходе беседы, объяснения, дискуссии, из учебной или методической литературы, через экранное пособие в "готовом" виде.
- ✓ **Репродуктивный метод** обучения – деятельность обучающихся носит алгоритмический характер, выполняется по инструкциям, предписаниям, правилам в аналогичных, сходных с показанным образцом ситуациях.
- ✓ **Метод проблемного изложения в обучении** – прежде чем излагать материал, перед обучающимися необходимо поставить проблему, сформулировать познавательную задачу, а затем, раскрывая систему доказательств, сравнивая точки зрения, различные подходы, показать способ решения поставленной задачи. Обучающиеся становятся свидетелями и соучастниками научного поиска.
- ✓ **Частично-поисковый** – метод обучения заключается в организации активного поиска решения выдвинутых в обучении (или самостоятельно сформулированных) познавательных задач в ходе подготовки и реализации творческих проектов.

Принципы обучения

Образовательный процесс строится на следующих **принципах**:

- Принцип научности. Его сущность состоит в том, чтобы ребёнок усваивал реальные знания, правильно отражающие действительность, составляющие основу соответствующих научных понятий.
- Принцип наглядности. Наглядные образы способствуют правильной организации

мыслительной деятельности ребёнка. Наглядность обеспечивает понимание, прочное запоминание;

- Принцип доступности, учета возрастных и индивидуальных особенностей детей в процессе обучения по программе. Предполагает соотнесение содержания, характера и объёма учебного материала с уровнем развития, подготовленности детей. Переходить от лёгкого к трудному, от известного к неизвестному. Но доступность не тождествится слёгкостью. Обучение, оставаясь доступным, сопряжено с приложением серьёзных усилий, что приводит к развитию личности;
- Принцип осознания процесса обучения. Данный принцип предполагает необходимость развития у ребёнка рефлексивной позиции: как я узнал новое, как думал раньше. Если ребёнок видит свои достижения, это укрепляет в нём веру в собственные возможности, побуждает к новым усилиям. И если ребёнок понимает, в чём и почему он ошибся, что ещё не получается, то он делает первый шаг на пути к самовоспитанию;
- Принцип воспитывающего обучения. Обучающая деятельность педагога, как правило, носит воспитывающий характер. Содержание обучения, формы его организации, методы и средства оказывают влияние на формирование личности в целом.

Ожидаемые результаты освоения программы

Предметные результаты:

- Освоят основы программирования и алгоритмизации.
- Изучат основы веб-разработки и мобильной разработки.
- Познакомятся с микроконтроллерами и технологиями «интернета вещей» (IoT).
- Овладеют языком Python и основными принципами объектно-ориентированного программирования.

Метапредметные результаты:

- Разовьют навыки критического и логического мышления.
- Сформируют творческое мышление и способность к самостоятельному решению задач.
- Приобретут навыки работы в команде и публичных выступлений.
- Научатся проектировать и решать комплексные задачи.

Личностные результаты:

- Воспитают интерес к науке, технике, программированию и информационным технологиям.
- Разовьют такие качества, как упорство, дисциплинированность и ответственность.
- Сформируют уважительное отношение к цифровому обществу и современным технологиям.

Ожидаемые результаты освоения программы сформулированы отдельно для каждого модуля обучения и соответствуют уровню его освоения.

Критерии и способы определения результативности

Для отслеживания результативности образовательной деятельности по программе проводятся следующие виды контроля:

- *Входной контроль* – при поступлении в группу проводится входной тест для определения уровня владения компьютером. (*Приложение 1*)
- *Текущий контроль* проводится в течение года по определению уровня подготовки обучающихся по усвоению изучаемых тем, а также их практических умений. (*Приложение 2, 3*)
- *Промежуточный контроль* проводится в конце планируемых этапов овладения умениями и знаниями изучаемых тем и уровня практических умений. (*Приложение 4, 5*)
- *Итоговый контроль* проводится по окончанию реализации программы. (*Приложение 6, 7*)

Итоговая практико-значимая работа рассматривается как обобщение опыта усвоения данного курса, систематизирует знания, практические умения и навыки, способы творческой деятельности, полученные в ходе практических занятий, выполнения самостоятельных и практических работ. Итоговая практико-значимая работа представляется в форме законченного проекта.

Формы контроля

Педагогическое наблюдение, тестирование, выполнение практических заданий педагога, анализ качества выполнения работы.

В ходе реализации данной образовательной программы создается объективная картина таланта или отсутствие его у каждого занимающегося.

Контрольные занятия по проверке усвоения материала, опрос обучающихся по пройденному материалу;

- Контроль соблюдения техники безопасности в компьютерном классе;
- Тестирование на знание теоретического материала;
- Самостоятельные задания для каждого обучающегося;
- Творческие задания на практических заданиях;
- Представление своих работ перед группой;
- Участие в соревнованиях, хакатонах и конкурсах различного уровня.

Критерии оценивания

Уровень	Критерии оценивания
Низкий уровень	<p>слабо прослеживается освоение теоретического материала, низкий уровень личных достижений при выполнении кейсов;</p> <p>качество выполнения работ: работа выполнена технически плохо, неаккуратно, имеет низкую сложность, выполнен небольшой объем работ; слабая увлеченность выполнением работы;</p> <p>не активное желание изучать достижения современной науки, дизайна, техники; обучающийся не заинтересован получить первоначальные сведения о профессиональных предпочтениях;</p> <p>самостоятельность: обучающийся при выполнении задания опирается на помощь педагога, нуждается в дополнительных пояснениях, помощи, поощрении действий;</p> <p>общительность и культура общения в группе: не поддерживает беседу в группе, ведет себя отстраненно или иное.</p>
Средний уровень	<p>удовлетворительно (достаточно хорошо) прослеживается освоение теоретического материала, средний уровень личных достижений при выполнении кейсов;</p> <p>качество выполнения работ: работа выполнена технически удовлетворительно, аккуратно с небольшими помарками, имеет среднюю сложность, имеет уникальность, выполнен достаточный объем работ; обучающийся увлечен выполнением работы;</p> <p>обучающийся имеет желание изучать достижения современной науки, дизайна, техники; заинтересован получить первоначальные сведения о профессиональных предпочтениях;</p> <p>самостоятельность: обучающийся может работать самостоятельно, опираясь на словесный комментарий и демонстрацию действий педагогом; выполняет работу в соответствии с поставленным условием, иногда нуждается в дополнительных пояснениях со стороны педагога;</p> <p>общительность и культура общения в группе: участвует в обсуждениях, коллективной работе, поддерживает дружеские отношения и способствует созданию рабочей атмосферы в группе.</p>
Высокий уровень	<p>хорошо прослеживается освоение теоретического материала, высокий уровень личных достижений при выполнении кейсов;</p> <p>качество выполнения работ: работа выполнена технически идеально, аккуратно, имеет высокую сложность, отличается уникальностью и оригинальностью решения, выполнен большой объем работ; обучающийся увлечен выполнением работы;</p> <p>обучающийся имеет желание изучать достижения современной науки, дизайна, техники; заинтересован получить первоначальные сведения о профессиональных предпочтениях;</p> <p>самостоятельность: обучающийся справляется с поставленными задачами самостоятельно, не нуждается в дополнительной помощи со стороны педагога, старается использовать на занятиях уже имеющиеся знания и умения, творчески подходит к выполнению заданий;</p> <p>общительность и культура общения в группе: участвует в обсуждениях, коллективной работе, поддерживает дружеские отношения и способствует созданию рабочей атмосферы в группе.</p>

Содержание программы

№	Наименование модуля	Количество часов			
		Всего	Теория	Практика	
1 год обучения: Базовый					
Базовый уровень					
1.	Основы алгоритмизации и программирования в программе Kodu game lab.	36	8	28	
Углубленный уровень					
2.	Основы алгоритмизации и программирования в программе Scratch	36	10	26	
Продвинутый уровень					
3.	Кодинг с нуля	36	10	26	
Итого		108	28	80	
2 год обучения: Углубленный					
Базовый уровень					
1.	Основы веб-разработки: HTML и CSS	36	8	28	
Углубленный уровень					
2.	Знакомство с Django	36	10	26	
Продвинутый уровень					
3.	Программирование микроконтроллеров Arduino	36	8	28	
Итого		108	26	82	
3 год обучения: Продвинутый					
Базовый уровень					
1.	Python для продвинутых	36	9	27	
Углубленный уровень					
2.	Объектно-ориентированное программирование	36	9	27	
Продвинутый уровень					
3.	Умные устройства	36	12	24	
Итого		108	30	78	
Всего за 3 года обучения		324	84	240	

1 год обучения: Базовый.

Первый базовый год обучения по дополнительной образовательной программе направлен на постепенное погружение обучающихся в мир программирования и алгоритмики. В течение года ребята осваивают базовые навыки программирования и алгоритмизации в двух визуально привлекательных средах: Kodu Game Lab и Scratch. Они получают представление о понятиях алгоритма, основных алгоритмических конструкциях, типах данных и правилах их обработки. Также вводится элемент «кодинга с нуля», направленный на первоначальный контакт с настоящим программированием на компьютере. Таким образом, обучающиеся укрепляют интерес к программированию, развиваются алгоритмическое мышление и готовят базу для перехода на следующий уровень.

Модуль базового уровня сложности

«Основы алгоритмизации и программирования в программе Kodu game lab»

Цель: Формирование навыков правильного и рационального планирования действий, развитие способностей ясно и чётко мыслить, свободное владение приёмами обработки информации, стимулирование устойчивого интереса к программированию путём изучения возможностей среды визуально-объектного программирования Kodu Game Lab.

Задачи:

Обучающие:

- обучить общим принципам построения алгоритмов;
- ознакомить с основными алгоритмическими конструкциями;
- сформировать понятие об алгоритмических структурах — линейных, циклических, с ветвлением;
- сформировать навыки работы с объектами;
- сформировать первоначальные навыки программирования с применением учебных визуальных сред программирования и навыки работы с различными исполнителями;
- научить разрабатывать алгоритмы для различных исполнителей.

Развивающие

- Способствовать формированию навыков алгоритмического подхода к решению задач;
- способствовать расширению кругозора, развитию памяти, внимания, творческого воображения, математического и образного мышления;
- обучить поиску, отбору, организации и использованию информации для решения стоящих задач и достижения поставленных целей;
- сформировать навыки планирования целенаправленной деятельности человека, в том числе учебной деятельности.

Воспитательные:

- Способствовать начальному формированию и развитию логического мышления и пространственного воображения в оптимальные срок
- воспитать у обучающихся умение самостоятельно работать, делать выводы, грамотно излагать свои мысли, рассуждать по аналогии, по индукции, способствовать развитию творческого воображения;
- мотивировать к получению образования в ИТ-сфере посредством организации продуктивной творческой деятельности и создания ситуации успеха.

Учебно-тематический план модуля базового уровня сложности
«Основы алгоритмизации и программирования в программе Kodu game lab»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теори я	практика	
1.	Основные понятия алгоритмов.	2	2		прослушивание
2.	Основные алгоритмические конструкции	2	1	1	Теория/ Практическая работа
3.	Данные	2	2		Теория
4.	Логические основы алгоритмизации	2	1	1	Теория/ Практическая работа
5.	Программирование в среде KoduGameLab	2	2		Теория
6.	Знакомство с интерфейсом и основными принципами работы с программой KoduGameLab	2		2	Практическая работа
7.	Создание ландшафтов (миров), добавление объектов. Создание персонажей с учетом игровой среды, действия персонажей	4		4	Практическая работа
8.	Перемещение персонажей в макросреде Kodu с использованием клавиатуры и мыши.	2		2	Практическая работа
9.	Режим программирования, основные операторы Kodu	2		2	Практическая работа
10.	Создание путей, выбор поведения персонажей. Создание клонов.	2		2	Практическая работа
11.	Страницы, функции, ракурс обзора.	2		2	Практическая работа
12.	Таймер, индикатор здоровья. Опция «Родитель»	2		2	Практическая работа
13.	Создание нового игрового мира	4		4	Workshop/ Консультации
14.	Планирование и создание собственного проекта. Защита проекта	6		6	Workshop/ Консультации
		36	8	28	

Содержание модуля базового уровня сложности
«Основы алгоритмизации и программирования в программе Kodu game lab»

1. Основные понятия алгоритмов.

Теория: Вводный инструктаж. Ознакомление с оборудованием кабинета. Понятие алгоритма. Свойства алгоритмов. Формы записей алгоритмов. Общие принципы построения алгоритмов.

2. Основные алгоритмические конструкции.

Теория: Описание линейных и разветвляющихся структур алгоритмов. Циклические структуры. Организация алгоритмов циклической структуры. Циклические структуры с заданным числом повторений и итерационные циклы. Алгоритмическое описание вложенных циклических структур. Классификация постановок технических задач. Примеры построения алгоритма решения задачи.

Практика:

Составление блок-схем линейных алгоритмов

Составление блок-схем разветвляющихся алгоритмов

Составление блок-схем циклических алгоритмов

Составление блок-схем алгоритмов сортировки данных

3. Данные.

Теория: Понятие и типы. Основные базовые типы данных и их характеристика. Структурированные типы данных и их характеристика. Методы сортировки данных.

4. Логические основы алгоритмизации

Теория: основы алгебры логики. Логические операции с высказываниями: конъюнкция, дизъюнкция, инверсия. Законы логических операций. Таблицы истинности.

Практика: Составление таблиц истинности.

5. Программирование в среде Kodu Game Lab

Теория: Kodu Game Lab – визуальная среда разработки трехмерных игр. Знакомство с визуальной средой Kodu. Возможности Kodu. Главное меню Kodu.

6. Знакомство с интерфейсом и основными принципами работы с программой KoduGameLab

Теория: Интерфейс программы Kodu. Возможности программы Kodu.. Главное меню программы. Режим просмотра проекта.

Практика: Практическая работа: «Знакомство с Kodu».

7. Создание ландшафтов (миров), добавление объектов. Создание персонажей с учетом игровой среды, действия персонажей

Практика: Исследование миров. Просмотр миров. Просмотр кода. Параметры команд. Объекты

Kodu. Импорт и редактирование объектов. «Мир в Kodu». Практическая работа: «Работа с объектами». Создание нового мира. Создание ландшафта. Создание ландшафта с помощью «Кисти земли». Практическая работа: «Работа с ландшафтом». Техника рисования. Цветокоррекция.

8. Перемещение персонажей в макросреде Kodu с использованием клавиатуры и мыши.

Практика: Управление объектами. Взаимодействие объектов.

9. Режим программирования, основные операторы Kodu

Практика: Разработка кода программы. Рассмотрение основных ошибок при написании кода программы. Написание кода программы.

10. Создание путей, выбор поведения персонажей. Создание клонов.

Практика: Создание клонов и порождаемых объектов. Практическая работа: «Создание нового объекта игры»

11. Страницы, функции, ракурс обзора.

Практика: Управление камерой игры. Изменение ракурса обзора. «Работа с камерой в Kodu». Программирование страниц. Практическая работа: «Работа со страницами».

12. Таймер, индикатор здоровья. Опция «Родитель»

Программирование таймера игры. Подсчет очков. Понятие таймера. Запуск таймера.

Практическая часть: Практическая работа: «Работа с таймером».

13. Создание нового игрового мира

Практика: Создание собственной 3д игры. Практическая работа: «Поедание яблок».

14. Планирование и создание собственного проекта.

Выбор темы, разработка алгоритма программы, написание программы. Отладка программы.

Подготовка презентации проекта. Подготовка выступления. Защита творческих проектов

Планируемые результаты освоения модуля продвинутого уровня сложности

«Основы алгоритмизации и программирования в программе Kodu game lab»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Познакомятся: с основными понятиями и свойствами алгоритмов.
- Научатся: использовать основные алгоритмические конструкции (линейные, разветвляющиеся, циклические).
- Рассмотрят: типы данных и операции с ними (простые и структурированные типы).
- Приобретут: навыки работы в визуальной среде программирования Kodu Game Lab.
- Научатся: создавать трехмерные игровые пространства, персонажей и объекты.
- Запомнят: последовательность программирования простых игровых сюжетов, обработки событий и движения объектов.
- Получат: опыт разработки и публичной защиты творческих проектов в области программирования игр.

Метапредметные результаты:

- Разовьют: навыки планирования и алгоритмического мышления.
- Научатся: структурировать задачу и подбирать подходящий алгоритм.
- Будут способны: организовать и контролировать процесс решения задачи.
- Выработают: привычку экспериментировать и рассматривать разные варианты решений.
- Демонстрируют: аккуратность и бережливость в обращении с оборудованием и материалами.

Личностные результаты:

- Научатся: преодолевать трудности и доводить начатые дела до конца.
- Разовьют: способность концентрировать внимание и долго сохранять сосредоточенность.
- Проявят интерес: к сферам, связанным с ИТ и развитием информационных технологий.
- Выработают: внутреннюю дисциплину и самостоятельность в учебной деятельности.
- Сформируют: умение ставить цели и идти к их достижению своими силами.

Модуль базового уровня сложности модуля углубленного уровня сложности

«Основы алгоритмизации и программирования в программе Scratch»

Модуль направлен на ознакомление с азами программирования и алгоритмизации через увлекательную среду Scratch. Курс способствует формированию интереса к современным информационным технологиям, развивает логическое и творческое мышление, стимулирует самостоятельность и умение доводить начатые дела до конца. Ребята знакомятся с созданием интерактивных историй, мультфильмов и игр, развивая одновременно навыки планирования, анализа и решения проблем, полезные в дальнейшем обучении и жизни.

Цель: развитие интереса и способностей, обучающихся к программированию и информационной культуре через погружение в игровую образовательную среду Scratch, формирующую навыки логического мышления, алгоритмической грамотности и креативного выражения.

Задачи:

Обучающие:

- Освоить основы программирования и алгоритмизации в игровой форме.
- Научиться пользоваться инструментарием среды Scratch для создания анимаций, игр и мультимедийных проектов.
- Узнать ключевые понятия программирования: сценарий, объект, событие, цикл, условие, переменная.
- Получить навыки конструирования простых линейных, циклических и условных алгоритмов.

Развивающие:

- Развить логическое и пространственное мышление.
- Развить творческие способности и воображение через создание оригинальных проектов.
- Стимулировать внимание, память и концентрацию при анализе условий задач и составлении алгоритмов.
- Способствовать развитию навыков планирования и пошагового решения поставленных задач.

Воспитательные:

- Сформировать положительное отношение к учебе и самостоятельной познавательной деятельности.
- Воспитывать чувство удовлетворения и гордости за достигнутые успехи.
- Способствовать развитию инициативности и интереса к активной учебно-познавательной деятельности.
- Способствовать: выработке навыков работы в коллективе, уважительному отношению к мнению товарищей и умению договариваться.

Учебно-тематический план модуля углубленного уровня сложности
«Основы алгоритмизации и программирования в программе Scratch»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Вводное занятие. Знакомство с интерфейсом Scratch.	2	1	1	Прослушивание
2.	Линейные алгоритмы.	4	2	2	Теория/ Практическая работа
3.	Работа с переменными.	6	2	4	Теория/ Практическая работа
4.	Условные алгоритмы.	6	2	4	Теория/ Практическая работа
5.	Циклические алгоритмы.	8	2	6	Теория/ Практическая работа
6.	Работа со списками.	4	1	3	Теория/ Практическая работа
7.	Создание подпрограмм.	2		2	Теория/ Практическая работа
8.	Планирование и создание собственного проекта. Защита проекта	4		4	Workshop/ Консультации
		36	10	26	

Содержание модуля углубленного уровня сложности

«Основы алгоритмизации и программирования в программе Scratch»

1. Вводное занятие. Знакомство с интерфейсом Scratch.

Теория: Вводный инструктаж. Ознакомление с оборудованием кабинета. Изучение основных элементов интерфейса среды Scratch. Основные понятия и определения среды: скрипт, спрайт, сцена и пр. Векторная и растровая графика.

Практика: Приёмы работы со спрайтами, приёмы работы с фоном, составление простых скриптов из различных блоков.

2. Линейные алгоритмы.

Теория: Понятие линейного алгоритма. Блок схема линейного алгоритма. Способы записи алгоритмов.

Практика: Построение и выполнение линейных алгоритмов. Работа с основными блоками в среде Scratch.

3. Работа с переменными

Теория: Назначение переменных. Создание переменных. Использование переменных для создания игр. Разработка плана игры по заданной теме. Создание программного кода для спрайтов.

Практика: Практическая работа по созданию игры с применением переменных. Использование переменных в ветвлении. Хранение счетчика действий в переменной.

4. Условные алгоритмы.

Теория: Условные операторы. Варианты ветвления программ. Виды сенсоров. Взаимодействие сенсоров и условных операторов.

Практика: Написание скриптов с ветвлением различной сложности. Вложенные ветвлении. Использование сенсоров управления движением.

5. Циклические алгоритмы.

Теория: Необходимость применения цикла. Виды циклов. Способы задержки выполнения скрипта.

Практика: Создание скриптов с применением циклов и задержек. Использование цикла с предусловием. Цикл с определенным количеством повторений. Задержка выполнения скрипта внутри цикла.

6. Работа со списками

Теория: Списки элементов. Команды управления списками. Создание списков. Доступ к элементам списка. Нумерационные списки.

Практика: Практическая работа по созданию проекта с использованием списков. Последовательный доступ к элементам списка. Поиск элемента в списке.

7. Создание подпрограмм.

Теория: Ознакомление с возможностью создания подпрограмм. Раздел «Другие блоки», создание блока, параметры блока.

Практика: Создание блоков-подпрограмм.

8. Планирование и создание собственного проекта.

Выбор темы, разработка алгоритма программы, написание программы. Отладка программы. Подготовка презентации проекта. Подготовка выступления. Защита творческих проектов

Планируемые результаты модуля углубленного уровня сложности «Основы алгоритмизации и программирования в программе Scratch»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Научатся: различать линейные, циклические и условные алгоритмы в среде Scratch.
- Сумеют: использовать инструменты графического редактора Scratch для создания костюмов и фонов в простых мультимедийных проектах.
- Изучат: команды разделов среды Scratch («Движение», «Внешность», «Звук», «Контроль», «Сенсоры», «Переменные») и научатся применять их для создания линейных, циклических и условных алгоритмов.
- Освоят: технику создания анимаций по заданному сюжету.
- Создадут: простые игры, используя интерактивные возможности программной среды Scratch.

Метапредметные результаты:

- Научатся: оценивать свои успехи и видеть прогресс в достижении учебных целей.
- Сформируют: устойчивый интерес к программированию и инженерным профессиям.
- Разовьют: навыки выделения главного и структурирования информации при чтении учебных материалов.

Личностные результаты:

- Разовьют: аналитическое и логическое мышление.
- Сформируют: способность ставить цели, планировать свою работу и следовать разработанному плану.
- Приобретут: навыки самостоятельности и самоорганизации и навыки командной работы.
- Овладеют: культурой речи и умением грамотно выражать свои мысли.

Модуль продвинутого уровня сложности

«Кодинг с нуля»

Продвинутый модуль базового года обучения построен таким образом, чтобы последовательно вводить обучающегося в основы программирования, развивая понимание базовых концепций и способность самостоятельно решать поставленные задачи.

Цель — формирование начальных компетенций, обучающихся в области программирования, развитие алгоритмического мышления и интереса к информационным технологиям через освоение основ языка Python и решение практических задач.

Задачи

Обучающие:

- Ознакомить обучающихся с основными компонентами компьютерной системы и общими принципами функционирования программ.
- Научить писать простые программы на языке Python, включая установку среды разработки и написание первых скриптов.
- Передать ученикам знание о структуре данных, переменных и операциях ввода-вывода.
- Дать представление о механизмах ветвления и циклов, научить эффективно управлять процессом выполнения программы.
- Привить опыт самостоятельного составления функций, улучшения читаемости и удобства поддержки программ.
- Предоставить возможность творчески реализоваться в проектах, применяя изученные инструменты и технологии.

Развивающие:

- Способствовать развитию мышления, способствующего быстрому поиску решений возникающих задач.
- Способствовать повышению способности анализировать ситуации и находить оптимальные пути выхода из затруднений.
- Улучшить внимание и память, позволяющие запоминать команды и приемы программирования.
- Расширить кругозор, представив широкие перспективы современных ИТ-технологий и цифровых инструментов.

Воспитательные:

- Формировать: устойчивое позитивное отношение к учебе и познавательной деятельности.
- Прививать: навыки ответственности и привычки завершать начатое дело.
- Укреплять: веру в собственные силы и повышать самооценку через успехи в программировании.
- Вдохновлять обучающихся на дальнейшие исследования в сфере информационных технологий и науки.

Учебно-тематический план модуля продвинутого уровня сложности
«Кодинг с нуля»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Вводное занятие. Начало путешествия в мир программирования	3	1	2	Теория/ Практическая работа
2.	Переменные и элементарные вычисления	6	4	2	Теория/ Практическая работа
3.	Структура принятия решения и булевая логика	6	2	4	Теория/ Практическая работа
4.	Структуры с повторением	8	1	7	Теория/ Практическая работа
5.	Функции	9	2	7	Теория/ Практическая работа
6.	Планирование и создание собственного проекта. Защита проекта	4		4	Workshop/ Консультации
		36	10	26	

Содержание модуля продвинутого уровня сложности

«Кодинг с нуля»

1. Вводное занятие. Начало путешествия в мир программирования.

Теория: Вводный инструктаж. Ознакомление с оборудованием кабинета. Компьютер и программы. Установка среды разработки

Практика: Первая программа на Python. Элементарные эксперименты с выводом текста и простых расчетов.

2. Переменные и элементарные вычисления

Теория: Переменные и присваивание значений. Введение в черепашью графику

Практика: Чуть подробнее об инструкции print. Решение задач по теме переменные

3. Структура принятия решения и булевая логика

Теория: Инструкция if. Инструкция if-else

Практика: Сравнение строковых значений. Вложенные структуры принятия решения и инструкции if-elif-else. Логические операторы. Черепашья графика: определение состояния черепахи. Решение задач по теме структура принятия решения.

4. Структуры с повторением

Теория: Введение в циклы.

Практика: Цикл while. Цикл for. Вычисление нарастающего итога. Вложенные циклы. Черепашья графика: применение циклов для рисования узоров. Решение задач по теме циклы.

5. Функции

Теория: Введение в функции.

Практика: Проектирование программ при помощи функции. Локальные переменные. Глобальные переменные. Написание функции с возрастом значения. Математический модуль math и хранение функции в модулях. Черепашья графика: модуляризация кода при помощи функции. Решение задач по теме функции.

Планируемые результаты освоения модуля продвинутого уровня сложности

«Кодинг с нуля»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Приобретут первоначальные знания о компьютерах, программировании и устройстве среды разработки.
- Умеют создать первые программы на языке Python, используя инструкции вывода и примитивные вычисления.
- Знают, как использовать переменные, присваивание и операцию печати результата.
- Умеют применять алгоритмы ветвления и циклов для обработки данных и логического контроля потоков выполнения.
- Интегрируют внешние модули (например, библиотека Turtle) в свои программы.
- Отлаживают созданный код, исправляют найденные ошибки и улучшают работоспособность программы.
- Демонстрируют готовность продолжить изучение программирования, став участником творческих конкурсов и соревнований.

Метапредметные результаты:

- Имеют навык анализа и синтеза при разработке программных решений.
- Владеют алгоритмизацией и способностью выстраивать последовательные шаги для решения проблем.
- Применяют различные методы тестирования и отладки своего кода.
- Разовьют способности работать в команде над совместными проектами.

Личностные результаты:

- Проявляют интерес к изучению информационных технологий и программистскому творчеству.
- Демонстрируют ответственное отношение к выполнению задания.
- Испытывают чувство удовлетворения от успешно выполненных проектов.
- Способны критически оценивать свою деятельность и стремиться к улучшению качества выполняемых работ.

Эти результаты позволяют учащимся уверенно ориентироваться в начальной теории программирования, грамотно строить вычислительные модели и решать практические задачи в учебной и проектной деятельности.

2 год обучения: Углубленный.

Второй углублённый год обучения по дополнительной образовательной программе развивает знания и навыки, полученные в первом году, и предлагает более серьёзные и сложные темы. На втором году обучающиеся начинают осваивать основы веб-разработки, изучая HTML и CSS, что даёт им возможность самостоятельно создавать простые веб-страницы. Далее идёт знакомство с мощным инструментом для разработки веб-приложений — фреймворком Django, позволяющим приступить к созданию полноценных веб-приложений. Наконец, в третьем разделе дети знакомятся с основами программирования микроконтроллеров Arduino, открывая для себя область «интернета вещей» и разработки устройств. Этот год помогает глубже понять и закрепить знания в области программирования и готовит обучающихся к ещё более сложному третьему году обучения.

Модуль базового уровня сложности «Основы веб-разработки: HTML и CSS»

Модуль базового уровня сложности «Основы веб-разработки: HTML и CSS» направлен на формирование профессиональных навыков начинающих разработчиков в области веб-разработки. Программа включает подробное изучение основ HTML и CSS, начиная с базовых тегов и заканчивая современными техниками компоновки страниц с использованием Flexbox и Grid. Особое внимание уделяется вопросам адаптации дизайна под разные экраны и проведению полноценной практики по верстке реальных макетов сайтов. Курс готовит участников к самостоятельному созданию качественных и функциональных веб-проектов.

Цель: формирование у обучающихся твердых теоретических знаний и практических навыков создания статических веб-ресурсов, включающее изучение правил разметки документов HTML, оформление внешнего вида элементов с помощью CSS, адаптацию дизайна под различные устройства и приобретение опыта самостоятельной верстки сайтов согласно предоставленным макетам.

Задачи:

Обучающие:

- Изучить основы языка гипертекстовой разметки HTML, структуру документа и использование основных тегов.
- Обучить созданию форм, таблиц и списков на страницах HTML.
- Познакомить с принципами работы каскадных таблиц стилей CSS, типы селекторов и стандартными свойства оформления.
- Освоить современные методы компоновки элементов веб-страниц с помощью Flexbox и Grid.
- Научить создавать адаптивные дизайны, подходящие для разных экранов устройств.
- Применить полученные знания на практике, выполнив полный цикл верстки сайта.

Развивающие:

- Развить аналитическое мышление и внимательность при составлении семантической структуры веб-страницы.
- Развить визуальное восприятие и дизайнерские навыки при выборе цветовых схем, шрифтов и общего стиля оформления сайта.
- Тренировать навыки командной работы и совместной деятельности при коллективных проектах.
- Улучшать навыки самоконтроля и самодисциплины при регулярной практической работе над сайтами.

Воспитательные:

- Воспитать ответственность за качественное исполнение проектов и соблюдение принятых стандартов веб-разработки.
- Привить уважение к чужой интеллектуальной собственности и соблюдению авторских прав при использовании сторонних изображений и ресурсов.
- Пробудить профессиональный интерес к развитию карьеры в сфере веб-разработки и поощрить стремление совершенствоваться в дальнейшем обучении.

Учебно-тематический план модуля базового уровня сложности

«Основы веб-разработки: HTML и CSS»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Введение в HTML: теги, структура страницы	2	2		Теория/ Практическая работа
2.	Формы, таблицы, списки в HTML	6	2	4	Теория/ Практическая работа
3.	Основы CSS: селекторы, свойства	6	2	4	Теория/ Практическая работа
4.	Flexbox и Grid	4	2	2	Теория/ Практическая работа
5.	Адаптивный дизайн	4	2	2	Теория/ Практическая работа
6.	Практикум: создание лендинга	8		8	Workshop/ Консультации
7.	Верстка макета сайта	6		6	Workshop/ Консультации
	Итого	36	10	26	

Содержание модуля базового уровня сложности
«Основы веб-разработки: HTML и CSS»

1. Введение в HTML: теги, структура страницы.

Теория: Вводный инструктаж. Структура HTML-документа. Теговая система разметки документов. Атрибуты тегов, семантика элементов.

2. Формы, таблицы, списки в HTML

Теория: Типы форм ввода данных. Таблицы: заголовки, строки, ячейки. Списки и навигационные элементы. Неупорядоченные и упорядоченные списки.

Практика: Оптимизация таблиц для мобильных устройств. Формирование формы регистрации Навигационное меню с использованием списков. Составление навигационного меню сайта

3. Основы CSS: селекторы, свойства

Теория: Селекторы классов и ID. Блочная модель CSS

Практика: Использование CSS для оформления текста и фона. Оформление дизайна веб-страниц с помощью CSS.

4. Flexbox и Grid.

Теория: Гибкое позиционирование блоков с помощью Flexbox.

Практика: Распределение пространства между элементами с помощью Grid. Применение flexbox и grid для построения адаптивной сетки страниц

5. Адаптивный дизайн

Теория: Принципы отзывчивого дизайна

Практика: Медиа-запросы и breakpoints. Проектирование адаптивного шаблона сайта

6. Практикум: создание лендинга

Сборка простых проектов: лендингов, портфолио, визиток. Инструменты тестирования кроссбраузерности и мобильности. Полноценная разработка простого лендинга.

7. Верстка макета сайта

Планируемые результаты освоения модуля базового уровня сложности

«Основы веб-разработки: HTML и CSS»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Научатся объяснять значение основных понятий и терминов HTML и CSS, отличать типы тегов и атрибутов.
- Умеют применять различные виды списков, таблиц и форм для создания структурированных веб-страниц.
- Смогут грамотно оформлять внешний вид элементов страницы с помощью CSS, использовать технологии Flexbox и Grid.
- Создают адаптивные интерфейсы с помощью медиазапросов и обеспечивать доступность сайтов на разных экранах.
- Разрабатывают небольшие веб-сайты и лэндинги с нуля.

Метапредметные результаты:

- Планируют собственную учебную деятельность, определять последовательность этапов выполнения проекта.
- Ищут и систематизируют необходимую информацию, осваивают дополнительные материалы для повышения качества разработок.
- Проводят самооценку выполненных работ, анализируют допущенные ошибки и предлагают пути их устранения.
- Эффективно взаимодействуют с командой разработчиков, совместно принимая решения и распределяя обязанности.
- Аргументированно обосновывают выбор тех или иных подходов и технологий в разработке.

Личностные результаты:

- Отвечают за своевременность и качество выполненных заданий, соблюдают установленные сроки.
- Ставятся активными участниками процесса обучения, проявляют инициативу и креативность при решении профессиональных задач.
- Испытывают интерес к повышению своей квалификации и осознанно подходят к выбору направлений дальнейшего развития.
- Воспринимают себя частью команды профессионалов, понимают важность сотрудничества и взаимоподдержки.

Освоив этот модуль, обучающиеся получат востребованные навыки и компетенции, необходимые для успешной карьеры в сфере веб-разработки.

Модуль углубленного уровня сложности

«Знакомство с Django»

Модуль углубленного уровня сложности «Знакомство с Django» предназначен для обучающихся, обладающих начальными знаниями Python и желающих освоить популярный фреймворк для быстрой и качественной разработки веб-приложений. В ходе модуля слушатели подробно изучат архитектуру Django, особенности моделей данных, системы аутентификации пользователей, механизм маршрутизации запросов и шаблонизацию. Большое внимание уделяется практике развертывания и поддержки полноценных веб-сервисов, решению распространенных задач и интеграции сторонних библиотек. Модуль нацелен на подготовку специалистов, способных эффективно применять Django в реальной профессиональной деятельности.

Цель: овладение обучающимися глубокими знаниями и практическими навыками разработки высокопроизводительных веб-приложений с использованием популярного фреймворка Django, подготовка квалифицированных специалистов, способных быстро создавать масштабируемые и надежные веб-продукты.

Обучающие:

- Ознакомить слушателей с особенностями архитектуры Django и спецификой его компонентов (модели, маршруты, представления, шаблоны);
- научить эффективному использованию ORM-функциональности Django для работы с базой данных;
- раскрыть принципы проектирования RESTful API и взаимодействие фронтенд- и бэкенд-компонентов;

Развивающие:

- Развить навыки проектирования и документирования web-приложений;
- способствовать улучшению понимания архитектуры современных веб-решений и best practices в разработке backend;
- углубить знания в области тестирования и деплоя Django-проектов;
- стимулировать развитие самостоятельности и ответственности за принятие технических решений.

Воспитательные:

- Сформировать ответственный подход к качеству кода и обеспечению надежности разрабатываемых продуктов.
- Развить навыки постановки стратегических целей и планирования проектов.
- Привить потребность в постоянном профессиональном росте и самосовершенствовании.
- Показать преимущества командной работы и регулярного обмена опытом внутри профессионального сообщества программистов.

Учебно-тематический план модуля углубленного уровня сложности
«Знакомство с Django»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Вводное занятие. Установка Django и создание проекта	4	2	2	Теория/ Практическая работа
2.	Модели (Models) и ORM	3	2	1	Теория/ Практическая работа
3.	Админ-панель Django	2		2	Теория/ Практическая работа
4.	Представления (Views) и URL-маршрутизация	4	2	2	Теория/ Практическая работа
5.	Шаблоны, контексты, передача данных	3	2	1	Теория/ Практическая работа
6.	Подключение статики и медиа	4	1	3	Теория/ Практическая работа
7.	Формы и валидация	4	1	3	Теория/ Практическая работа
8.	Практикум: создание блога	8		8	Workshop/ Консультации
9.	Деплой проекта на Heroku	4		4	Workshop/ Консультации
Итого		36	10	26	

Содержание модуля углубленного уровня сложности

«Знакомство с Django»

1. Вводное занятие. Установка Django и создание проекта

Теория: Вводный инструктаж. Основные возможности и сильные стороны Django. Компоненты Django-проекта: app, project, settings. Что такое виртуальное окружение и зачем оно нужно? Первая установка Django и запуск тестового проекта. Настройки среды разработки (settings.py) и старт локального сервера.

Практика: Создать виртуальную среду и установить Django. Подготовить первый минимальный проект Django. Ознакомиться с структурой стандартного Django-проекта. Запустить сервер разработки и проверить доступность страницы приветствия.

2. Модели (Models) и ORM

Теория: Объектно-реляционное отображение (ORM): концепция и цели. Создание моделей в Django: определение атрибутов и связей. Реляционные связи (ForeignKey, ManyToManyField, OneToOneField). Простое чтение и запись данных с помощью ORM. Операторы фильтров и операций с записями (фильтрация, агрегация, обновление).

Практика: Определить и создать простую модель для блога (например, Post, Author). Выполнить различные запросы к базе данных с применением ORM. Создать демонстрационный пример простого блога с созданием и просмотром записей.

3. Админ-панель Django

Теория: Как работает панель администратора в Django. Зачем нужна регистрация моделей в admin.py. Возможности кастомизации интерфейса (добавление списков, фильтраций, действий). Правильная организация интерфейса панели администратора.

Практика: Зарегистрировать созданные ранее модели в админ-панели. Настроить удобный интерфейс для добавления и изменения записей. Проверить работоспособность администраторской части и управление объектами.

4. Представления (Views) и URL-маршрутизация

Теория: Функциональные и классовые представления (CBV и FBV). Назначение URL и привязка к view. Маршруты URL: именованные и динамические. Примеры простых GET и POST запросов.

Практика: Написать функциональные представления для различных типов запросов. Использовать маршрутизацию URL для разных адресов. Продемонстрировать работу с динамическим сегментом URL.

5. Шаблоны, контексты, передача данных

Теория: Понятие шаблонов и шаблонизатора Django. Миграция данных в шаблоны (контекст). Конструкция шаблонов (переменные, циклы, условия). Инъекция CSS и JS в страницу через шаблоны.

Практика: Оформить домашнюю страницу и форму блога с помощью шаблонов. Программно передать переменные и списки данных в шаблоны. Сделать структуру страниц адаптивной с разделением блоков.

6. Подключение статики и медиа

Теория: Что такое static files и media files в Django. Пути к файлам стилей, изображений и другим ресурсам. Как правильно настроить хранение и обработку медийных файлов. Деплой статики и работа с хранилищами типа Amazon S3 или Cloudinary.

Практика: Организовать загрузку картинок и других мультимедийных материалов в блоге. Перенаправлять доступ к CSS файлам. Протестировать правильное отображение всех ресурсов на странице.

7. Формы и валидация

Теория: Создание форм в Django (ModelForm, Form). Правила валидации данных в формах. Генерация сообщений об ошибках и помощь пользователям. Пример использования форм для входа и регистрации.

Практика: Создать собственную форму обратной связи и систему комментирования. Провести проверку введённых данных (валидация email, пароля и др.). Отобразить правильные сообщения об успехе или ошибках.

8. Практикум: создание блога

Теория: Общий обзор процесса проектирования полноценного блога. Архитектура и модули блога: посты, комментарии, пользователи. Интерфейс административного раздела для управления содержимым. Дополнительные улучшения (SEO-метаданные, пагинация, RSS-каналы).

Практика: Спроектировать и реализовать простой блог-сайт с основным функционалом (создание поста, просмотр, комменты). Установить интеграцию с системой комментария Disqus или аналогичной. Завершить финальную версию блога с основными функциями и возможностями расширения.

9. Деплой проекта на Heroku

Теория: Какие существуют облачные платформы для развёртывания Django. Особенности выбора Heroku для деплоя. Настройка зависимостей и переменных окружения.

Практика: Подготовить проект для деплоя на Heroku. Опубликовать приложение на платформе и протестировать его доступность онлайн. Собрать отзывы и внести правки в код, исправляя найденные проблемы. Предоставить итоговую ссылку на размещённый сайт.

Планируемые результаты модуля углубленного уровня сложности

«Знакомство с Django»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Изучат основные концепции архитектуры Django и специфику его ключевых компонентов (модели, маршруты, представления, шаблоны).
- Овладеют методами эффективной работы с базами данных через встроенную ORM Django.
- Освоят принципы построения RESTful API и взаимодействия фронтенд- и бэкенд компонентов.
- Научатся проектировать и разрабатывать сложные многофункциональные веб-приложения с высоким уровнем производительности и надежности.

Метапредметные результаты:

- Разовьют способности к грамотному проектированию и документированию веб-приложений.
- Улучшат понимание современной архитектуры веб-разработки и усвоить лучшие практики backend-разработки.
- Приобретут опыт тестирования и развертывания Django-проектов в производственных условиях.
- Сформируют самостоятельный подход к техническим решениям и развьют чувства ответственности за свою работу.

Личностные результаты:

- Научатся чётко формулировать профессиональные цели и разрабатывать стратегию их достижения.
- Сформируют устойчивую привычку к постоянному развитию и профессиональному росту.
- Осознают ценность совместного труда и необходимость активного участия в профессиональных сообществах для обмена опытом и профессионального обогащения.

Модуль продвинутого уровня сложности

«Программирование микроконтроллеров Arduino»

Модуль продвинутого уровня сложности «Программирование микроконтроллеров Arduino» направлен на изучение основ программирования устройств Arduino, освоение аппаратных и программных аспектов взаимодействия с внешними устройствами, расширение навыков практической разработки электронных схем и прикладного программирования на языке C++. Модуль ориентирован на тех, кто обладает начальным знанием электроники и желает развить компетенции в сфере автоматизации процессов и создания умных устройств.

Цель — формирование у обучающихся уверенных навыков программирования микроконтроллеров Arduino, оперативных методов решения задач автоматизации и эффективных способов применения соответствующих технологий и оборудования в процессе создания собственных проектов.

Задачи

Обучающие:

- Познакомить: участников с архитектурами микроконтроллеров серии AVR и STM32, используемых в платах Arduino.
- Научить: писать программы на C++ для управления светодиодами, датчиками, двигателями и другими исполнительными устройствами.
- Показать: способы подключения внешних модулей и сенсоров, организацию обмена данными по последовательным каналам (UART, I²C, SPI).
- Объяснить: механизмы прерывания и таймеров, оптимизации производительности и энергоэффективности.

Развивающие:

- Способствовать: развитию инженерного мышления и аналитических навыков, необходимых для анализа технических требований и проектирования решений.
- Развитие: практических навыков сборки прототипов устройств и самостоятельной диагностики неисправностей.
- Укрепить: способность самостоятельно искать необходимую техническую документацию и разбираться в спецификациях оборудования.
- Расширить: кругозор обучающихся в области новых технологий и инновационных подходов к созданию умных устройств.

Воспитательные:

- Прививать: интерес к изучению информационных технологий и электроники, желание постоянно расширять технические познания.
- Развивать: ответственность за принятые инженерные решения и техническое качество

реализуемых проектов.

- Формировать: уважительное отношение к интеллектуальному труду коллег и авторов готовых решений, опубликованных в открытых источниках.
- Поощрять: творческий подход к выполнению заданий и экспериментальной проверке гипотез.

**Учебно-тематический план модуля продвинутого уровня сложности
«Программирование микроконтроллеров Arduino»**

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Вводное занятие. Основы платформы Arduino	4	2	2	Теория/ Практическая работа
2.	Взаимодействие с цифровыми сигналами	4	2	2	Теория/ Практическая работа
3.	Работа с аналоговыми сигналами	4	1	3	Теория/ Практическая работа
4.	Работа с выводом информации	4	1	3	Теория/ Практическая работа
5.	Прерывания и синхронизация	4	1	3	Теория/ Практическая работа
6.	Связь и коммуникации	4	1	3	Теория/ Практическая работа
7.	Энергосберегающие проекты	4	1	3	Теория/ Практическая работа
8.	Сетевые проекты и IoT	4	1	3	Теория/ Практическая работа
9.	Планирование и создание собственного проекта. Защита проекта	4		4	Workshop/ Консультации
	Итого	36	10	26	

Содержание модуля продвинутого уровня сложности «Программирование микроконтроллеров Arduino»

1. Основы платформы Arduino

Теория: Вводный инструктаж. Введение в платформу Arduino. История появления Arduino, сравнение версий плат, среда разработки IDE.

Практика: Язык программирования Arduino. Начало работы с Arduino IDE, синтаксис языка, первая программа ("Hello World"). Аппаратная платформа Arduino. Исследование аппаратной составляющей платы Arduino Uno/Nano/Micro, распиновка выводов, подключение питания.

2. Взаимодействие с цифровыми сигналами

Теория: Управление выводами (GPIO). Логика работы GPIO, режимы вывода, схемы подключений светодиодов и кнопок.

Практика: Цифровые датчики и сигнализация. Подключение цифровых датчиков (датчик движения PIR, кнопки, переключатели), получение сигнала. Чтение цифровых датчиков. Решение задач считывания состояния датчика, использование дебаунса для стабильного чтения сигналов.

3. Работа с аналоговыми сигналами

Теория: Аналоговые сигналы и A/D преобразователи. Понятие аналоговых сигналов, схема работы АЦП (Аналого-Цифрового Преобразования), ограничители напряжения.

Практика: Измерение аналоговых величин. Использование потенциометра, фоторезистора, термодатчика, измерение освещенности, температуры, влажности воздуха.

4. Работа с выводом информации

Теория: Работа с дисплеями и индикаторами. Типы экранов (LCD, OLED, LED-индикаторы), библиотека LiquidCrystal, протокол I²C/SPI.

Практика: Проекты с LCD-дисплеем. Подключение и вывод текста на дисплей, показ графической информации, изменение шрифтов и символов.

5. Прерывания и синхронизация

Теория: Таймеры и прерывания. Принципы работы таймеров, виды прерываний, алгоритм обработки событий.

Практика: Применение таймеров и прерываний. Реализация мигания светодиодом по таймеру, опрос клавиатуры, контроль события нажатия кнопки.

6. Связь и коммуникации

Теория: Последовательная связь (UART, I²C, SPI). Стандарт UART, шина I²C, протокол SPI, применение в Arduino.

Практика: Коммуникация с внешним оборудованием. Организация обмена данными с внешним устройством (сенсор, модуль ESP8266, датчик температуры DS18B20).

7. Энергосберегающие проекты

Теория: Питание и энергосбережение. Режимы пониженного энергопотребления, батареи и аккумуляторы, стабилизаторы напряжения.

Практика: Энергоэффективность и автономность. Сборка автономного устройства на батарейном питании, снижение потребления энергии в режиме ожидания.

8. Сетевые проекты и IoT

Теория: Wi-Fi, Bluetooth и Интернет вещей (IoT). Сети Ethernet/Wifi/Bluetooth, MQTT-протокол, безопасность сетей.

Практика: Проектирование устройств с сетевыми связями. Соединение Arduino с модулем ESP8266, отправка данных в облако, реализация IoT-приложения.

9. Планирование и создание собственного проекта.

Демонстрация разработанных устройств, отчет по выполненному проекту, устная защита.

Планируемые результаты модуля продвинутого уровня сложности

«Программирование микроконтроллеров Arduino»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

Будут знать: устройство и характеристики платформы Arduino, , методы обработки цифровых и аналоговых сигналов, протоколы связи (UART, I²C, SPI), принципы конструирования и программирования устройств.

- Будут **уметь**: читать и понимать электрическую схему, собирать схемы на макетной плате, програмировать Arduino, отлаживать программы, интегрировать внешние устройства и модули в свои проекты.
- Будут **владеть**: техникой безопасной эксплуатации электрических цепей, инструментами разработки (Arduino IDE), пониманием базовых понятий цифровой электроники и программирования.

Метапредметные результаты:

- Будут **знать**: правила проектирования и моделирования, этапы разработки электронного устройства, способы устранения проблем и отладки программ.
- Будут **уметь**: самостоятельно формулировать задачи и достигать поставленных целей, вести исследования и эксперименты, оценивать риски и последствия принимаемых решений.
- Будут **владеть**: навыками целеполагания, рефлексии, оценки собственного прогресса, способностью организовывать своё рабочее пространство и продуктивно взаимодействовать в команде.

Личностные результаты:

- Будут **знать**: этические нормы и стандарты поведения инженеров-электронщиков, влияние собственных решений на окружающую среду и общество.
- Будут **уметь**: мотивированно и ответственно относиться к своему образованию и профессиональной деятельности, проявлять инициативу и креативность в решении нестандартных задач.
- Будут **владеть**: компетенциями самоорганизации, самоконтроля, принятия решений и готовности развиваться профессионально дальше.

3 год обучения: Продвинутый.

Третий продвинутый год обучения по дополнительной образовательной программе предназначен для углублённого изучения программирования и разработки современных ИТ-решений. В течение года, обучающиеся значительно повышают уровень владения языком Python, изучая его продвинутые возможности и применение в реальных задачах. Затем переходят к объектно-ориентированному программированию, усваивая главные принципы и методы разработки сложного программного обеспечения. Третий модуль посвящён созданию умных устройств и систем «интернет вещей» (IoT), где школьники осваивают проектирование и программирование микроконтроллеров Arduino, создание и управление «умными» устройствами. Год заканчивается разработкой самостоятельных проектов, интегрирующих знания и навыки, полученные на протяжении всего периода обучения.

Модуль базового уровня сложности

«Python для продвинутых»

Модулю базового уровня сложности «Python для продвинутых» посвящена подробная работа с языком программирования Python, направленная на углубленное изучение возможностей языка и приобретение практических навыков для реализации комплексных проектов. Участники осваивают современные библиотеки и фреймворки, учатся обрабатывать большие объемы данных, проектировать многопользовательские приложения и заниматься высокоуровневым программированием. Особое внимание уделено вопросам оптимизации производительности, безопасности и сопровождения кода. Данный модуль позволяет учащимся перейти от уровня новичка к уверенному профессионалу, готовому к применению Python в реальных задачах бизнеса и науки.

Цель - формирование у обучающихся устойчивых навыков работы с основными структурами данных и операторами языка Python, необходимых для решения стандартных задач программирования и анализа данных.

Задачи модуля

Обучающие задачи:

- Научить: работе с основными структурами данных Python (списками, кортежами, словарями, множествами, строками) и возможностям их применения.
- Познакомить: с технологией файловой обработки, способами обработки исключений и методами работы с пакетами визуализации данных.
- Сформировать: навыки правильного составления программ с соблюдением правил оформления и читаемости кода.
- Передать: опыт создания собственных программных решений, основанных на

последовательности действий и инструкций.

Развивающие задачи:

- Развить: абстрактное и логическое мышление путем решения практических задач на уровне синтаксиса и семантики языка Python.
- Углубить: понимание внутренних механизмов работы интерпретатора Python и взаимосвязи основных элементов языка.
- Тренировать: способность ставить и решать конкретные задачи в рамках программирования, эффективно распределяя временные ресурсы.
- Поддержать: самостоятельное развитие и улучшение навыков программирования через решение дополнительных упражнений и заданий повышенной сложности.

Воспитательные задачи:

- Привить: аккуратность и внимательность к деталям при написании программ, формировать привычку регулярно проверять правильность своего кода.
- Помочь: обрести уверенность в своих силах и заинтересоваться миром программирования и анализа данных.
- Настроить: позитивное отношение к процессу обучения и привить вкус к постоянному повышению квалификации и профессиональному росту.
- Направить: на соблюдение норм академической честности и объективности в оценивании своих достижений и успехов окружающих.

Учебно-тематический план модуля базового уровня сложности
«Python для продвинутых»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Вводное занятие.	1	1		Теория/ Практическая работа
2.	Файлы и исключения	8	2	6	Теория/ Практическая работа
3.	Списки и кортежи	8	2	6	Теория/ Практическая работа
4.	Подробнее о строковых данных	8	2	6	Теория/ Практическая работа
5.	Словари и множества	8	2	6	Теория/ Практическая работа
6.	Планирование и создание собственного проекта. Защита проекта	3		3	Workshop/ Консультации
Итого		36	9	27	

Содержание модуля базового уровня сложности
«Python для продвинутых»

1. Вводное занятие.

Теория: Вводный инструктаж. Основные понятия и конструкции Python для работы с файлами, коллекциями и обработкой ошибок. Советы по правильной структуре и оформлению кода.

2. Файлы и исключения

Теория: Введение в файловый ввод и вывод.

Практика: Применение циклов для обработки файлов. Обработка записей. Исключение. Решение задач по теме файл.

3. Списки и кортежи

Теория: Последовательности и введение в список.

Практика: Работа со списками. Кортежи. Построение графиков с данными списков при помощи пакета matplotlib. Решение задач по теме списки.

4. Подробнее о строковых данных

Теория: Строковые данные. Базовые строковые операции.

Практика: Проверка, поиск и манипуляция строковыми данными. Решение задач по теме строковые данные.

5. Словари и множества

Теория: Словари. Множества.

Практика: Работа с словарями. Решение задач по теме словари и множества.

6. Планирование и создание собственного проекта. Защита проекта

Выбор темы, разработка алгоритма программы, написание программы. Отладка программы.

Подготовка презентации проекта. Подготовка выступления. Защита творческих проектов.

Планируемые результаты модуля базового уровня сложности

«Python для продвинутых»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Научатся: работать с основными структурами данных Python (списки, кортежи, словари, множества, строки) и применять их на практике.
- Освоят: технологию файловой обработки, обработку исключений и методы визуализации данных.
- Овладеют: правилами оформления и написания читаемого кода.
- Получат: опыт создания собственных программных решений, основанных на пошаговом подходе и соблюдении инструкций.

Метапредметные результаты:

- Разовьют: абстрактное и логическое мышление через решение практических задач на уровне синтаксиса и семантики языка Python.
- Углубят: понимание внутреннего устройства интерпретатора Python и взаимосвязи элементов языка.
- Разовьют: навыки постановки и решения задач в рамках программирования, эффективно распоряжаясь временем и ресурсами.
- Придут к выводу: о пользе самостоятельного развития и регулярных тренировок навыков программирования.

Личностные результаты:

- Выработают: привычку внимательно относиться к мелочам и регулярно проверять чистоту и функциональность своего кода.
- Сформируют: уверенность в своих силах и обретут устойчивый интерес к программированию и анализу данных.
- Будут стремиться: к постоянному профессиональному росту и развитию.
- Примут: принципы честности и объективности в отношении к собственному и чужому труду

Модуль углубленного уровня сложности

«Объектно-ориентированное программирование»

Модуль посвящен детальному изучению парадигмы объектно-ориентированного программирования (ООП) на примере языка Python. Обучающиеся освоят фундаментальные концепции ООП: классы, объекты, наследование, полиморфизм и инкапсуляцию. В процессе обучения рассматриваются паттерны проектирования, а также специальные методы и декораторы, позволяющие организовать эффективный процесс разработки сложных приложений. Основное внимание уделяется закреплению теории на практике через реализацию реального проекта с использованием ООП-подхода.

Цель модуля углубленного уровня сложности «Объектно-ориентированное программирование» — формирование глубокого понимания и увереных навыков объектно-ориентированной разработки на языке Python, позволяющих участникам эффективно проектировать, реализовывать и сопровождать сложные программные продукты, применяя принципы ООП.

Задачи:

Обучающие задачи:

- Познакомить: с основными принципами объектно-ориентированного программирования (ООП), выделить достоинства и недостатки такого подхода.
- Научить: создавать классы и объекты, правильно применять конструкторы и деструкторы.
- Исследовать: внутреннее устройство классов и принципы их работы.
- Освоить: механизм наследования, в частности, многословное наследование и проблему ромба.
- Покажем: суть полиморфизма и продемонстрируем, как его использовать на практике.
- Изучить: понятие рекурсии, алгоритмы с рекурсией и потенциальные опасности рекурсивных вызовов.
- Овладеть: практическими навыками создания и отладки программ, разработки алгоритмов и защите проектов.

Развивающие задачи:

- Развить: навыки логического и абстрактного мышления через создание и модификацию классов и объектов.
- Сформировать: понимание внутреннего устройства языка Python и взаимосвязи элементов языка.
- Стимулировать: самостоятельное развитие и желание решать задачи повышенной сложности.
- Работать: над повышением точности и выразительности программирования, углубляя понимание базовых и продвинутых концепций языка.

Воспитательные задачи:

- Воспитать: внимательность и аккуратность при написании программ, выработку привычки

проверять свой код и избегать ошибок.

- Способствовать в развитии: уверенности в своих силах и развить устойчивый интерес к программированию и программным проектам.
- Мотивировать: на дальнейшее развитие и совершенствование навыков программирования.
- Направить: к пониманию важности постоянного профессионального роста и своевременного повышения квалификации.

Учебно-тематический план модуля углубленного уровня сложности

«Объектно-ориентированное программирование»

№ п. п.	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Вводное занятие.	1	1		Теория/ Практическая работа
2.	Классы и объектно- ориентированное программирование	10	4	6	Теория/ Практическая работа
3.	Наследовательность	10	2	8	Теория/ Практическая работа
4.	Рекурсия	10	2	8	Теория/ Практическая работа
5.	Планирование и создание собственного проекта. Защита проекта	5		5	Workshop/ Консультации
Итого		36	9	27	

Содержание модуля углубленного уровня сложности

«Объектно-ориентированное программирование»

1. Вводное занятие.

Теория: Вводный инструктаж. Введение в объектно-ориентированное программирование (ООП). Преимущества и недостатки объектно-ориентированного подхода.

2. Классы и объектно-ориентированное программирование

Теория: Объектно-ориентированное программирование. Классы и объекты. Конструкторы и деструкторы. Внутренний механизм классов.

Практика: Классы. Работа с экземплярами. Приемы конструирования классов. Разбор задач на создание классов и экземпляров объектов. Реализация небольших программ с использованием классов и свойств.

3. Наследовательность

Теория: Введение в наследовательность. Введение в понятие наследования. Понятия родительского и дочернего классов. Переопределение методов и атрибутов. Особенности наследования в Python: множественное наследование, проблема ромба.

Практика: Практическая работа по созданию иерархии классов с использованием наследования. Полиморфизм. Проработка понятия полиморфизма: разные реализации методов в дочерних классах.

4. Рекурсия

Теория: Введение в рекурсию. Понятие рекурсивных функций. Алгоритмы с рекурсией: факториал, числа Фибоначчи, обход дерева. Потенциальные опасности рекурсии: ограничение глубины стека вызовов, опасность бесконечной рекурсии.

Практика: Решение задач на основе рекурсии.

5. Планирование и создание собственного проекта. Защита проекта

Выбор темы, разработка алгоритма программы, написание программы. Отладка программы. Подготовка презентации проекта. Подготовка выступления. Защита творческих проектов.

Планируемые результаты модуля углубленного уровня сложности

«Объектно-ориентированное программирование»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

Будут знать: основные концепции и принципы объектно-ориентированного программирования (ООП), такие как классы, объекты, наследование, инкапсуляция, полиморфизм, а также детали реализации этих принципов в языке Python.

- Будут **уметь**: создавать классы и объекты, определять поведение объектов, устанавливать взаимоотношения между классами (наследование, композиция), эффективно использовать механизм инкапсуляции и полиморфизма.
- Будут **владеть**: навыками разработки крупных проектов с использованием объектно-ориентированного подхода, умея выбирать подходящую структуру классов и интерфейсов.

Метапредметные результаты:

- **Разовьют**: навыки логического и алгоритмического мышления, необходимого для качественного проектирования сложных программных систем.
- **Сформируют**: способность самостоятельно изучать новые конструкции языка и эффективно применять их в разработке программ.
- **Овладеют**: методом анализа и синтеза информации, выделяя главное и существенное при решении задач программирования.

Личностные результаты:

- **Будут иметь**: уверенность в себе и своих силах при проектировании больших программных систем.
- **Нацелятся**: на постоянное развитие и пополнение багажа знаний в области объектно-ориентированного программирования.
- **Оценят**: роль программирования в современном мире и значимость хорошего образования для успешной карьеры в ИТ-индустрии.

Модуль продвинутого уровня сложности

«Умные устройства»

Модуль продвинутого уровня сложности «Умные устройства» предназначен для опытных пользователей, желающих глубоко погрузиться в мир IoT и интеллектуальных систем. В ходе курса участники детально изучат разработку и программирование умных устройств на базе набора «Матрешка», «Малина» и популярных платформ (Raspberry Pi, Arduino, ESP32). Будут рассмотрены технологии беспроводной связи (Wi-Fi, Bluetooth, ZigBee), программирование микроконтроллеров и создание проектов с удалённым управлением и аналитикой данных. Итогом станет полноценный проект умного устройства с возможностью интеграции в IoT-инфраструктуру и удалённого мониторинга.

Цель модуля продвинутого уровня сложности «Умные устройства» — формирование у слушателей устойчивого комплекса навыков, включающего проектирование, программирование и интеграцию интеллектуальных устройств на базе IoT-платформ (Raspberry Pi, Arduino, ESP32 и др.), что обеспечит возможность уверенно разрабатывать и эксплуатировать полномасштабные IoT-проекты.

Задачи:

Обучающие:

- Обучить: основам работы с платформами Raspberry Pi, Arduino, ESP32 и другими IoT-устройствами.
- Предоставить: навыки проектирования и программирования интеллектуальных устройств и систем.
- Научить: работе с беспроводными сетями и протоколами (Wi-Fi, Bluetooth, ZigBee, LoRaWAN).
- Овладеть: принципами построения и эксплуатации IoT-проектов с использованием облачных сервисов.

Развивающие:

- Сформировать: способность комплексного подхода к проектированию и реализации IoT-устройств.
- Способствовать развитию навыков критического мышления и аналитического подхода при выборе технологий и архитектуры проекта.
- Научить справляться: одновременно с вопросами железа и программного обеспечения.
- Стимулировать: творческую активность и самостоятельность в принятии решений при разработке устройств.

Воспитательные:

- Сформировать: ответственное отношение к технологиям будущего и тенденциям в отрасли IoT.
- Выработать: уверенность в своих силах и способность смело начинать и завершать самостоятельные проекты.
- Подчеркнуть: важность безопасного и этичного использования технологий умных устройств.
- Вдохновить: интересом к миру инновационных технологий и желанием исследовать новые горизонты в сфере интеллектуальной электроники.

Учебно-тематический план Модуля "Умные устройства"

№ п.п	Наименование разделов и тем	всего	в том числе		Формы аттестации контроля
			теория	практика	
1.	Введение в «Интернет вещей» и базовые концепции	6	2	4	Теория/ Практическая работа
2.	Микроконтроллеры и встраиваемые системы	6	2	4	Теория/ Практическая работа
3.	Работа с датчиками и исполнительными устройствами	6	2	4	Теория/ Практическая работа
4.	Беспроводные технологии и протоколы	6	2	4	Теория/ Практическая работа
5.	Визуализация и удалённый мониторинг	6	2	4	Теория/ Практическая работа
6.	Создание умных устройств	6	2	4	Workshop/ Консультации
		36	12	24	

Содержание модуля продвинутого уровня сложности

«Умные устройства»

1. Введение в «Интернет вещей» и базовые концепции

Теория: Введение в Интернет вещей (IoT): история, принципы работы, применение. Основные составляющие IoT-устройств: микроконтроллеры, датчики, исполнительные устройства. Основы безопасности и защиты персональных данных в IoT.

Практика: Подключение и инициализация датчиков (температурный, освещённости, звука и т.д.). Первичная работа с микроконтроллерами Arduino и Raspberry Pi. Запись и анализ значений датчиков в программах на Python и Arduino.

2. Микроконтроллеры и встраиваемые системы

Теория: Виды микроконтроллеров: Arduino, ESP32, Raspberry Pi. Внешние устройства и взаимодействие с ними (порты GPIO, ADC, PWM). Выбор подходящего микроконтроллера для конкретного проекта.

Практика: Установка и настройка среды разработки Arduino IDE. Простые программы для Arduino: мигалка, температурный монитор, кнопка-сторож. Первые опыты с Raspberry Pi: включение и выключение светодиода, управление нагрузкой.

3. Работа с датчиками и исполнительными устройствами

Теория: Классификация датчиков и исполнительных устройств: физические величины, точность измерений, потребление энергии. Универсальность датчиков: совместимость с разными микроконтроллерами. Ограничения и нюансы использования исполнительных устройств (моторы, реле, лампы и т.д.).

Практика: Сборка и калибровка различных датчиков (DHT11, MQ-2, TCS34725 и др.). Управление нагрузками (светодиоды, моторы, звуковые сигналы). Создание простого IoT-устройства на базе Arduino с датчиком температуры и влажностью.

4. Беспроводные технологии и протоколы

Теория: Основные протоколы передачи данных: Wi-Fi, Bluetooth, Zigbee, LoRaWAN. Каналы связи и уровни энергоснабжения. Локальные и глобальные сети для IoT-устройств.

Практика: Простой проект на Arduino с Wi-Fi-модулем ESP8266. Передача данных через Bluetooth с помощью ESP32. Защита каналов связи и шифрование данных.

5. Визуализация и удалённый мониторинг

Теория: Облачные сервисы для IoT: Blynk, ThingSpeak, Google Firebase. Возможность удалённого управления и мониторинга через смартфоны и ПК. Принципы визуализации данных и графики (Matplotlib, Plotly, Grafana).

Практика: Создание IoT-проекта с графиками и диаграммами на ThingSpeak. Внедрение облачного сервиса Blynk для удалённого управления своим IoT-устройством. Интерактивная демонстрация

работы с облачным сервисом (мониторинг и управление).

6. Создание умных устройств

Теория: Этапы проектирования IoT-устройств: от идеи до реализации. Эффективные подходы к проектированию и испытанию прототипов. Лучшие практики в разработке умных устройств.

Практика: Разработка и сборка небольшого проекта IoT-устройства (смарт-розетка, погодная станция, умный светильник и т.п.). Финальная защита проекта с презентацией устройства и описанием использованной технологии.

Планируемые результаты модуля углубленного уровня сложности

«Умные устройства»

По завершении обучения модуля, обучающиеся смогут продемонстрировать следующий набор знаний, умений и навыков:

Предметные результаты:

- Освоят основные принципы работы с платформами Raspberry Pi, Arduino, ESP32 и другими IoT-устройствами.
- Получат навыки проектирования и программирования интеллектуальных устройств и систем.
- Овладеют работой с беспроводными сетями и протоколами (Wi-Fi, Bluetooth, ZigBee, LoRaWAN).
- Освоят принципы построения и эксплуатации IoT-проектов с использованием облачных сервисов.

Метапредметные результаты:

- Сформируют способность к комплексному подходу при проектировании и реализации IoT-устройств.
- Разовьют навыки критического мышления и аналитического подхода при выборе технологий и архитектуры проекта.
- Научатся решать задачи, требующие совмещения аппаратной и программной составляющих.
- Придут к самостоятельности и инициативе в принятии решений при разработке устройств.

Личностные результаты:

- Воспитают ответственное отношение к технологиям будущего и актуальным тенденциям в отрасли IoT.
- Почувствуют уверенность в своих силах и способность смело приступать к самостоятельным проектам.
- Поймут важность безопасного и этичного использования технологий умных устройств.
- Найдут вдохновение в исследовании и развитии инновационных технологий в сфере интеллектуальной электроники.

Материально-техническое обеспечение программы

- Компьютерные рабочие станции (ПК или ноутбуки) с установленной операционной системой Windows/ «РЕД ОС» /Linux.
- Программа Kodu Game Lab для обучения основам алгоритмизации и программирования.
- Программы для начинающих программистов Scratch и подобные образовательные среды.
- Онлайн-ресурсы и учебники по основам программирования и алгоритмизации.
- Преподавательские материалы (видеоуроки, инструкции, презентации).
- Средства разработки и программирования на Python: редакторы кода (Python IDE, Visual Studio Code, PyCharm, Sublime Text), терминальная оболочка, интерпретатор Python.
- Web-сервер для локального тестирования Django-приложений (Apache, Nginx, WSGI-серверы).
- Пакеты и библиотеки Python (Django, Flask, Requests, BeautifulSoup, Selenium и др.).
- Электронные устройства и комплектующие для изучения Arduino (микроконтроллеры Arduino, датчики, провода, breadboards и др.).
- Руководства и документация по языкам программирования и инструментам разработки.
- Специализированные среды разработки для Python (Python IDE, PyCharm Pro, Jupyter Notebook, Anaconda Distribution).
- Серверные платформы для запуска Django и иных веб-приложений (Docker, Vagrant, Digital Ocean, AWS).
- Arduino и сопутствующие аксессуары для изучения программирования микроконтроллеров.
- Образовательные наборы для знакомства с IoT: AMP –S010 Матрешка - 5 шт., AMP –S039 Малина – 5 шт., AMP –S013 Образовательный набор «Амперка», Дополнительные модули для набора «Йодо» - 5 шт.
- Облачные сервисы для хранения и анализа данных.

Общесистемные требования:

- Высокоскоростной доступ в интернет для скачивания библиотек, обновления документов и онлайн-обучения.
- Экран для фронтального объяснения и демонстрации уроков (интерактивная доска или проектор).
- Оборудование для конференций и online-зачётов.
- Кабинет оборудован мебелью для комфортного обучения (рабочие столы, стулья, шкафы для хранения материалов).

Итоговое материально-техническое обеспечение включает:

- Ноутбук 15.6 НР – 9 шт с необходимым программным обеспечением.

- Соответствующие вспомогательные устройства: Проектор BenQ – 1 шт, МФУ HP LaserJet
- Документацию и руководство для преподавателей и обучающихся.
- Электронные пособия и методические материалы.

Это гарантирует высокое качество учебного процесса и достижение заявленных образовательных целей.

Организационно-педагогические условия реализации образовательной программы

Реализация образовательной программы обеспечивается наличием следующих организационно-педагогических условий:

Кадровый состав педагогов:

- Наличие педагогических работников, прошедших специальную подготовку по дополнительным программам повышения квалификации в области современного программирования и компьютерных наук.
- Педагоги обладают достаточным педагогическим стажем и профильным образованием, что подтверждено соответствующими квалификационными документами.
- Постоянное прохождение педагогами курсов повышения квалификации и профорганизации для поддержания высокого уровня знаний и мастерства.

Информационно-методическое сопровождение:

- Разработаны учебные планы, календарно-тематические планы, рабочие программы, диагностические материалы и дидактические материалы для каждой ступени обучения.
- Регулярное обновление содержания программ в соответствии с изменениями в технологиях и потребностями рынка труда.
- Предоставление дополнительного материала и ссылок на информационные ресурсы для самостоятельного изучения.

Материально-техническая база:

- Современное компьютерное оборудование, соответствующее требованиям программ обучения.
- Налаженная инфраструктура информационно-коммуникационных технологий (ИКТ), включая скоростной интернет и серверное оборудование.
- Наличие лицензионного программного обеспечения, соответствующего заявленным направлениям подготовки.
- Специальные лаборатории и кабинеты для проведения практических занятий, оснащённые необходимой аппаратурой и материалами (наборы Arduino, Raspberry Pi, микросхемы, датчики и т.д.).

Психолого-педагогическое сопровождение:

- Диагностика уровня знаний и мотивации обучающихся на каждом этапе обучения.
- Корректировка образовательного процесса в зависимости от успеваемости и потребностей обучающихся.
- Проведение консультаций и индивидуальных встреч с педагогами для оказания дополнительной поддержки.

Оценочно-диагностический инструментарий:

- Регулярные промежуточные аттестации и зачёты для определения текущего уровня освоения материала.
- Итоговая аттестация в форме экзамена или защиты проекта.
- Открытые мероприятия (конференции, выставки, хакатоны) для демонстрации достижений обучающихся.

Междисциплинарные связи:

- Организация совместных мероприятий с другими направлениями и специальностями для укрепления междисциплинарных связей и формирования широкого взгляда на сферу информационных технологий.
- Сотрудничество с вузами, компаниями-партнёрами и организациями, занимающимися продвижением ИТ-образования.

Эти организационно-педагогические условия обеспечивают качественную реализацию образовательной программы и позволяют достичь поставленных образовательных задач.

Календарный учебный график 1 года обучения

№ п/п	Дата проведения занятия	Время проведения занятия	Кол-во часов	Тема занятия	Форма занятия	Место проведения
1.	03.09.2025	16.00-17.30	2	Основные понятия алгоритмов.	Вводное занятие	«IT квантум» (каб.№2)
2.	05.09.2025	16.00-16.40	1	Основные алгоритмические конструкции	Теория/Практикум	«IT квантум» (каб.№2)
3.	10.09.2025	16.00-17.30	2	Основные алгоритмические конструкции/ Данные	Практикум/ Теория	«IT квантум» (каб.№2)
4.	12.09.2025	16.00-16.40	1	Данные	Теория	«IT квантум» (каб.№2)
5.	17.09.2025	16.00-17.30	2	Логические основы алгоритмизации	Теория/Практикум	«IT квантум» (каб.№2)
6.	19.09.2025	16.00-16.40	1	Программирование в среде KoduGameLab	Теория	«IT квантум» (каб.№2)
7.	24.09.2024	16.00-17.30	2	Программирование в среде KoduGameLab/ Знакомство с интерфейсом и основными принципами работы с программой KoduGameLab	Теория/Практикум	«IT квантум» (каб.№2)
8.	26.09.2025	16.00-16.40	1	Знакомство с интерфейсом и основными принципами работы с программой KoduGameLab	Практикум	«IT квантум» (каб.№2)
9.	01.10.2025	16.00-17.30	2	Создание ландшафтов (миров), добавление объектов. Создание персонажей с учетом игровой среды, действия персонажей	Практикум	«IT квантум» (каб.№2)
10.	03.10.2025	16.00-16.40	1	Создание ландшафтов (миров), добавление объектов. Создание персонажей с учетом игровой среды, действия персонажей	Практикум	«IT квантум» (каб.№2)
11.	08.10.2025	16.00-17.30	2	Создание ландшафтов (миров), добавление объектов. Создание персонажей с учетом игровой среды, действия персонажей Перемещение персонажей в макросреде Kodu с использованием клавиатуры и мыши.	Практикум	«IT квантум» (каб.№2)
12.	10.10.2025	16.00-16.40	1	Перемещение персонажей в	Практикум	«IT квантум» (каб.№2)

				макросреде Kodu с использованием клавиатуры и мыши.		
13.	15.10.2025	16.00-17.30	2	Режим программирования, основные операторы Kodu	Практикум	«IT квантум» (каб.№2)
14.	17.10.2025	16.00-16.40	1	Создание путей, выбор поведения персонажей. Создание клонов.	Практикум	«IT квантум» (каб.№2)
15.	22.10.2025,	16.00-17.30	2	Создание путей, выбор поведения персонажей. Создание клонов Страницы, функции, ракурс обзора.	Практикум	«IT квантум» (каб.№2)
16.	24.10.2025	16.00-16.40	1	Страницы, функции, ракурс обзора.	Практикум	«IT квантум» (каб.№2)
17.	29.10.2025	16.00-17.30	2	Таймер, индикатор здоровья. Опция «Родитель»	Практикум	«IT квантум» (каб.№2)
18.	31.10.2025	16.00-16.40	1	Создание нового игрового мира	Workshop /консультации	«IT квантум» (каб.№2)
19.	05.11.2025	16.00-17.30	2	Создание нового игрового мира	Workshop /консультации	«IT квантум» (каб.№2)
20.	07.11.2025	16.00-16.40	1	Создание нового игрового мира	Workshop /консультации	«IT квантум» (каб.№2)
21.	12.11.2025	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop /консультации	«IT квантум» (каб.№2)
22.	14.11.2025	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop /консультации	«IT квантум» (каб.№2)
23.	19.11.2025	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop /консультации	«IT квантум» (каб.№2)
24.	21.11.2025	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop /консультации	«IT квантум» (каб.№2)
25.	26.11.2025	16.00-17.30	2	Вводное занятие. Знакомство с интерфейсом Scratch.	Вводное занятие	«IT квантум» (каб.№2)
26.	28.11.2025	16.00-16.40	1	Линейные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
27.	03.12.2025	16.00-17.30	2	Линейные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
28.	05.12.2025	16.00-16.40	1	Линейные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
29.	10.12.2025	16.00-17.30	2	Работа с переменными.	Теория/ Практикум	«IT квантум» (каб.№2)
30.	12.12.2025	16.00-16.40	1	Работа с переменными.	Теория/ Практикум	«IT квантум» (каб.№2)

31.	17.12.2025	16.00-17.30	2	Работа с переменными.	Теория/ Практикум	«IT квантум» (каб.№2)
32.	19.12.2025	16.00-16.40	1	Работа с переменными.	Теория/ Практикум	«IT квантум» (каб.№2)
33.	24.12.2025	16.00-17.30	2	Условные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
34.	26.12.2025	16.00-16.40	1	Условные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
35.	31.12.2025	16.00-17.30	2	Условные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
36.	09.01.2026	16.00-16.40	1	Условные алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
37.	14.01.2026	16.00-17.30	2	Циклические алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
38.	16.01.2026	16.00-16.40	1	Циклические алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
39.	21.01.2026	16.00-17.30	2	Циклические алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
40.	23.01.2026	16.00-16.40	1	Циклические алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
41.	28.01.2026	16.00-17.30	2	Циклические алгоритмы.	Теория/ Практикум	«IT квантум» (каб.№2)
42.	30.01.2026	16.00-16.40	1	Работа со списками.	Теория/ Практикум	«IT квантум» (каб.№2)
43.	04.02.2026	16.00-17.30	2	Работа со списками.	Теория/ Практикум	«IT квантум» (каб.№2)
44.	06.02.2026	16.00-16.40	1	Работа со списками.	Теория/ Практикум	«IT квантум» (каб.№2)
45.	11.02.2026	16.00-17.30	2	Создание подпрограмм.	Практикум	«IT квантум» (каб.№2)
46.	13.02.2026	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop/ консультации	«IT квантум» (каб.№2)
47.	18.02.2026	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
48.	20.02.2026	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop/ консультации	«IT квантум» (каб.№2)
49.	25.02.2026	16.00-17.30	2	Вводное занятие. Начало путешествия в мир программирования	Теория/ Практикум	«IT квантум» (каб.№2)
50.	27.02.2026	16.00-16.40	1	Вводное занятие. Начало путешествия в мир программирования	Теория/ Практикум	«IT квантум» (каб.№2)
51.	04.03.2026	16.00-17.30	2	Переменные и элементарные вычисления	Теория/ Практикум	«IT квантум» (каб.№2)
52.	06.03.2026	16.00-16.40	1	Переменные и элементарные вычисления	Теория/ Практикум	«IT квантум» (каб.№2)
53.	11.03.2026	16.00-17.30	2	Переменные и элементарные	Теория/ Практикум	«IT квантум» (каб.№2)

				вычисления		
54.	13.03.2026	16.00-16.40	1	Переменные и элементарные вычисления	Теория/ Практикум	«IT квантум» (каб.№2)
55.	18.03.2026	16.00-17.30	2	Структура принятия решения и булевая логика	Теория/ Практикум	«IT квантум» (каб.№2)
56.	20.03.2026	16.00-16.40	1	Структура принятия решения и булевая логика	Теория/ Практикум	«IT квантум» (каб.№2)
57.	25.03.2026	16.00-17.30	2	Структура принятия решения и булевая логика	Теория/ Практикум	«IT квантум» (каб.№2)
58.	27.03.2026	16.00-16.40	1	Структура принятия решения и булевая логика	Теория/ Практикум	«IT квантум» (каб.№2)
59.	01.04.2026	16.00-17.30	2	Структуры с повторением	Теория/ Практикум	«IT квантум» (каб.№2)
60.	03.04.2026	16.00-16.40	1	Структуры с повторением	Теория/ Практикум	«IT квантум» (каб.№2)
61.	08.04.2026	16.00-17.30	2	Структуры с повторением	Теория/ Практикум	«IT квантум» (каб.№2)
62.	10.04.2026	16.00-16.40	1	Структуры с повторением	Теория/ Практикум	«IT квантум» (каб.№2)
63.	15.04.2026	16.00-17.30	2	Структуры с повторением	Теория/ Практикум	«IT квантум» (каб.№2)
64.	17.04.2026	16.00-16.40	1	Функции	Теория/ Практикум	«IT квантум» (каб.№2)
65.	22.04.2026	16.00-17.30	2	Функции	Теория/ Практикум	«IT квантум» (каб.№2)
66.	24.04.2026	16.00-16.40	1	Функции	Теория/ Практикум	«IT квантум» (каб.№2)
67.	29.04.2026	16.00-17.30	2	Функции	Теория/ Практикум	«IT квантум» (каб.№2)
68.	06.05.2026	16.00-16.40	1	Функции	Теория/ Практикум	«IT квантум» (каб.№2)
69.	08.05.2026	16.00-17.30	2	Функции	Теория/ Практикум	«IT квантум» (каб.№2)
70.	13.05.2026	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
71.	15.05.2026	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
72.	20.05.2026	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)

Календарный учебный график 2 года обучения

№ п/п	Дата проведения занятия	Время проведения занятия	Кол-во часов	Тема занятия	Форма занятия	Место проведения
1.	09.09.2026	16.00-17.30	2	Введение в HTML: теги, структура страницы	Теория/Практикум	«IT квантум» (каб.№2)
2.	11.09.2026	16.00-16.40	1	Формы, таблицы, списки в HTML	Теория/Практикум	«IT квантум» (каб.№2)
3.	16.09.2026	16.00-17.30	2	Формы, таблицы, списки в HTML	Теория/Практикум	«IT квантум» (каб.№2)
4.	18.09.2026	16.00-16.40	1	Формы, таблицы, списки в HTML	Теория/Практикум	«IT квантум» (каб.№2)
5.	23.09.2026	16.00-17.30	2	Формы, таблицы, списки в HTML	Теория/Практикум	«IT квантум» (каб.№2)
6.	25.09.2026	16.00-16.40	1	Основы CSS: селекторы, свойства	Теория/Практикум	«IT квантум» (каб.№2)
7.	30.09.2026	16.00-17.30	2	Основы CSS: селекторы, свойства	Теория/Практикум	«IT квантум» (каб.№2)
8.	02.10.2026	16.00-16.40	1	Основы CSS: селекторы, свойства	Теория/Практикум	«IT квантум» (каб.№2)
9.	07.10.2026	16.00-17.30	2	Основы CSS: селекторы, свойства	Теория/Практикум	«IT квантум» (каб.№2)
10.	09.10.2026	16.00-16.40	1	Flexbox и Grid	Теория/Практикум	«IT квантум» (каб.№2)
11.	14.10.2026	16.00-17.30	2	Flexbox и Grid	Теория/Практикум	«IT квантум» (каб.№2)
12.	16.10.2026	16.00-16.40	1	Flexbox и Grid	Теория/Практикум	«IT квантум» (каб.№2)
13.	21.10.2026	16.00-17.30	2	Адаптивный дизайн	Теория/Практикум	«IT квантум» (каб.№2)
14.	23.10.2026	16.00-16.40	1	Адаптивный дизайн	Теория/Практикум	«IT квантум» (каб.№2)
15.	28.10.2026	16.00-17.30	2	Адаптивный дизайн Практикум: создание лендинга	Workshop / консультации	«IT квантум» (каб.№2)
16.	30.10.2026	16.00-16.40	1	Практикум: создание лендинга	Workshop / консультации	«IT квантум» (каб.№2)
17.	04.11.2026	16.00-17.30	2	Практикум: создание лендинга	Workshop / консультации	«IT квантум» (каб.№2)
18.	06.11.2026	16.00-16.40	1	Практикум: создание лендинга	Workshop / консультации	«IT квантум» (каб.№2)
19.	11.11.2026	16.00-17.30	2	Практикум: создание лендинга	Workshop / консультации	«IT квантум» (каб.№2)
20.	13.11.2026	16.00-16.40	1	Практикум: создание лендинга	Workshop / консультации	«IT квантум» (каб.№2)
21.	18.11.2026	16.00-17.30	2	Верстка макета сайта	Workshop / консультации	«IT квантум» (каб.№2)
22.	20.11.2026	16.00-16.40	1	Верстка макета сайта	Workshop / консультации	«IT квантум» (каб.№2)
23.	25.11.2026	16.00-17.30	2	Верстка макета сайта	Workshop / консультации	«IT квантум» (каб.№2)
24.	27.11.2026	16.00-16.40	1	Верстка макета сайта	Workshop / консультации	«IT квантум» (каб.№2)
25.	02.12.2026	16.00-17.30	2	Вводное занятие. Установка Django и	Теория/Практикум	«IT квантум» (каб.№2)

				создание проекта		
26.	04.12.2026	16.00-16.40	1	Вводное занятие. Установка Django и создание проекта	Теория/ Практикум	«IT квантум» (каб.№2)
27.	09.12.2026	16.00-17.30	2	Вводное занятие. Установка Django и создание проекта. Модели (Models) и ORM	Теория/ Практикум	«IT квантум» (каб.№2)
28.	11.12.2026	16.00-16.40	1	Модели (Models) и ORM	Теория/ Практикум	«IT квантум» (каб.№2)
29.	16.12.2026	16.00-17.30	2	Модели (Models) и ORM. Админ-панель Django	Теория/ Практикум	«IT квантум» (каб.№2)
30.	18.12.2026	16.00-16.40	1	Админ-панель Django	Теория/ Практикум	«IT квантум» (каб.№2)
31.	23.12.2026	16.00-17.30	2	Представления (Views) и URL-маршрутизация	Теория/ Практикум	«IT квантум» (каб.№2)
32.	25.12.2026	16.00-16.40	1	Представления (Views) и URL-маршрутизация	Теория/ Практикум	«IT квантум» (каб.№2)
33.	30.12.2026	16.00-17.30	2	Представления (Views) и URL-маршрутизация. Шаблоны, контексты, передача данных	Теория/ Практикум	«IT квантум» (каб.№2)
34.	13.01.2027	16.00-16.40	1	Шаблоны, контексты, передача данных	Теория/ Практикум	«IT квантум» (каб.№2)
35.	15.01.2027	16.00-17.30	2	Шаблоны, контексты, передача данных. Подключение статики и медиа	Теория/ Практикум	«IT квантум» (каб.№2)
36.	20.01.2027	16.00-16.40	1	Подключение статики и медиа	Теория/ Практикум	«IT квантум» (каб.№2)
37.	22.01.2027	16.00-17.30	2	Подключение статики и медиа	Теория/ Практикум	«IT квантум» (каб.№2)
38.	27.01.2027	16.00-16.40	1	Формы и валидация	Теория/ Практикум	«IT квантум» (каб.№2)
39.	29.01.2027	16.00-17.30	2	Формы и валидация	Теория/ Практикум	«IT квантум» (каб.№2)
40.	03.02.2027	16.00-16.40	1	Формы и валидация	Теория/ Практикум	«IT квантум» (каб.№2)
41.	05.02.2027	16.00-17.30	2	Практикум: создание блога	Workshop / консультации	«IT квантум» (каб.№2)
42.	10.02.2027	16.00-16.40	1	Практикум: создание блога	Workshop / консультации	«IT квантум» (каб.№2)
43.	12.02.2027	16.00-17.30	2	Практикум: создание блога	Workshop / консультации	«IT квантум» (каб.№2)
44.	17.02.2027	16.00-16.40	1	Практикум: создание блога	Workshop / консультации	«IT квантум» (каб.№2)
45.	19.02.2027	16.00-17.30	2	Практикум: создание блога	Workshop / консультации	«IT квантум» (каб.№2)
46.	24.02.2027	16.00-16.40	1	Деплой проекта на Heroku	Workshop / консультации	«IT квантум» (каб.№2)
47.	26.02.2027	16.00-17.30	2	Деплой проекта на Heroku	Workshop / консультации	«IT квантум» (каб.№2)
48.	03.03.2027	16.00-16.40	1	Деплой проекта на Heroku	Workshop / консультации	«IT квантум» (каб.№2)

49.	05.03.2027	16.00-17.30	2	Вводное занятие. Основы платформы Arduino	Теория/ Практикум	«IT квантум» (каб.№2)
50.	10.03.2027	16.00-16.40	1	Вводное занятие. Основы платформы Arduino	Теория/ Практикум	«IT квантум» (каб.№2)
51.	12.03.2027	16.00-17.30	2	Вводное занятие. Основы платформы Arduino. Взаимодействие с цифровыми сигналами	Теория/ Практикум	«IT квантум» (каб.№2)
52.	17.03.2027	16.00-16.40	1	Взаимодействие с цифровыми сигналами	Теория/ Практикум	«IT квантум» (каб.№2)
53.	19.03.2027	16.00-17.30	2	Взаимодействие с цифровыми сигналами	Теория/ Практикум	«IT квантум» (каб.№2)
54.	24.03.2027	16.00-16.40	1	Работа с аналоговыми сигналами	Теория/ Практикум	«IT квантум» (каб.№2)
55.	26.03.2027	16.00-17.30	2	Работа с аналоговыми сигналами	Теория/ Практикум	«IT квантум» (каб.№2)
56.	31.03.2027	16.00-16.40	1	Работа с аналоговыми сигналами	Теория/ Практикум	«IT квантум» (каб.№2)
57.	02.04.2027	16.00-17.30	2	Работа с выводом информации	Теория/ Практикум	«IT квантум» (каб.№2)
58.	07.04.2027	16.00-16.40	1	Работа с выводом информации	Теория/ Практикум	«IT квантум» (каб.№2)
59.	09.04.2027	16.00-17.30	2	Работа с выводом информации Прерывания и синхронизация	Теория/ Практикум	«IT квантум» (каб.№2)
60.	14.04.2027	16.00-16.40	1	Прерывания и синхронизация	Теория/ Практикум	«IT квантум» (каб.№2)
61.	16.04.2027	16.00-17.30	2	Прерывания и синхронизация	Теория/ Практикум	«IT квантум» (каб.№2)
62.	21.04.2027	16.00-16.40	1	Связь и коммуникации	Теория/ Практикум	«IT квантум» (каб.№2)
63.	23.04.2027	16.00-17.30	2	Связь и коммуникации	Теория/ Практикум	«IT квантум» (каб.№2)
64.	28.04.2027	16.00-16.40	1	Связь и коммуникации	Теория/ Практикум	«IT квантум» (каб.№2)
65.	30.04.2027	16.00-17.30	2	Энергосберегающие проекты	Теория/ Практикум	«IT квантум» (каб.№2)
66.	05.05.2027	16.00-16.40	1	Энергосберегающие проекты	Теория/ Практикум	«IT квантум» (каб.№2)
67.	07.05.2027	16.00-17.30	2	Энергосберегающие проекты. Сетевые проекты и IoT	Теория/ Практикум	«IT квантум» (каб.№2)
68.	12.05.2027	16.00-16.40	1	Сетевые проекты и IoT	Теория/ Практикум	«IT квантум» (каб.№2)
69.	14.05.2027	16.00-17.30	2	Сетевые проекты и IoT	Теория/ Практикум	«IT квантум» (каб.№2)
70.	19.05.2027	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
71.	21.05.2027	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
72.	26.05.2027	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)

Календарный учебный график 3 года обучения

№ п/п	Дата проведения занятия	Время проведения занятия	Кол-во часов	Тема занятия	Форма занятия	Место проведения
1.	08.09.2027	16.00-16.40	1	Вводное занятие.	Теория	«IT квантум» (каб.№2)
2.	10.09.2027	16.00-17.30	2	Файлы и исключения	Теория/Практикум	«IT квантум» (каб.№2)
3.	15.09.2027	16.00-16.40	1	Файлы и исключения	Теория/Практикум	«IT квантум» (каб.№2)
4.	17.09.2027	16.00-17.30	2	Файлы и исключения	Теория/Практикум	«IT квантум» (каб.№2)
5.	22.09.2027	16.00-16.40	1	Файлы и исключения	Теория/Практикум	«IT квантум» (каб.№2)
6.	24.09.2027	16.00-17.30	2	Файлы и исключения	Теория/Практикум	«IT квантум» (каб.№2)
7.	29.09.2027	16.00-16.40	1	Списки и кортежи	Теория/Практикум	«IT квантум» (каб.№2)
8.	01.10.2027	16.00-17.30	2	Списки и кортежи	Теория/Практикум	«IT квантум» (каб.№2)
9.	06.10.2027	16.00-16.40	1	Списки и кортежи	Теория/Практикум	«IT квантум» (каб.№2)
10.	08.10.2027	16.00-17.30	2	Списки и кортежи	Теория/Практикум	«IT квантум» (каб.№2)
11.	13.10.2027	16.00-16.40	1	Списки и кортежи	Теория/Практикум	«IT квантум» (каб.№2)
12.	15.10.2027	16.00-17.30	2	Списки и кортежи Подробнее о строковых данных	Теория/Практикум	«IT квантум» (каб.№2)
13.	20.10.2027	16.00-16.40	1	Подробнее о строковых данных	Теория/Практикум	«IT квантум» (каб.№2)
14.	22.10.2027	16.00-17.30	2	Подробнее о строковых данных	Теория/Практикум	«IT квантум» (каб.№2)
15.	27.10.2027	16.00-16.40	1	Подробнее о строковых данных	Теория/Практикум	«IT квантум» (каб.№2)
16.	29.10.2027	16.00-17.30	2	Подробнее о строковых данных	Теория/Практикум	«IT квантум» (каб.№2)
17.	03.11.2027	16.00-16.40	1	Подробнее о строковых данных	Теория/Практикум	«IT квантум» (каб.№2)
18.	05.11.2027	16.00-17.30	2	Словари и множества	Теория/Практикум	«IT квантум» (каб.№2)
19.	10.11.2027	16.00-16.40	1	Словари и множества	Теория/Практикум	«IT квантум» (каб.№2)
20.	12.11.2027	16.00-17.30	2	Словари и множества	Теория/Практикум	«IT квантум» (каб.№2)
21.	17.11.2027	16.00-16.40	1	Словари и множества	Теория/Практикум	«IT квантум» (каб.№2)
22.	19.11.2027	16.00-17.30	2	Словари и множества	Теория/Практикум	«IT квантум» (каб.№2)
23.	24.11.2027	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
24.	26.11.2027	16.00-17.30	2	Планирование и создание собственного	Workshop / консультации	«IT квантум» (каб.№2)

				проекта. Защита проекта		
25.	01.12.2027	16.00-16.40	1	Вводное занятие.	Теория	«IT квантум» (каб.№2)
26.	03.12.2027	16.00-17.30	2	Классы и объектно-ориентированное программирование	Теория/ Практикум	«IT квантум» (каб.№2)
27.	08.12.2027	16.00-16.40	1	Классы и объектно-ориентированное программирование	Теория/ Практикум	«IT квантум» (каб.№2)
28.	10.12.2027	16.00-17.30	2	Классы и объектно-ориентированное программирование	Теория/ Практикум	«IT квантум» (каб.№2)
29.	15.12.2027	16.00-16.40	1	Классы и объектно-ориентированное программирование	Теория/ Практикум	«IT квантум» (каб.№2)
30.	17.12.2027	16.00-17.30	2	Классы и объектно-ориентированное программирование	Теория/ Практикум	«IT квантум» (каб.№2)
31.	22..12.2027	16.00-16.40	1	Классы и объектно-ориентированное программирование	Теория/ Практикум	«IT квантум» (каб.№2)
32.	24.12.2027	16.00-17.30	2	Классы и объектно-ориентированное программирование. Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
33.	29.12.2027	16.00-16.40	1	Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
34.	31.12.2027	16.00-17.30	2	Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
35.	12.01.2028	16.00-16.40	1	Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
36.	14.01.2028	16.00-17.30	2	Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
37.	19.01.2028	16.00-16.40	1	Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
38.	21.01.2028	16.00-17.30	2	Наследовательность	Теория/ Практикум	«IT квантум» (каб.№2)
39.	26.01.2028	16.00-16.40	1	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
40.	28.01.2028	16.00-17.30	2	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
41.	02.02.2028	16.00-16.40	1	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
42.	04.02.2028	16.00-17.30	2	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
43.	09.02.2028	16.00-16.40	1	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
44.	11.02.2028	16.00-17.30	2	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
45.	16.02.2028	16.00-16.40	1	Рекурсия	Теория/ Практикум	«IT квантум» (каб.№2)
46.	18.02.2028	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)

47.	25.02.2028	16.00-16.40	1	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
48.	01.03.2028	16.00-17.30	2	Планирование и создание собственного проекта. Защита проекта	Workshop / консультации	«IT квантум» (каб.№2)
49.	03.03.2028	16.00-16.40	1	Введение в «Интернет вещей» и базовые концепции	Теория/ Практикум	«IT квантум» (каб.№2)
50.	10.03.2028	16.00-17.30	2	Введение в «Интернет вещей» и базовые концепции	Теория/ Практикум	«IT квантум» (каб.№2)
51.	15.03.2028	16.00-16.40	1	Введение в «Интернет вещей» и базовые концепции	Теория/ Практикум	«IT квантум» (каб.№2)
52.	17.03.2028	16.00-17.30	2	Введение в «Интернет вещей» и базовые концепции	Теория/ Практикум	«IT квантум» (каб.№2)
53.	22.03.2028	16.00-16.40	1	Микроконтроллеры и встраиваемые системы	Теория/ Практикум	«IT квантум» (каб.№2)
54.	24.03.2028	16.00-17.30	2	Микроконтроллеры и встраиваемые системы	Теория/ Практикум	«IT квантум» (каб.№2)
55.	29.03.2028	16.00-16.40	1	Микроконтроллеры и встраиваемые системы	Теория/ Практикум	«IT квантум» (каб.№2)
56.	31.03.2028	16.00-17.30	2	Микроконтроллеры и встраиваемые системы	Теория/ Практикум	«IT квантум» (каб.№2)
57.	05.04.2028	16.00-16.40	1	Работа с датчиками и исполнительными устройствами	Теория/ Практикум	«IT квантум» (каб.№2)
58.	07.04.2028	16.00-17.30	2	Работа с датчиками и исполнительными устройствами	Теория/ Практикум	«IT квантум» (каб.№2)
59.	12.04.2028	16.00-16.40	1	Работа с датчиками и исполнительными устройствами	Теория/ Практикум	«IT квантум» (каб.№2)
60.	14.04.2028	16.00-17.30	2	Работа с датчиками и исполнительными устройствами	Теория/ Практикум	«IT квантум» (каб.№2)
61.	19.04.2028	16.00-16.40	1	Беспроводные технологии и протоколы	Теория/ Практикум	«IT квантум» (каб.№2)
62.	21.04.2028	16.00-17.30	2	Беспроводные технологии и протоколы	Теория/ Практикум	«IT квантум» (каб.№2)
63.	26.04.2028	16.00-16.40	1	Беспроводные технологии и протоколы	Теория/ Практикум	«IT квантум» (каб.№2)
64.	28.04.2028	16.00-17.30	2	Беспроводные технологии и протоколы	Теория/ Практикум	«IT квантум» (каб.№2)
65.	03.05.2028	16.00-16.40	1	Визуализация и удалённый мониторинг	Теория/ Практикум	«IT квантум» (каб.№2)
66.	05.05.2028	16.00-17.30	2	Визуализация и удалённый мониторинг	Теория/ Практикум	«IT квантум» (каб.№2)
67.	10.05.2028	16.00-16.40	1	Визуализация и удалённый мониторинг	Теория/ Практикум	«IT квантум» (каб.№2)
68.	12.05.2028	16.00-17.30	2	Визуализация и удалённый мониторинг	Теория/ Практикум	«IT квантум» (каб.№2)
69.	17.05.2028	16.00-16.40	1	Создание умных устройств	Workshop / консультации	«IT квантум» (каб.№2)

70.	19.05.2028	16.00-17.30	2	Создание умных устройств	Workshop / консультации	«IT квантум» (каб.№2)
71.	24.05.2028	16.00-16.40	1	Создание умных устройств	Workshop / консультации	«IT квантум» (каб.№2)
72.	26.05.2028	16.00-17.30	2	Создание умных устройств	Workshop / консультации	«IT квантум» (каб.№2)

План воспитательной работы

Программа ДООП технической направленности «ИТ квантум» реализуется в детском мини-технопарке «Квантум» в рамках национального проекта «Образование». Все мероприятия, проводимые в объединении нацелены на работу и реализацию социально значимых проектов в детском мини-технопарке.

№п/п	Мероприятия	Срок выполнения	Воспитательный компонент
1. Обеспечение функционирования объединения			
1.1.	Организация мероприятий для учащихся в каникулярный период	ноябрь, январь, март	развитие у учащихся интереса к современным технологиям и техническим профессиям, формирование позитивного отношения к STEM-образованию, воспитание инициативы и предприимчивости, подготовка к осознанному выбору будущей профессии.
1.2.	Организация участия учащихся в профильных сменах летнего детского отдыха	июнь-июль	воспитание технологического мышления, поддержка мотивации к изучению ИТ и инженерии, формирование навыков работы в команде и общения в профессиональной среде, развитие лидерских качеств и умения презентовать свои проекты.
1.3.	Организация и ведение работы в объединениях	В течение учебного года	развитие навыков проектирования и технического творчества, воспитание ответственности за результат своей работы, развитие инженерного мышления и творческих способностей, формирование убеждённости в необходимости постоянных знаний и профессионального роста.
1.4.	Организация мероприятий внутри объединения	В течение учебного года	формирование интереса к инновационным технологиям, развитие навыков работы с техническими устройствами и программами, воспитание самостоятельности и ответственности за принятые решения, поддержка инициативности и интереса к исследованию новых технологий.
1.5.	Проведение родительского собрания в объединениях	Октябрь-январь	привлечение родителей к поддержке технического образования детей, разъяснение преимуществ изучения ИТ-направленности, формирование партнёрских отношений между семьей и организацией дополнительного образования детей в вопросах воспитания и развития ребёнка, ориентация семей на поддержку интереса к техническим наукам и профессиям.
2. Обеспечение участия учащихся ОУ и жителей города в работе мини-технопарка			
2.1.	Проведение презентаций детского мини-технопарка на муниципальных площадках.	Сентябрь-октябрь	Развитие интереса к научно-техническому творчеству и инновациям: мероприятия направлены на популяризацию технических профессий и науки, привлекают молодёжь к исследовательской и проектной деятельности, развивают любопытство и желание узнавать новое. Патриотическое воспитание: организация тематических мероприятий, посвящённых Великой Победе, праздникам учителей и
2.2.	Проведение обзорных экскурсий по детскому мини-технопарку «Квантум»	В течении года	
2.3.	Проведение дней	В течении	

	открытых дверей детского мини-технопарка и мастер-классов в квантумах «Что мы можем?»	года	женскому дню, укрепляет чувство гордости за нашу страну и вызывает интерес к истории.
2.4.	Инженерные каникулы для детей ОВЗ: мастер-классы в квантумах «Вместе мы можем».	Декабрь	Интеграция детей с особыми нуждами: специальная программа «Вместе мы можем» создает равные условия для детей с ограниченными возможностями здоровья, содействует их социальному и творческому развитию.
2.5.	Онлайн конкурс "С Днем учителя"	Сентябрь-октябрь	Формирование ответственности и самостоятельности: через активное участие в конкурсах и мастер-классах дети учатся планировать, брать на себя ответственность за результат, публично демонстрировать свои достижения.
2.6.	Областной дистанционный конкурс технических проектов "В гостях у новогодней сказки"	Декабрь	Развитие коммуникативных навыков и командной работы: мероприятия предполагают активное взаимодействие между участниками, развивают навыки сотрудничества, коммуникации и общественного взаимодействия.
2.7.	Областной дистанционный конкурс технических проектов, посвящённый Великой Победе, «Никто не забыт! Ничто не забыто!»	Май	Привлечение широкой аудитории к деятельности мини-технопарка: мероприятия включают разнообразные формы участия, доступные жителям города и представителям учреждений образования, что способствует привлечению большего числа молодых талантов в сферу технических наук и инноваций.
2.8.	Онлайн-викторина «Мир современных технологий»	Январь	
2.9.	Городской онлайн-конкурс интерактивных открыток к 8 марта "Кванто-открытка"	Март	

3.Участие обучающихся мини-технопарка «Квантум» в мероприятиях различного уровня

3.1.	Конкурс изобретений «ТехноМир» в рамках областного проекта Марафон «Академия технического творчества»	Октябрь-декабрь	Развитие инженерного мышления и интереса к науке: участие в конкурсах и мероприятиях формирует устойчивый интерес к техническим наукам и профессиям, развивает навыки проектного мышления и творческого подхода к решению задач.
3.2	Участие в областном конкурсе "Новогодний кванто-сувенир", «Квантоелка»	Декабрь	Патриотическое воспитание: участие в тематических конкурсах и выставках, посвящённых космической тематике, формирует чувство гордости за достижения российской науки и техники, способствует развитию патриотизма и уважения к историческому наследию.
3.4.	Участие в региональном этапе всероссийского конкурса Шустрик	Апрель-май	Формирование навыков командной работы и лидерства: большинство мероприятий предполагает командную работу, что способствует развитию навыков общения, координации и выработки коллективных решений.
3.5.	Участие в областном конкурсе «Космическая техника и технологии»	Февраль	
3.6.	Участие в областном конкурсе «Путь к звездам»	Февраль	Самореализация и мотивация: участие в конкурсах и победа на них положительно влияет на самооценку, мотивирует на дальнейший рост и развитие, способствует осознанному выбору будущей профессии.
3.7.	Участие в областном конкурсе «Золотое сечение»	Март-май	Профессиональная ориентация: посещение различных площадок и участие в мероприятиях
3.8.	Участие в областном конкурсе фотографии городской среды "Прогулка"	Май	

3.9.	Участие в Региональном этапе Всероссийского робототехнического фестиваля «РобоФест- Приволжье»	Февраль	<p>позволяет школьникам ближе ознакомиться с различными областями науки и техники, принять осознанное решение относительно выбора будущей профессии.</p> <p>Социальная адаптация и творческое развитие: мероприятия способствуют расширению круга знакомств, развитию творческих способностей и формированию навыков публичного выступления и презентации своих проектов.</p> <p>Таким образом, воспитательный компонент направлен на всестороннее развитие личности, стимулирование познавательной активности и подготовку молодёжи к успешному трудуустройству в сфере высоких технологий.</p> <p>4. Организация мероприятий внутри мини-технопарка «Квантум»</p>
3.11.	Региональный молодежный проект для обучающихся "ТехноFun"	Март	
3.13	Участие в областном Хакатоне "Умный дом"	Ноябрь	
3.14	Участие в областном Хакатоне ИТ	Январь	
4.1.	Разработка положения и проведение квест - игры «Напряги извилины» между командами мини-технопарка	февраль	<ul style="list-style-type: none"> ○ Развитие критического и логического мышления: через квест-игры и круглый стол участники учатся анализировать информацию, делать выводы и принимать взвешенные решения.
4.5.	«Круглый стол»– «Моя игра», для обучающихся ИТ – квантума детского мини-технопарка «Квантум»	ноябрь	<ul style="list-style-type: none"> ○ Формирование навыков командной работы и коммуникации: мероприятия предусматривают коллективное взаимодействие, совместное решение задач и обмен идеями, что способствует развитию навыков общения и сотрудничества.
4.7.	Трех недельный марафон мастер-классов, которые проводятся воспитанниками мини-технопарка «Школа наставничества»	Апрель	<ul style="list-style-type: none"> ○ Саморазвитие и инициатива: активное участие в мастер-классах и квестах стимулирует интерес к новым знаниям, развивает творческое мышление и инициативу. ○ Мотивация к научно-техническому творчеству: мероприятие призвано вызвать интерес к научно-технической деятельности, познакомить участников с современными технологиями и помочь выявить их способности и таланты. ○ Повышение самооценки и уверенности в себе: успешное участие в мероприятиях, особенно выступление на «круглых столах» и марафонах, помогает укрепить уверенность в своих силах и способствует развитию лидерских качеств. ○ Социальная адаптация: активная вовлеченность в жизнь мини-технопарка способствует формированию позитивных социальных контактов, облегчает социализацию и адаптацию в обществе. <p>Таким образом, мероприятия внутри мини-технопарка служат целям развития личностных качеств, стимулирования интереса к науке и технике, а также формирования необходимых компетенций для успешного взросления и профессионального роста.</p>

Список литературы

1. А. Хрусталёв, А. Кириченко — «HTMLS + CSS3. Основы современного WEB-дизайна», «Наука и Техника», 2018, 352 стр..
2. А. С. Путиной, под ред. В. В. Тарапаты. «Scratch 2.0: от новичка к продвинутому пользователю. Пособие для подготовки к Scratch-Олимпиаде» М.: Лаборатория знаний, 2019.
3. Бейдер Дэн, Эймос Дэвид, Яблонски Джоанна, Хейслер Флетчер. Знакомство с Python. — СПб.: Питер, 2023. — 512 с.. atsp-life.org
4. Бешенков С. А., Е. А. Ракитина, Моделирование и формализация. Методическое пособие. — М.: Лаборатория Базовых Знаний, 2002. [урок.rph](http://urok.rph)
5. Бешенков С.А., Е.А. Ракитина, Моделирование и формализация. Методическое пособие/— М.: Лаборатория Базовых Знаний, 2002.
6. Бизли Дэвид. Python. Исчерпывающее руководство. — СПб.: Питер, 2023. — 368 с.. atsp-life.org
7. Блум Джереми «Изучаем Arduino: инструменты и методы технического волшебства», СПб.: БХВ-Петербург, 2015.
8. Быков В. Г. «Введение в компьютерное моделирование управляемых механических систем. От маятника к роботу» — СПб: Наука, 2011 — 418 с..
9. В. В. Тарапаты, Б. В. Прокофьева. «Учимся вместе со Scratch. Программирование, игры, робототехника», М.: Лаборатория знаний, 2019.
10. Гаско Рик. Простой Python просто с нуля [Электронный ресурс] / Рик Гаско. — М.: СОЛОН-Пресс, 2019. ciur.ru
11. Дауни Аллен Б. Основы Python. Научитесь думать как программист / Аллен Б. Дауни ; пер. с англ. С. Черникова ; [науч. ред. А. Родионов]. — Москва : Манн, Иванов и Фербер, 2021. — 304 с.. atsp-life.org
12. Дженифер Нидерст Роббинс — «HTML5. Карманный справочник», «Вильямс», 2016, 192 стр..
13. Джереми Блум, «Изучаем Arduino. Инструменты и методы технического волшебства»,— СПб.: БХВ-Петербург, 2015 — 336 с..
14. Джекф Форсье, Пол Биссекс, Уэсли Дж. Чанэ «Django. Разработка веб-приложений на Python».
15. Джон Дакетт — «HTML и CSS. Разработка и дизайн веб-сайтов», «Эксмо», 2017, 480 стр..
16. Документация по AVR микроконтроллерам /Сообщество EasyElectronics.ru.— [Электронное издание]— URL: <http://we.easylelectronics.ru/AVR/dokumentaciya-po-avr-mikrokontrolleramvse-na-russkom.html>

17. Дэвид Макфарланд — «Новая большая книга CSS», «Питер», 2016, 720 стр..
18. Златопольский Д. М. Основы программирования на языке Python. — М.: ДМК Пресс, 2017. — 284 с.: ил.. multiurok.ru
19. Златопольский, Д. М. Сборник задач по программированию [Электронный ресурс] / Д. М. Златопольский. — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2011.
20. Зыков, С. В. Программирование. Объектно-ориентированный подход: учебник и практикум для академического бакалавриата.— М.: Издательство Юрайт, 2017. — 155 с. — (Серия: Бакалавр. Академический курс). — ISBN 978-5-534-00850-0.
21. Изучении основ алгоритмизации и программирования в программе Kodu Game Lab
22. Казакова Д. А. Развитие алгоритмического мышления у младших школьников на уроках информатики // Современные научные исследования и разработки. 2018. Т. 1. № 4 (21). cyberleninka.ru
23. Казанский, А. А. Программирование на visual c# 2013: учебное пособие для прикладного бакалавриата/А. А. Казанский. — М.: Издательство Юрайт, 2017. — 191 с. — (Серия: Бакалавр. Прикладной курс). — ISBN 978-5-534-00592-9.
24. Карвинен Т., Киммо Карвинен, Вилле Валтокари, пер. с англ. и ред. И. В. Василенко, «Делаем сенсоры. Проекты сенсорных устройств на базе ARDUINO и RASPBERRY PI», Москва: Вильямс, 2015.
25. Кит Грант — «CSS для профи», «Питер», 2019, 496 стр.
26. Колодезникова А. А., Курилкина В. Н. Визуальный язык 3D-программирования Kodu Game Lab как средство формирования основ алгоритмического мышления в 5 классе общеобразовательной школы // Проблемы современного образования. 2023. №6. cyberleninka.ru
27. Костюк А. М., Ярова С. С. Развитие алгоритмического мышления у дошкольников и учащихся начальной школы на занятиях по программированию и робототехнике // Информатика в школе. cyberleninka.ru
28. Крис Минник, Эд Титтел — «HTML5 и CSS3 для чайников», «Диалектика», 2016, 400 стр..
29. Лия Веру — «Секреты CSS. Идеальные решения ежедневных задач», «Питер», 2017, 336 стр..
30. Мангина Ю. А., Рямов И. С., Старкова Л. Н. Формирование алгоритмического мышления у школьников посредством визуального языка программирования KODU // Актуальные вопросы преподавания математики, информатики и информационных технологий. Екатеринбург, 2016. cyberleninka.ru
31. Мюллер Джон Пол. Python для чайников [Электронный ресурс] / Джон Пол Мюллер. — 2-е изд.: Пер с англ. — СПб.: ООО «Диалектика», 2019. ciur.ru

- 32.** Обзор систем умный дом: функции, устройства и советы по выбору /журнал «Идеи вашего дома» Flash [Электронное издание]—URL: <https://www.ivd.ru/stroitelstvo-iremont/bezopasnost-i-domasnaa-avtomatika/obzor-sistem-umnyj-dom-funkcii-ustrojstva-i-sovety-po-vyboru-25771>
- 33.** Окулов С. М. Основы программирования: учебное пособие [Электронный ресурс] / С.М. Окулов. — 10-е изд., электрон. — Москва : Лаборатория знаний, 2020. [ciur.ru](#)
- 34.** Петин В. А., «Проекты с использованием контроллера Arduino», СПб.: БХВ-Петербург, 2014.
- 35.** Прохоренок Н. А. Python 3. Самое необходимое [Электронный ресурс] / Н. А. Прохоренок, В. А. Дронов. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2019. [ciur.ru](#)
- 36.** Прохоренок, Н. А. Python 3. Самое необходимое [Электронный ресурс] / Н. А. Прохоренок, В. А. Дронов. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2019.
- 37.** Пэйн, Брайсон. Python для детей и родителей [Электронный ресурс] / Брайсон Пейн [пер. с англ. М. А. Райтмана]. — М.: Издательство «Э», 2017.
- 38.** Руководство пользователя отладочного комплекса Амперка-Матрешка, Малина
- 39.** Саймон Монк «Программируем Arduino. Профессиональная работа со скетчами», СПб.: Питер, 2017.
- 40.** Свейгарт Эл. Учим Python, делая крутые игры [Электронный ресурс] / Эл. Свейгарт. — М: Эксмо, 2018. [ciur.ru](#)
- 41.** Седер Наоми. Python. Экспресс-курс [Электронный ресурс] / Наоми Седер. — 3-е изд. — СПб.: Питер, 2019.
- 42.** Селезнев, В. А. Компьютерная графика: учебник и практикум для академического бакалавриата / В. А. Селезнев, С. А. Дмитренко. — 2-е изд., испр. и доп. — М.: Издательство Юрайт, 2017. — 228 с. — (Серия: Бакалавр. Академический курс). — ISBN 978-5-534-01464-8.
- 43.** Соммер У., «Программирование микроконтроллерных плат Arduino/Freeduino», 2-е изд., СПб.: БХВ-Петербург, 2016.
- 44.** Столяров, А. В. Оформление программного кода [Электронный ресурс] / А. В. Столяров. — 2-е изд., испр. и доп. — Москва: МАКС Пресс, 2019.
- 45.** Страуструп Бьерн. Программирование. Принципы и практика с использованием C++, М.: Вильямс, 2016. — 1328 с.
- 46.** Т. Е. Сорокиной, А. Ю. Босовой, под ред. Л. Л. Босовой, «Информатика. 5–6 класс: Практикум по программированию в среде Scratch» М.: БИНОМ. Лаборатория знаний, 2019.
- 47.** Тero Карвинен, Киммо Карвинен, Вилле Валтокари, Вильямс, «Делаем сенсоры. Проекты сенсорных устройств на базе Arduino и Raspberry Pi», 2015 — 448 с..

- 48.** В. В. Трофимов, Т. А. Павловская; под ред. В. В. Трофимова. Алгоритмизация и программирование: учебник для академического бакалавриата. — М.: Издательство Юрайт, 2017. — 137 с. — (Серия: Бакалавр. Академический курс. Модуль.). — ISBN 978-5-9916-9866-5.
- 49.** Тузовский, А. Ф. Объектно-ориентированное программирование: учебное пособие для прикладного бакалавриата / А. Ф. Тузовский. — М.: Издательство Юрайт, 2017. — 206 с. — (Серия: Университеты России). — ISBN 978-5-534-00849-4.
- 50.** У. Соммер «Программирование микроконтроллерных плат Arduino/Freeduino»,— СПб.: БХВ-Петербург, 2012 — 256 с..
- 51.** Устройства для систем «Умный дом» /страница сайта Flash . – [Электронное издание] – URL: <https://lifecontrol.ru/devices/> (для знакомства с техническими характеристиками и принципом действия различных устройств: умная лампа, центральное устройство, датчик движения, датчик качества воздуха, умная розетка, датчик дыма, датчик открытия-закрытия, датчик протечки воды и др.)
- 52.** Фалина И., Богомолова Т., Большаякова Е., Гущин И., Шухардина В. «Алгоритмизация и программирование» — М.: Кудиц-Пресс, 2007 — 337 с..
- 53.** Шень, А. Программирование: теоремы и задачи [Электронный ресурс] / А. Шень. — 6-е изд., дополненное. — М.: МЦНМО, 2017.
- 54.** Шуман, Х. Г. Python для детей [Электронный ресурс] / Х. Г. Шуман [пер. с нем. М. А. Райтмана]. — М.: ДМК Пресс, 2019.
- 55.** Эрик А. Майер — «CSS. Карманный справочник», «Вильямс», 2017, 288 стр..
- 56.** Эстель Вейл — «HTML5: Разработка приложений для мобильных устройств», 2015, 480 стр..

Приложение 1
«Входной контроль»

Цель: проведение **диагностирующего** среза для определения исходного уровня знаний...

Задачи:

1. Выявить уровень общей компьютерной грамотности и опыта работы с цифровыми устройствами.
2. Определить наличие предварительного опыта в программировании и знакомство со средами разработки.
3. Изучить познавательные интересы и ожидания обучающихся в ИТ-сфере для повышения мотивации.

Анкета для входящего тестирования обучающегося

Дата заполнения: _____

ФИО обучающегося: _____

Возраст: _____

Школа, класс: _____

Блок 1. Общая компьютерная грамотность

1. Как часто ты используешь компьютер/ноутбук?
 - Каждый день
 - Несколько раз в неделю
 - Редко
 - Почти не использую
2. Для чего ты обычно используешь компьютер? (можно выбрать несколько)
 - Учёба
 - Игры
 - Социальные сети
 - Просмотр видео/фильмов
 - Работа с документами (Word, Excel, PowerPoint)
 - Другое: _____
3. Умеешь ли ты работать с файлами и папками?
 - Да, уверенно
 - Немного

- Нет

Блок 2. Опыт в программировании

4. Знаком(а) ли ты с понятием «программирование»?
 - Да, пробовал(а) писать код
 - Знаю в общих чертах
 - Нет, не знаю
5. С какими программами/средами программирования ты уже работал(а)?
 - Scratch
 - Kodu Game Lab
 - Python
 - HTML/CSS
 - Arduino
 - Другое: _____
6. Если пробовал(а) программировать, что ты создавал(а)?
 - Простые игры
 - Мультфильмы/анимации
 - Веб-сайты
 - Роботов/устройства
 - Другое: _____

Блок 3. Интересы и ожидания

7. Что тебе интереснее всего в IT? (можно выбрать несколько)
 - Создание игр
 - Веб-дизайн и сайты
 - Программирование роботов
 - Мобильные приложения
 - Графика и анимация
 - Защита информации
 - Другое: _____
8. Что ты хочешь научиться делать на занятиях?

9. Как ты узнал(а) о программе «IT-квантум»?

- От родителей
- От друзей
- В школе
- В интернете
- Другое: _____

Спасибо за ответы!

Приложение 2.
«Текущий контроль I»

Карта наблюдения педагога (для текущего контроля)

Цель: фиксирование динамики развития метапредметных и личностных компетенций в ходе каждого занятия.

Задачи:

1. Отслеживать активность, включенность и прогресс в освоении практических навыков.
2. Оценивать качество взаимодействия в группе и способность к сотрудничеству.
3. Собирать данные для оперативной обратной связи и корректировки методов работы с конкретным обучающимся.

Дата	ФИО обучающегося	Активность на занятии (вопросы, ответы)	Кач-во выполнения практич. задания	Взаимодействие в группе	Примечания

Приложение 3.
«Текущий контроль 2»

**Пример практической работы для текущего контроля
(Модуль «Кодинг с нуля. Python»)**

Цель: Проверить уровень усвоения конкретного блока теоретических знаний и умение применять их для решения стандартной практической задачи.

Задачи:

1. Провести объективную оценку понимания ключевых алгоритмических конструкций (циклы, условия, переменные)
2. Отработать навык пошагового решения задачи и написания корректного кода
3. Выявить типичные ошибки для их последующего разбора и устранения

Тема: Работа с циклами и условными операторами

Время выполнения: 20 минут

Задание: Напишите программу, которая:

1. Запрашивает у пользователя число.
2. Проверяет, является ли число положительным, отрицательным или нулём.
3. Если число положительное, выводит на экран все числа от 1 до этого числа.

Критерии оценки:

- Корректный ввод данных – 1 балл
- Использование условного оператора if-elif-else – 2 балла
- Использование цикла for – 2 балла
- Правильный вывод результата – 1 балл

Приложение 4.
«Промежуточный контроль»

Тест для промежуточного контроля

(Модуль «Основы алгоритмизации в Scratch»)

Цель: комплексное оценивание полученных в рамках модуля теоретических знаний и концептуального понимания.

Задачи:

1. Проверить усвоение основных понятий, терминов и принципов работы технологии модуля.
2. Оценить способность обучающегося анализировать и применять знания в знакомых и новых контекстах.

ФИО обучающегося: _____

Дата проведения: _____

Время выполнения: 30 минут

Часть 1. Выберите правильный ответ

1. Какой блок в Scratch отвечает за бесконечное повторение действий?
 - а) повторить 10 раз
 - б) ждать 1 секунд
 - в) всегда
2. г) если... то
3. Какой блок используется для создания переменной?
 - а) задать
 - б) изменить на
 - в) создать переменную
 - г) список
4. Какой блок отвечает за движение спрайта?
 - а) говорить
 - б) иди 10 шагов
 - в) следующий костюм
 - г) играть звук

Часть 2. Дайте краткий ответ

4. Для чего используется блок «переменная»?

-
5. Объясните разницу между блоками ждать и повторить.
-

Часть 3. Практическое задание

6. Опишите алгоритм создания простой анимации движения спрайта от левого края экрана к правому.
-

Оценка: Часть 1 (3 балла) + Часть 2 (4 балла) + Часть 3 (3 балла) = **10 баллов**

Приложение 5.
«Промежуточный контроль»

Критериальная карта оценки модуля

Критерий	Макс. балл	Обучающийся А	Обучающийся Б
А. Теоретические знания (тест)	20		
Б. Практический проект модуля	50		
<i>Качество реализации</i>	(20)		
<i>Соответствие ТЗ</i>	(15)		
<i>Защита проекта</i>	(15)		
В. Активность и работа на занятиях	30		
ИТОГО за модуль	100		

Уровень освоения:

- 85-100 баллов – **высокий** (5)
- 65-84 балла – **средний** (4)
- 50-64 балла – **удовлетворительный** (3)
- менее 50 – **низкий** (не засчитано)

Приложение 6.
«Итоговый контроль»

Требования к итоговому проекту года

Цель: Комплексная оценка результатов освоения программы за учебный год.

Требования к проекту:

1. Проект должен демонстрировать интеграцию знаний, полученных в течение года
2. Должен иметь практическую значимость
3. Должен быть полностью реализован (рабочий прототип)
4. К проекту должна быть подготовлена презентация и документация

Примерные темы для разных годов обучения:

- **1 год:** «Интерактивная обучающая игра» (Scratch/Kodu)
- **2 год:** «Веб-сайт-портфолио» или «Умное устройство на Arduino»
- **3 год:** «Телеграм-бот» или «Система умного дома»

Приложение 7.
«Итоговый контроль»

Оценочный лист итогового проекта

Название проекта: _____

Автор(ы): _____

Дата защиты: _____

Критерий оценки	Баллы (1-10)	Комментарии
1. Техническая сложность реализации		
2. Корректность кода/схемы		
3. Креативность и оригинальность		
4. Полнота реализации (соответствие ТЗ)		
5. Качество презентации		
6. Ответы на вопросы		
7. Работа в команде (если применимо)		
8. Соблюдение сроков		
ИТОГО	/80	

Итоговый уровень:

- 64-80 баллов: **Высокий уровень**
- 40-63 балла: **Средний уровень**
- менее 40 баллов: **Низкий уровень**

Рекомендации: _____

Педагог: _____ / _____

Приложение 8.

«Контроль воспитательной работы»

Цель: мониторинг и оценка эффективности воспитательного компонента программы, направленного на формирование ценностных ориентаций и социально значимых качеств личности.

Задачи:

1. Оценить сформированность гражданской позиции, патриотизма, уважения к отечественным достижениям.
2. Отследить развитие ответственности, дисциплины, коммуникативной культуры и цифровой этики.
3. Задокументировать проявления творческой и социальной активности обучающегося.

Карта оценки личностных и воспитательных результатов

ФИО обучающегося: _____

Период оценки: с _____ по _____

Критерий оценки	Показатели	Балл (1-5)	Комментарии
1. Гражданская позиция и патриотизм	Интерес к достижениям отечественной науки		
	Участие в тематических мероприятиях		
2. Ответственность и дисциплина	Своевременное выполнение заданий		
	Бережное отношение к оборудованию		
3. Коммуникативные навыки	Умение работать в команде		
	Культура общения с педагогом и сверстниками		
4. Цифровая грамотность и безопасность	Понимание важности защиты личных данных		
	Этическое поведение в цифровой среде		

Критерий оценки	Показатели	Балл (1-5)	Комментарии
5. Творческая активность	Инициативность в проектной работе		
	Участие в конкурсах и мероприятиях		

Итоговая оценка: _____

Характеристика обучающегося:

Педагог: _____ / _____

Анкета для родителей (обратная связь)

Цель: получение внешней оценки качества реализации программы, степени удовлетворенности участников образовательных отношений и выявить направления для совершенствования.

Задачи:

1. Оценить влияние программы на интерес, мотивацию и личностное развитие ребенка с точки зрения семьи.
2. Получить обратную связь об организации процесса, работе педагога и материально-техническом обеспечении.
3. Укрепить партнерские отношения с родителями, вовлечь их в образовательный процесс.

Уважаемые родители!

Просим вас заполнить эту анкету для улучшения качества образовательной программы «IT-квантум».

ФИО родителя: _____

ФИО ребенка: _____

Дата заполнения: _____

1. Как ваш ребенок относится к занятиям в «IT-квантуме»?

- С большим интересом и энтузиазмом
- Интересуется, но без особого энтузиазма
- Занимается без особого интереса
- Не хочет ходить на занятия

2. Что, по вашему мнению, дает ребенку участие в программе? (можно выбрать несколько)

- Новые знания и навыки
- Развитие логического мышления
- Уверенность в себе
- Интерес к учебе в целом
- Новых друзей
- Другое: _____

3. Замечали ли вы изменения в поведении/интересах ребенка после начала занятий?

- Да, стал больше интересоваться техникой/программированием
- Стал более усидчивым и внимательным

- Научился лучше планировать свое время
 - Изменений не заметили
 - Другое: _____
-

4. Как вы оцениваете организацию учебного процесса?

- Расписание занятий: [] Удобно [] Неудобно
- Информированность о успехах ребенка: [] Достаточно [] Недостаточно
- Материально-техническое оснащение: [] Хорошее [] Удовлетворительное [] Плохое

5. Насколько вы удовлетворены работой педагога?

- Профессионализм: [] Высокий [] Средний [] Низкий
- Взаимодействие с детьми: [] Отличное [] Хорошее [] Удовлетворительное
- Контакт с родителями: [] Активный [] Удовлетворительный [] Слабый

6. Что бы вы хотели изменить/добавить в программу?

7. Порекомендовали бы вы программу «IT-квантум» другим родителям?

- Да, обязательно
 - Возможно
 - Нет
 - Почему? _____
-

8. Ваши пожелания и предложения:

Благодарим за сотрудничество!

Шаблон портфолио обучающегося

Портфолио участника программы «IT-квантум»

ФИО: _____

Год обучения: _____

Педагог: _____

1. Личные данные

- Дата рождения: _____
- Контакты: _____
- Школа, класс: _____

2. Изученные модули и результаты

Модуль	Оценка	Лучшая работа

3. Проекты и достижения

- Участие в конкурсах:

- Грамоты и дипломы:

- Самостоятельные проекты:

4. Отзыв педагога

Дата формирования: _____

Лист самооценки обучающегося

Цель: сформирование у обучающегося навыка рефлексии, критического анализа собственной учебной деятельности и постановки целей.

Задачи:

1. Научить ребенка анализировать свои успехи, трудности и пути их преодоления.
2. Стимулировать самоорганизацию и осознанное отношение к процессу обучения.
3. Получить для педагога информацию о субъективном восприятии учебного процесса обучающимся для индивидуального подхода.

ФИО: _____

Модуль/период: _____

Дата: _____

1. Что я узнал нового:

2. Что было самым интересным:

3. С какими трудностями столкнулся:

4. Как я их преодолел:

5. Моя оценка своей работы (по 10-балльной шкале):

6. Мои цели на следующий период:

Подпись обучающегося: _____