

Государственное автономное образовательное учреждение дополнительного
образования «Центр для одаренных детей «Поиск»

УТВЕРЖДЕНО
приказом Центра «Поиск»
№ 148 от 27 марта 2026 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«ПРОГРАММИРОВАНИЕ»

Направление: наука

Возраст обучающихся: 8 - 18 лет

Срок реализации: от 1 до 9 лет

Форма обучения: очная

Авторы программы: Никотина Лидия Леонидовна, заместитель
директора по учебной работе ГАОУ ДО «Центр
для одаренных детей «Поиск»;
Пономаренко Елена Александровна,
руководитель направления информационные
технологии ГАОУ ДО «Центр для одаренных
детей «Поиск»

Ставрополь
2026

ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ.....	2
ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	3
КУРС «АЛГОРИТМИКА».....	11
КУРС «ГЕЙМДИЗАЙН И МОДЕЛИРОВАНИЕ».....	15
КУРС «ВИЗУАЛЬНОЕ ПРОГРАММИРОВАНИЕ».....	21
КУРС «ОСНОВЫ АЛГОРИТМИКИ И ЛОГИКИ».....	25
КУРС «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ».....	35
КУРС «ПРОГРАММИРОВАНИЕ НА PYTHON».....	43
КУРС «ВЕБ-ПРОГРАММИРОВАНИЕ».....	49
КУРС «ЯЗЫК ПРОГРАММИРОВАНИЯ СИ++».....	53
КУРС «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ».....	61
КУРС «РАЗРАБОТКА ИГР НА UNITY».....	69
КУРС «ОСНОВЫ МОБИЛЬНОЙ РАЗРАБОТКИ».....	77
КУРС «УМНЫЙ ДОМ».....	83
СПИСОК ЛИТЕРАТУРЫ.....	88
СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ.....	91

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Сегодня во всех отраслях российской экономики происходят значительные изменения и перестройки. Особенно остро это касается ИТ-отрасли: производители зарубежного программного обеспечения уходят с рынка, российские серверы и сайты российских компаний подвергаются негативному внешнему воздействию. На этом фоне актуально быстрое и эффективное увеличение числа программистов, способных на федеральном уровне решать задачи государства.

Программирование занимает значительное место в системе подготовки профессионалов в области информационных технологий. В федеральном проекте «Развитие кадрового потенциала ИТ-отрасли» национальной программы «Цифровая экономика РФ» обозначена уникальность и ценность программирования как учебного вида деятельности. Готовить профессиональных программистов нужно уже в школе.

Школьники, умеющие разработать и реализовать проект компьютерной программы среднего уровня сложности с графическим пользовательским интерфейсом и доступом к базам данных и сети Интернет, способны стать лидерами в школе и в обществе. Они лучше успевают и по другим предметам, поскольку культура их мышления выше, а ассортимент методов выполнения различных работ богаче. В связи с этим представляется достаточно важным привить учащимся навыки программирования.

1. Основные характеристики программы

Научно-технический прогресс диктует новые требования к содержанию и организации образовательного процесса. Появляются новые технологии и средства информации, которые благодаря приоритетному национальному проекту «Образование» активно поступают в российские образовательные учреждения.

Внедрение новых информационных и коммуникационных технологий на базе Internet сегодня является одним из важнейших резервов повышения эффективности непрерывного образования и самообразования детей.

Программа «Программирование» предназначена для детей, желающих развить свои умственные способности, получить углубленные теоретические и практические знания и навыки по актуальным в настоящее время направлениям в сфере новых информационных технологий.

Основной идеей программы, отличающее ее от существующих программ, является формирование у обучающихся которая заключается в популяризации и развитии интереса детей школьного возраста к программированию, формировании у них знаний основ программирования, умения составлять план

деятельности, развитии познавательной активности, исследовательских, прикладных способностей, алгоритмического, логического и критического мышления.

1.1. Направленность программы

Дополнительная общеобразовательная общеразвивающая программа «Программирование» имеет техническую направленность, в связи с этим рассматриваются два актуальных аспекта изучения:

1. технологический: программирование рассматривается как средство формирования образовательного потенциала, позволяющего развивать наиболее передовые на сегодняшний день технологии – цифровые.

2. общеобразовательный: программирование рассматривается как средство развития основных познавательных процессов, умения анализировать, выявлять сущности и отношения, описывать планы действий и делать логические выводы.

1.2. Адресат программы

Программа предназначена для одаренных школьников 2-11 классов, проявляющих повышенный интерес к программированию, а так же формирует знания основ программирования, умения составлять план деятельности, развитии познавательной активности, исследовательских, прикладных способностей, алгоритмического, логического и критического мышления.

Актуальность программы

Совершенствование технологических и программных средств привело к снижению количества часов, отводимых для изучения программирования в Программе среднего общего образования по информатике. Современные визуальные и мультимедийные пользовательские среды являются теми конкурентами, которые вытесняют разработку программ из сферы интересов школьников. Для работы за компьютером для поиска информации в сети пользователь имеет простые инструменты, не требующие мыслительных усилий при применении. Как следствие, в большинстве школ отсутствует системная работа по обучению школьников 7-10 классов языкам программирования.

Дополнительная общеобразовательная общеразвивающая программа «Программирование» обеспечивает изучение востребованных языков программирования, нацелена на формирование математического аппарата описания и построения процессов обработки информации, в том числе человеком и технологическим устройством, создания и исследования числовых и нечисловых математических моделей, создания компьютерных программ с графическим пользовательским интерфейсом, доступом к базам данных и сети Интернет.

1.3. Отличительные особенности/новизна программы

При планировании и создании программы учитывается, что раздел «Программирование» становится одним из ведущих в изучении информатики. Обучение на основе проектов стимулирует обучающихся к решению сложных реальных задач. Они исследуют, делают заключения, анализируют и обобщают информацию. Начало XXI века ознаменовалось бурным развитием IT-технологий. Мировые лидеры IT-индустрии периодически обращаются к школьникам с призывом изучать программирование. Становится понятно, что чем раньше ученик начнёт овладевать навыками программирования, тем больший запас знаний, умений и навыков он получит к моменту выбора основного рода деятельности. Даже если в будущем карьерный путь ребёнка не будет связан с программированием, умение разбираться в сложных системах и взаимодействовать с новыми технологиями пригодится ему в любой сфере, ведь цифровые технологии используются повсеместно.

Кроме того, изучение программирования позволит ребёнку познакомиться с сообществом таких же заинтересованных ребят, введёт в подробности и тонкости проектной деятельности. Овладевая навыками программирования, ребёнок затрагивает и смежные направления, такие как логика, математика, теория вероятности, а также и другие области знаний: география, биология, физика, литература — в зависимости от интересов ребёнка и выбора вектора развития собственного проекта. Когда у ребёнка сформирован необходимый набор знаний, умений и навыков, выполнен ряд задач и упражнений по разным темам, он может, используя их, работать над собственным проектом. Это позволяет развивать такие важные компетенции XXI века, как критическое мышление, креативность и любознательность, коммуникация и работа в команде.

Вид программы – модульная.

Программа представляет собой совокупность 12 самостоятельных логически завершённых курсов, которые реализуются в очной форме.

№	Название курса	Форма обучения	Класс обучающегося
1.	Алгоритмика	очная	2-3
2.	Геймдизайн и моделирование (Нейросмена 1.0)	очная	4-5
3.	Визуальное программирование	очная	3-4

4.	Основы мобильной разработки	очная	6-8
5.	Основы алгоритмики и логики	очная	5-6
6.	Разработка игр на Unity	очная	6-7
7.	Основы алгоритмизации и программирования	очная	7-8
8.	Программирование на Python	очная	7-11
9.	Веб-программирование	очная	8-11
10.	Умный дом	очная	8-10
11.	Язык программирования СИ++	очная	9-11
12.	Разработка мобильных приложений	очная	9-11

Цели программы

1.6 Цели и задачи программы

Цель программы - создание условий для повышения качества образования через обеспечение участников образовательного процесса углубленными знаниями и компетенциями из области программирования, востребованными современной цифровой экономикой, подготовка базы для последующего профессионального изучения программирования в высшей школе.

В соответствии с целью реализуются **задачи программы**:

1) **Обучающие:**

- ✓ - формирование умений в области создания текстов программ для персональных компьютеров на основе глубоких знаний среды программирования и языков программирования;
- ✓ расширение и углубление представлений об автоматизированной обработке информации;
- ✓ освоение методов программирования: выдвижение и обоснование идеи решения задачи, структурирование этой идеи, формализация элементов полученной структуры средствами выбранного языка, анализ результатов решения задачи при различных значениях исходных данных.

2) **Развивающие:**

- социализация и адаптация обучающихся к жизни в обществе, формирование общей культуры обучающихся;
- формирование культуры здорового и безопасного образа жизни;
- развитие умения работать с компьютерными программами;
- развитие умения работать с дополнительными источниками информации;

- повышение интеллектуального уровня обучающихся;
- развитие умения вступать в контакт с группой, заявлять о себе, влиять на то, что происходит в контакте;
- усиление интереса подростков к процессу личностного роста;
- создание условий для дальнейшего самопознания, саморазвития и самосовершенствования подростков;
- развитие критического мышления;
- рост уровня самоактуализации и осмысленности жизни участниками программы.

3) Воспитательные:

- создание и обеспечение необходимых условий для личностного развития, профессионального самоопределения и творческих способностей обучающихся;
- формирование определенного мировоззрения, противодействующего терроризму и экстремизму, связанного с устоями и обычаями, национальными и культурными традициями, историей региона, межнациональной и межрелигиозной толерантностью;
- воспитание культуры безопасного труда при работе за компьютером;
- воспитание культуры работы в глобальной сети;
- осознание ребятами собственной значимости и уникальности;
- формирование качеств мышления, необходимых для адаптации в современном информационном обществе;
- формирование внутренней готовности обучающихся к обретению нового опыта и активной жизненной позиции.

Категория обучающихся

Программа предназначена для детей, желающих развить свои умственные способности, получить углубленные теоретические и практические знания и навыки по актуальным в настоящее время направлениям в сфере новых информационных технологий.

Возраст обучающихся: 8 – 18 лет.

Наполняемость группы: 10-12 человек.

Состав групп: разновозрастной.

Условия приема детей

На курсы зачисляются все желающие при наличии свободных мест.

Сроки реализации программы

Для обучения на всех курсах программы отводится 9 лет.

Продолжительность отдельного курса составляет от 2-х недель до 1 учебного года (в зависимости от курса).

Формы реализации программы – очная.

Формы организации деятельности обучающихся: индивидуальная, групповая, фронтальная.

Методы обучения:

- 1) по способу организации занятий – словесные, наглядные, практические;
- 2) по уровню деятельности обучающихся – объяснительно-иллюстративные, репродуктивные, частично-поисковые, исследовательские.

Типы занятий: комбинированные, теоретические, практические, репетиционные, контрольные.

Режим занятий

В зависимости от курса, возможен один из следующих режимов занятий:

- 1) один раз в неделю по два учебных часа;
- 2) один раз в неделю по три учебных часа;
- 3) один раз в неделю по четыре учебных часа;
- 4) два раза в неделю по два учебных часа.

Продолжительность учебного часа – 40 минут.

Планируемые результаты освоения программы

1) **Предметные результаты:**

- наличие навыков работы с компьютером как средством управления информацией, способность работать с информацией в глобальных компьютерных сетях;
- владение основными методами, способами и средствами получения, хранения, переработки информации;
- умение использовать системы электронного управления документами;
- знание принципов построения и архитектуры вычислительных систем;
- знание основных требований информационной безопасности, в том числе защиты коммерческой и государственной тайны;
- знание базовых основ нормативных документов в области информационной безопасности и защиты информации;
- владение базовыми методами и средствами защиты информации от несанкционированного доступа;
- умение использовать математические и инструментальные средства для обработки, анализа и систематизации информации;

- умение разрабатывать математические модели для решения прикладных задач;
- знание основных возможностей пакетов компьютерной графики, приемы работы в программах растровой и векторной графики;
- владение базовыми приемами создания и программирования интернет-ресурсов.
- систематизация знаний, относящихся к математическим объектам информатики; умение строить математические объекты информатики, в том числе логические формулы;
- сформированность базовых навыков и умений по соблюдению требований техники безопасности, гигиены и ресурсосбережения при работе со средствами информатизации;
- сформированность представлений об устройстве современных компьютеров, о тенденциях развития компьютерных технологий; о понятии «операционная система Linux» и основных функциях операционной системы; об общих принципах разработки и функционирования интернет-приложений;
- сформированность представлений о компьютерных сетях и их роли в современном мире; знаний базовых принципов организации и функционирования компьютерных сетей, норм информационной этики и права, принципов обеспечения информационной безопасности, способов и средств обеспечения надёжного функционирования средств ИКТ;
- понимания основ правовых аспектов использования компьютерных программ и работы в Интернете.

2) **Метапредметные результаты:**

- способность самостоятельно ставить цели учебной и исследовательской, проектной деятельности, планировать, осуществлять, контролировать и оценивать учебные действия в соответствии с поставленной задачей и условиями ее выполнения;
- умения самостоятельно планировать альтернативные пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;
- умения находить необходимую информацию, критически оценивать и интерпретировать информацию в различных источниках (в справочниках, литературе, Интернете), представлять информацию в различной форме (словесной, табличной, графической, символической), обрабатывать, хранить и передавать информацию в соответствии с познавательными или коммуникативными задачами;
- сформированность навыков осуществления познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания;
- владения навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и

оснований, границ своего знания и незнания, новых познавательных задач и средств их достижения;

- умения продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты;

- владения языковыми средствами;

- умения ясно, логично и точно излагать свою точку зрения, использовать адекватные языковые средства.

3) Личностные результаты:

- готовность обучающихся руководствоваться системой позитивных ценностных ориентаций и расширение опыта деятельности на ее основе и в процессе реализации основных направлений воспитательной деятельности;

- сформировать в образовательном процессе качества личности: мировоззрение, убеждения, нравственные принципы, систему ценностных отношений обучающихся к себе, другим людям, профессиональной деятельности, гражданским правам и обязанностям, государственному строю, духовной сфере, общественной жизни;

- сформировать мотивацию к обучению и целенаправленной познавательной деятельности, готовность к саморазвитию и личностному самоопределению, способность ставить цели и строить жизненные планы.

Способы определения результативности

Педагогическое наблюдение, педагогический анализ результатов анкетирования, тестирования, опросов, зачетов, активности обучающихся на занятиях, мониторинг.

Виды контроля – текущий, итоговый.

Формы подведения итогов реализации программы

Итоги реализации отдельного курса программы подводятся в одной из следующих форм: тестирование, зачет, экзамен, контрольное занятие, защита проекта, олимпиада, конференция, интеллектуальный конкурс.

Документальной формой подтверждения итогов реализации отдельного курса программы является документ об образовании «Сертификат» (с оценкой) или документ об обучении «Сертификат» (без оценки) установленного Центром «Поиск» образца.

КУРС «АЛГОРИТМИКА»

Цель курса:

- организовать учебно-познавательный процесс при помощи проектной деятельности на основе интегрированной компьютерной среды ПервоЛого.

Задачи курса:

- сформировать навыки работы в интегрированной графической среде ПервоЛого;
- сформировать у учащихся информационную и функциональную компетентности, способствовать развитию алгоритмического мышления;
- обучить методам работы в интегрированной графической среде ПервоЛого, создавая проекты из различных областей знаний.

Режим занятий: шесть раз в неделю по два учебных часа.

Форма реализации курса: очная

Форма проведения итоговой аттестации: персональный творческий проект.

Учебно-тематический план курса «Алгоритмика»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Интегрированная среда ПервоЛого.	4	4	8
2	Тема 2. Управление объектами. Алгоритм и программа.	4	6	10
3	Тема 3. Мультимедийные проекты.		6	6
Итого:		8	16	24

Содержание курса «Алгоритмика»

Уровень предъявления материала обеспечивает учащимся возможность самостоятельно создавать творческие проекты в интегрированной мультимедийной среде ПервоЛого.

Учащиеся должны знать:

- основные объекты среды ПервоЛого;
- технологию создания декораций микромира;
- технологию организации движения черепашки;
- понятие команды и входных параметров;
- понятие алгоритм и программа;
- правила оформления программы;
- технологию создания анимационного сюжета.

Учащиеся должны уметь:

- создавать и сохранять альбом;
- создавать, копировать, перемещать, изменять рисунки;
- использовать инструменты;
- работать с формами черепашки;
- программировать черепашку;
- работать с графической информацией и текстом;
- создавать мультимедийные проекты.

Формы занятий:

- фронтальная;
- индивидуальная;
- беседа-обсуждение;
- конкурс;
- защита проекта.

Тема 1. Интегрированная среда ПервоЛого.

Теория. Особенности среды ПервоЛого. Рабочее поле, инструменты, формы, команды управления черепашкой. Изменение облика черепашки.

Практика. Создание рисунка с использованием инструментов среды. Создание рисунка с использованием форм Черепашки. Работа с фрагментами рисунка. Создание микромира с помощью Поля форм и графического редактора. Наполнение микромира обитателями. Управление движением черепашки. Формирования траектории движения черепашки, изменение направлений.

Форма подведения итогов: контрольная работа.

Тема 2. Управление объектами. Алгоритм и программа.

Теория. Понятие алгоритма и программы. Назначение обязательных частей программы: заголовка, тела, признака завершения. Циклический алгоритм, вложенные циклы. Алгоритм с ветвлением. Координаты черепашки на рабочем поле. Задание координат. Измерения с помощью датчиков.

Практика. Создание программ управления черепашкой. Формирование цепочки команд управления черепашкой. Рисование многоугольников, дуг и окружностей. Задание координат на рабочем поле, управление черепашкой. Реагирование объектов друг на друга, цвет. Работа с текстом. Работа с альбомом: добавление и удаление листов альбом, вставка готовых файлов.

Форма подведения итогов: контрольная работа.

Тема 3.Мультимедийные проекты.

Теория. Этапы анимационного сюжета. Способы решения задач творческого характера.

Практика. Создание мультимедийных творческих проектов, используя возможности среды ПервоЛого.

Форма подведения итогов: итоговый творческий проект.

Методическое обеспечение курса «АЛГОРИТМИКА»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Интегрированная среда ПервоЛого.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) Справочные материалы в облаке; 2) Исходные файлы для создания проектов; 3) ПО ПервоЛого онлайн.	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проектная работа
Тема 2. Управление объектами. Алгоритм и программа.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) Справочные материалы в облаке; 2) Исходные файлы для создания проектов; 3) ПО ПервоЛого онлайн.	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проектная работа
Тема 3. Мультимедийные проекты.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) Справочные материалы в облаке; 2) Исходные файлы для создания проектов; 3) ПО ПервоЛого онлайн.	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Итоговый творческий проект

КУРС « ГЕЙМДИЗАЙН И МОДЕЛИРОВАНИЕ»

Цели курса:

- познакомить учащихся с основами геймдизайна, разработкой игровых миров и использованием нейросетей для создания игровых проектов с применением доступных программных средств.

Задачи курса:

- познакомить с профессией геймдизайнера и структурой дизайн-документа;
- обучить базовым инструментам создания игр;
- сформировать навык создания игровых уровней, 3D-объектов и тайлсетов;
- научить использовать российские нейросети для генерации сюжета, диалогов и изображений;
- развить алгоритмическое, пространственное и творческое мышление;
- сформировать навык публичной защиты проекта.

Режим занятий:

- шесть раз в неделю по два учебных часа;

Форма реализации курса: очная.

Форма проведения итоговой аттестации: защита творческого проекта

Учебно-тематический план курса

«Геймдизайн и моделирование»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Геймдизайн: профессия и дизайн-документ	2	2	4
2	Тема 2. Создание игрового мира.	2	4	6
3	Тема 3. Генерация идей и контента с	2	2	4

	помощью нейросетей			
4	Тема 4. Создание 2D/3D-объектов и тайлсетов	2	2	4
5	Тема 5. Создание квеста/сюжета: диалоги и условия	2	2	4
6	Защита авторского творческого проекта.		2	2
Итого:		10	14	24

Содержание курса «Геймдизайн и моделирование»

Учащиеся должны знать:

- основные понятия геймдизайна (жанры, дизайн-документ, уровень, NPC, тайлсет);
- интерфейс и базовые возможности RPG Paper Maker ;
- какие российские нейросети доступны для генерации текстов и изображений (YandexGPT, GigaChat, Kandinsky, Шедеврум) ;
- что такое тайлсет, пиксель-арт, 3D-объект.

Учащиеся должны уметь:

- составить простой дизайн-документ игры;
- создать игровую локацию в RPG Paper Maker;
- сгенерировать сюжет и диалоги с помощью YandexGPT или GigaChat ;
- сгенерировать изображения персонажа и локации с помощью Kandinsky или Шедеврума ;
- создать простой тайл в Piskel и импортировать его в движок;
- создать простой 3D-объект во встроенном редакторе;
- создать короткий квест с диалогами и условиями;
- публично представить свой проект.

Формы занятий:

- фронтальная;
- индивидуальная;

- беседа-обсуждение;
- конкурс;
- защита проекта.

Тема 1. Геймдизайн: профессия и дизайн-документ

Теория. Кто такой геймдизайнер. Жанры компьютерных игр. Что такое дизайн-документ (Game Design Document). Структура GDD: идея, механики, сюжет, персонажи, уровни. Примеры из известных игр.

Практика. Составление простого дизайн-документа для своей будущей игры. Выбор жанра, основной идеи, целевой аудитории. Презентация идеи группе.

Форма подведения итогов: проверка дизайн-документа.

Тема 2. Создание игрового мира

Теория. Знакомство с RPG Paper Maker — бесплатным движком для создания 3D-игр с 2D-спрайтами. Интерфейс программы: карта, слои, объекты. Система координат в 3D-пространстве. Основы работы с ландшафтом и текстурами.

Практика. Создание простого игрового уровня: создание карты (локации); добавление объектов окружения (деревья, дома, препятствия); настройка ландшафта и текстур; размещение стартовой точки персонажа, сохранение и экспорт проекта.

Форма подведения итогов: готовая локация в RPG Paper Maker.

Тема 3. Генерация идей и контента с помощью нейросетей

Теория. Что такое нейросети и как они помогают геймдизайнерам. Обзор доступных в России нейросетей.

Практика. С помощью YandexGPT или GigaChat сгенерировать сюжет и диалоги для своей игры. С помощью Kandinsky или Шедеврума сгенерировать изображения персонажа и локации. Отобрать лучшие варианты и встроить в дизайн-документ.

Форма подведения итогов: тексты и изображения, сгенерированные ИИ, с кратким комментарием.

Тема 4. Создание 2D/3D-объектов и тайлсетов

Теория. Что такое тайлсет и для чего он нужен. Пиксель-арт и его особенности. Разница между 2D и 3D-объектами в играх. Обзор доступных редакторов: Piskel (браузерный редактор пиксель-арта), встроенные инструменты RPG Paper Maker.

Практика. В редакторе Piskel создать простой тайл (16×16 или 32×32 пикселя) для пола или стены. Импортировать созданный тайл в RPG Paper Maker. Создать простой 3D-объект (например, сундук или дерево) во встроенном редакторе RPG Paper Maker. Разместить созданные объекты на карте.

Форма подведения итогов: собственный тайл и 3D-объект, встроенные в игровую локацию.

Тема 5. Создание квеста/сюжета: диалоги и условия

Теория. Основные понятия: карта, персонаж (NPC), событие, диалог, триггер. Простейшие сценарии: диалог, сбор предмета, переход между картами. Как создавать условия выполнения заданий.

Практика. Создание короткого квеста (2–3 шага) в RPG Paper Maker: создать новую карту; добавить NPC с диалогом (текст можно сгенерировать в YandexGPT); настроить условие (например, собрать предмет → отдать NPC → получить ключ → открыть дверь); добавить переход между картами. Загрузить готовый проект для презентации.

Форма подведения итогов: защита проекта.

Методическое обеспечение курса «Геймдизайн и моделирование»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Геймдизайн: профессия и дизайн-документ	Комбинированная	Лекция, беседа, работа в группах	Шаблон GDD, примеры игр	Проектор, ПК	Проверка GDD
Тема 2. Создание игрового мира	Практическая	Пошаговое выполнение, демонстрация	Видео-инструкция по RPG Paper Maker	ПК, RPG Paper Maker	Готовая локация
Тема 3. Генерация идей и контента с помощью российских нейросетей	Комбинированная	Частично-поисковый, исследовательский	Справочник промптов, списки нейросетей	ПК, доступ к YandexGPT, GigaChat, Kandinsky	Сгенерированный контент
Тема 4. Создание 2D/3D-объектов и тайлсетов	Практическая	Обучение через действие	Примеры тайлсетов, пиксель-арт гайды	ПК, Piskel, RPG Paper Maker	Тайл и 3D-объект в игре
Тема 5. Создание квеста/сюжета: диалоги и условия	Практическая	Проектный метод, решение кейса	Готовые шаблоны диалогов, карта квеста	ПК, RPG Paper Maker	Работающий прототип квеста
Защита итогового проекта	Защита проектов	Публичное выступление	Критерии оценки, форма отзыва	Проектор, ПК, презентация	Защита проекта

КУРС «ВИЗУАЛЬНОЕ ПРОГРАММИРОВАНИЕ»

Цели курса:

- сформировать понимание, – каким образом можно создавать основные алгоритмические конструкции в программе Scratch, а также реализовывать их при работе с играми, анимацией;
- развивать опыт творческой и проектной деятельности.

Задачи курса:

- закрепить обучение технологии работы на персональном компьютере в среде программирования Scratch;
- обучить принципам совместной работы, обмена идеями, особенностям построения различных базовых алгоритмов программы;
- обеспечить возможность для создания итогового авторского проекта, имеющего практическую ценность.

Режим занятий:

- один раз в неделю по два учебных часа 68 часов;

Форма реализации курса: очная.

Форма проведения итоговой аттестации: персональный творческий отчёт «Защита проекта».

Учебно-тематический план курса

«Визуальное программирование»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Первые шаги в Scratch.	10	16	26
2	Тема 2. Проектная деятельность в Scratch.	6	34	40
3	Защита авторского творческого проекта.		2	2
Итого:		16	52	68

Содержание курса «Визуальное программирование»

Уровень предъявления материала обеспечивает учащимся возможность создавать простейшие алгоритмы и реализовывать их в графической среде программирования Scratch.

Учащиеся должны знать:

- что такое Scratch и его назначение;
- базовые алгоритмические конструкции;
- принцип создания анимации и движения объектов;
- назначение пунктов меню и элементов интерфейса среды Scratch;
- что такое координаты объектов;
- основные команды и события в проектах Scratch;
- принцип взаимодействия объектов;
- способы создания объектов в среде Scratch;
- основы простейшего программирования;
- принципы совместной работы и обмена идеями;
- как работать в команде.

Учащиеся должны уметь:

- работать с цифровыми инструментами и системами;
- работать в графической среде программирования Scratch;
- размещать объекты на сцене, поворачивать их и масштабировать;
- создавать программы для движения спрайтов по сцене;
- озвучивать проекты в Scratch;
- поэтапно создавать компьютерную игру;
- разрабатывать интерфейс для Scratch проекта;
- использовать числа для задания звуков;
- использовать простые переменные для счётных операций;
- использовать случайные числа в диапазоне от 1 до 10;
- организовывать мозговой штурм для поиска новых решений;
- свободно общаться в устной или в письменной форме с использованием специальных терминов;
- создавать описание логической последовательности событий.

Формы занятий:

- фронтальная;
- индивидуальная;
- беседа-обсуждение;
- конкурс;

- защита проекта.

Тема 1. Первые шаги в Scratch.

Теория. Вводное занятие. Техника безопасности. Среда конструирования. Знакомство со средой Scratch. Основные базовые алгоритмические конструкции (линейные алгоритмы, с условным оператором, циклического типа с предусловием и постусловием) и их исполнение в среде Scratch. Способы записи алгоритма. Принцип создания анимации и движения объектов. Сцена. Текущие данные о спрайте. Стилль поворота. Панель инструментов, Новый спрайт. Координаты объекта. Режим представления. Окно скриптов. Окно блоков.

Практика. Постановка задачи. Постановка эксперимента. Использование инструментов для сбора информации. Обсуждение результатов исследований и их объяснение. Проведение испытаний. Наблюдения. Рассуждения и аргументация. Работа в команде.

Форма подведения итогов: письменный опрос.

Тема 2. Проектная деятельность в Scratch.

Теория. Понятие проект, его структура и реализация в среде Scratch. Основные компоненты проекта Scratch: спрайты и скрипты. Создание спрайтов. Смена костюма. Сложная анимация в Scratch. Вложенные конструкции. Тестировщики в Scratch.

Практика. Проектная деятельность: создание простейших игр, интерактивных баннеров, открыток, разработка составных проектов.

Форма подведения итогов: авторский творческий проект.

Методическое обеспечение курса «Визуальное программирование»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Первые шаги в Scratch	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	1. Справочные материалы в облаке.	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет. Программное обеспечение	письменный опрос
Тема 2. Проектная деятельность в Scratch	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	1. Справочные материалы в облаке. 3. Комплект занятий, книга Е.Патаракин. Программирование в среде Scratch	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет. Программное обеспечение Scratch 2.0	письменный опрос

КУРС «ОСНОВЫ АЛГОРИТМИКИ И ЛОГИКИ»

Цели курса:

- создание условий для повышения качества образования через обеспечение участников образовательного процесса углубленными знаниями и компетенциями из области программирования;
- подготовка базы для последующего профессионального изучения программирования в высшей школе.
- развитие алгоритмического мышления учащихся, творческих способностей, аналитических и логических компетенций, а также пропедевтика будущего изучения программирования на одном из современных языков;
- расширить опыт творческой и проектной деятельности.

Задачи курса:

- обучить работать в виртуальной среде Кулибин;
- сформировать умения построения различных алгоритмов для решения поставленных задач;
- сформировать умения построения различных видов алгоритмов (линейных, разветвляющихся, циклических) для решения поставленных задач;
- развить умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата.

Режим занятий: один раз в неделю по два учебных часа.

Форма реализации курса – очная.

Форма проведения итоговой аттестации – персональный творческий отчёт «Защита проекта».

Тематический план курса «Основы алгоритмики и логики»

№	Наименование модуля	Контактная работа обучающихся с преподавателем, часов			Формы контроля / аттестации
		Теория	Практика	Всего	
1.	Основы алгоритмизации и	2	2	4	тестирование

	логики				
2.	Работа со средой Scratch	5	5	10	презентация результатов
3.	Создание мультфильмов и мини-игр	8	16	24	презентация проектов
4.	Работа в виртуальной среде Кулибин	6	18	24	презентация проектов
5.	Проектная деятельность	2	4	6	публичная презентация проектов
Итого:		20	48	68	

В курсе рассматриваются основные конструкции языка программирования Scratch (данных, переменных, ветвлениях, циклах и функциях), изучение базовых типов и структур данных (целые и вещественные числа, строки, списки, кортежи, словари). Работа в виртуальном приложении программирования робота «Кулибин». Прохождение заданий Всероссийских олимпиад. Разработка мультфильмов и мини-игр.

В результате освоения учебного курса обучающийся должен:

знать:

- основные принципы построения программ на языке;
- основные способы представления структур данных;
- базовые конструкции языка Scratch (операции присваивания, ветвления, цикла, ввода/вывода, описание данных, запись констант и выражений);
- основные методы работы с контейнерами данных (списками, строками, кортежами, словарями);
- алгоритмы обработки строковых переменных;
- понятие пользовательских функции, способы организации функций, принципы модульной технологии программирования;

уметь:

- формализовать поставленную задачу;
- тестировать и отлаживать задачи с целью повышения надёжности и эффективности;
- использовать современные готовые библиотеки классов, технологии и инструментальные средства.

Содержание курса «Основы алгоритмики и логики»

Модуль 1. Основы алгоритмизации и логики.

Тема 1. Введение. Понятие алгоритма. Основные алгоритмические конструкции: линейный алгоритм, циклический алгоритм, алгоритмы ветвления.

Теория. Понятие алгоритма и исполнителя алгоритмов. Допустимые действия исполнителя. Непосредственное и программное управление исполнителем. Понятие отладки программы. Основные алгоритмические конструкции: линейный алгоритм, циклический алгоритм, алгоритмы ветвления. Способы представления алгоритмов (словесный, в виде блок-схемы (графический способ), в виде программы). Определяемые допустимые действия.

Практика. Составление собственных алгоритмов. Прорисовка блок-схем, составление алгоритмических программ для решения задач.

Форма подведения итогов: тестирование.

Модуль 2. Работа со средой Scratch.

Тема 2. Знакомство со средой Scratch. Команды движения и событий. Графический редактор. Условные алгоритмы.

Теория. Основные элементы пользовательского интерфейса. Внешний вид рабочего окна. Блочная структура систематизации информации. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Установка русского языка для Scratch. Создание и сохранение документа. Понятия спрайта, сцены, скрипта. Очистка экрана. Основной персонаж как исполнитель программ. Система команд исполнителя (СКИ). Блочная структура программы. Непосредственное управление исполнителем.

Практика. Составление программ с пользовательских функций. Решение задач с использованием функций.

Форма подведения итогов: тестирование.

Тема 3. Работа с переменными.

Теория. Основные приёмы добавления переменных в среде Scratch, использование основных блоков для работы с переменными, основные приёмы составления программ с использованием переменных в среде Scratch.

Практика. Составление и отладка программ с использованием переменных.

Форма подведения итогов: тестирование.

Модуль 3. Создание мультфильмов и мини-игр.

Тема 4. Понятие «циклический алгоритм», «список», «Другие блоки», вычисление и подсчет очков.

Теория. Основные приёмы составления циклических алгоритмов, использование основных блоков для составления циклических алгоритмов. создание списка, работа со свойствами раздела «другие блоки».

Практика. Составление и отладка программ, использующих списки, параметры блока, циклы и составление алгоритмов подсчета очков.

Форма подведения итогов: тестирование.

Тема 5. Векторные графические редакторы: состав, особенности, использование для решения задач.

Теория. Объекты, создание объектов. Выделение, перемещение и трансформация объектов. Инструменты создания кривых, настройки инструментов и область их применения. Кривые Безье. Способы создания кривых. Изменение геометрии объекта с помощью инструмента редактирования форм. Закраска областей. Виды градиентных переходов, особенности. Создание и сохранение новых градиентных переходов.

Практика. Настройка программного интерфейса. Работа с графикой.

Форма подведения итогов: тестирование.

Тема 6. Создание мультфильма. Прорисовка персонажей и фона. Разработка сценария.

Теория. Раскадровка. Продумывание диалогов и тайминга.

Практика. Создание собственных фонов, персонажей и декораций. Озвучка персонажей. Проработка сценария.

Форма подведения итогов: тестирование.

Тема 7. Создание игры с несколькими уровнями и подсчетом очков.

Теория. Интерфейс игры. Элементы интерфейса игры. Инструменты создания и редактирования фонов сцены. Способы реализации смены уровней игры (фонов сцены).

Практика. Составление алгоритма подсчета очков, смена уровней. Тестирование бета-версий игры.

Форма подведения итогов: тестирование.

Модуль 4. Работа в виртуальной среде Кулибин.

Тема 8. Основы управления Омегаботом в виртуальной среде.

Теория. Подготовка, установка и настройка виртуальной среды. Знакомство с полигоном, основы движения Омегабота.

Практика. Ориентироваться в интерфейсе приложения, перемещать Омегабота по локациям полигона, составлять по инструкции и запускать простые программы из нескольких блоков, самостоятельно составлять программы для движения по простым траекториям

Форма подведения итогов: тестирование.

Тема 9. Датчики, подключение и алгоритмы их работы.

Теория. Датчики Омегабота. Светодиод. Циклы. Условный оператор. Дальномер и датчик касания. Датчик линии. Следование по линии. Датчик освещённости.

Практика. Установка и подключение датчиков. Использовать светодиоды в работе Омегабота. Определять истинность и ложность логических выражений, объединять условия с помощью логического сложения (ИЛИ). Использовать в работе датчики: дальномер и датчик касания. Ориентироваться в новых локациях полигона.

Форма подведения итогов: тестирование.

Тема 10. Алгоритмы выполнения соревновательных заданий.

Теория. Алгоритм обхода лабиринта. Кегерлинг. Сортировка. Линия зиг-заг.

Практика. Установка и подключение датчиков. Тестированием определять диапазон значений датчика. Использовать датчик линии для следования по линии. Использование и условия применения магнита. Применять изученные конструкции с условиями. Использовать датчик линии для определения обрыва

Форма подведения итогов: тестирование.

Тема 11. Создание собственного виртуального пространства и его прохождение.

Теория. Разработка собственного пространства и учебного полигона.

Практика. Составить сложный алгоритм с вложенными условиями. Использовать комбинацию разных датчиков для одной задачи. Продемонстрировать полученные за предыдущие уроки знания и навыки.

Форма подведения итогов: тестирование.

Модуль 5. Проектная деятельность.

Тема 12. Выбор темы и направления итогового проекта.

Теория. Проверочные работы по знаниям алгоритмов. Изучение и выборы темы итогового проекта.

Практика. Повторение пройденного материала. Определение тематики и выбор команды, распределение ролей.

Форма подведения итогов: тестирование.

Тема 13. Разработка итогового проекта.

Теория. Разработка проекта в среде Scratch или виртуальной среде Кулибин.

Практика. Проработка всех элементов проектной деятельности. Запуск проекта в работу и создание его основных элементов.

Форма подведения итогов: обсуждение и выявление лучшего проекта.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Введение. Понятие алгоритма. Основные алгоритмические конструкции: линейный алгоритм, циклический алгоритм, алгоритмы ветвления.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский..	Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 2. Знакомство со средой Scratch. Команды движения и событий. Графический редактор.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский..	Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование

Условные алгоритмы.					
Тема 3. Работа с переменными.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 4. Понятие «циклический алгоритм», «список», «Другие блоки», вычисление и подсчет очков.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 5. Векторные графические редакторы: состав, особенности, использование для решения задач.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 6. Создание	Комбинированная	Объяснительно-иллюстративный.	Винницкий Ю. А. Scratch и Arduino для	Проекционное оборудование,	Тестирование

мультфильма. Прорисовка персонажей и фона. Разработка сценария.		Частично-поисковый. Исследовательский.	юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	ПК. Доступ к сети Интернет.	
Тема 7. Создание игры с несколькими уровнями и подсчетом очков.	Практическая	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Винницкий Ю. А. Scratch и Arduino для юных программистов и конструкторов./ Винницкий Ю. А. — СПб.: БХВ-Петербург, 2018. — 176 с	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 8. Основы управления Омегабот в виртуальной среде.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Методические рекомендации от создателей виртуальной среды и программируемого робота Омегабот.	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 9. Датчики, подключение и алгоритмы их работы.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Методические рекомендации от создателей виртуальной среды и программируемого робота Омегабот.	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 10. Алгоритмы выполнения	Практическая	Объяснительно-иллюстративный. Частично-	Методические рекомендации от создателей виртуальной	Проекционное оборудование, ПК.	Тестирование

соревновательных заданий.		поисковый. Исследовательский.	среды программируемого робота Омегабот. и	Доступ к сети Интернет.	
Тема 11. Создание собственного виртуального пространства и его прохождение.	Практическая	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Методические рекомендации от создателей виртуальной среды и программируемого робота Омегабот.	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 12.Выбор темы и направления итогового проекта.	Практическая	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Методические рекомендации.	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 13.Разработка итогового проекта.	Практическая	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Методические рекомендации.	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 14.Защита выпускного проекта.	Практическая	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Методические рекомендации.	Проекционное оборудование, ПК. Доступ к сети Интернет.	Защита

КУРС «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

Цели курса:

- на примере языка программирования «Паскаль» раскрыть суть фундаментальных основ технологии программирования.

Задачи курса:

- обучить технологии работы на персональном компьютере в программной среде Pascal;
- обучить методам реализации основных алгоритмических конструкций на языке программирования «Паскаль», создавая программы из различных научных областей.

Режим занятий – шесть раз в неделю по три учебных часа.

Режим занятий – один раз в неделю по три учебных часа.

Форма реализации курса – очная.

Форма проведения итоговой аттестации – тестирование.

Учебно-тематический план курса «Основы алгоритмизации и программирования»

Вариант 1 (стандартный, 102 ч)

№	Наименование раздела, темы	Количество часов		
		Тео- рия	Прак- тика	Всег- о
1	Тема 1. Основные понятия языка программирования Паскаль.	1	2	3
2	Тема 2. Операторы ввода-вывода языка программирования Паскаль.	2	4	6
3	Тема 3. Ветвления на языке программирования Паскаль.	3	6	9
4	Тема 4. Операторы цикла на языке программирования Паскаль.	4	9	12
5	Тема 5. Массивы на языке программирования Паскаль.	6	12	18

6	Тема 6. Процедуры и функции на языке программирования Паскаль.	2	4	6
7	Тема 7. Строки на языке программирования Паскаль.	3	3	6
8	Тема 8. Сведения о файловой системе языка программирования Паскаль.	1	2	6
9	Тема 9. ООП	11	22	33
10	Защита творческого проекта.	1	2	3
	Итого:	34	68	102

Вариант 2 (ускоренный, 36ч)

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Основные понятия языка программирования Паскаль.	1	3	4
2	Тема 2. Операторы ввода-вывода языка программирования Паскаль.	2	2	4
3	Тема 3. Ветвления на языке программирования Паскаль.	2	4	6
4	Тема 4. Операторы цикла на языке программирования Паскаль.	2	6	8
5	Тема 5. Массивы на языке программирования Паскаль.	2	6	8
6	Тема 6. Строки на языке программирования Паскаль.	1	3	4
7	Итоговая контрольная работа.		2	2
Итого:		10	26	36

Уровень предъявления материала обеспечивает учащимся знакомство с фундаментальными основами технологии программирования.

Учащиеся должны знать:

– базовые конструкции языка программирования Паскаль (операции присваивания, ветвления, цикла, ввода/вывода, описание данных, запись констант и выражений);

- основные методы сортировки массивов (метод простого выбора, метод простого обмена, метод простой вставки);
- алгоритмы обработки символьных переменных;
- понятие подпрограммы, функции пользователя;
- понятие формальных и фактических параметров;
- способы организации процедур (подпрограммы) и функций, принципы модульной технологии программирования;
- алгоритмы обработки массивов (поиск элемента с заданными свойствами, удаление элементов, вставка элементов, замена элементов, обмен местами групп элементов);
- технологию разработки программ методом детализации.

Учащиеся должны уметь:

- иллюстрировать на конкретных примерах понятие алгоритма и его свойства;
- записывать конструкции языка программирования Паскаль;
- составлять, отлаживать и исполнять на ПК программы решения учебных задач по программированию;
- применять методы вычислений для решения задач из курса физики, математики и др.;
- применять метод пошаговой детализации построения алгоритма для решения задач.

Формы занятий, используемые при изучении данного курса:

- фронтальная;
- индивидуальная;
- урок-консультация;
- компьютерное тестирование;
- контрольная работа.

Тема 1. Основные понятия языка программирования Паскаль.

Теория. Алфавит и словарь языка Паскаль. Зарезервированные слова. Идентификаторы. Структура программы. Константы. Типы данных. Математические функции.

Практика. Описание констант и переменных. Запись арифметических выражений.

Форма подведения итогов. контрольная работа.

Тема 2. Операторы ввода-вывода языка программирования Паскаль.

Теория. Вывод данных на экран. Форматированный вывод. Клавиатурный и программный ввод. Линейные программы. Стандартные функции и арифметические выражения.

Практика. Организация ввода-вывода данных. Составление линейных алгоритмов. Вычисление математических выражений с помощью стандартных арифметических функций.

Форма подведения итогов: контрольная работа.

Тема 3. Ветвления на языке программирования Паскаль.

Теория. Ветвления на Паскале. Условный оператор IF. Оператор выбора CASE. Данные логического типа. Логические операции. Логические выражения.

Практика. Составление программ с использованием оператора ветвления. Составление программ с использованием оператора CASE. Запись логических выражений.

Форма подведения итогов: контрольная работа.

Тема 4. Операторы цикла на языке программирования Паскаль.

Теория. Циклы на Паскале. Оператор цикла с известным числом повторов. Вложенные циклы. Счетчики. Оператор безусловного перехода. Оператор цикла с предусловием. Оператор цикла с постусловием.

Практика. Составление и отладка программ с использованием операторов цикла.

Форма подведения итогов: контрольная работа.

Тема 5. Массивы на языке программирования Паскаль.

Теория. Массивы. Объявление массива. Действия над массивами. Действия над элементами массива. Инициализация массива. Вывод массива на экран. Задание массива типизированной константой. Перестановка элементов массива, поиск максимального и минимального элементов. Сортировка массива.

Практика. Составление и отладка программ, использующих одномерные массивы. Составление и отладка программ, использующих двумерные и трёхмерные массивы.

Форма подведения итогов: контрольная работа.

Тема 6. Процедуры и функции на языке программирования Паскаль.

Теория. Процедуры, определяемые пользователем. Процедуры с параметрами. Функции, определяемые пользователем. Рекурсивные подпрограммы.

Практика. Составление и отладка программ, использующих процедуры и функции.

Форма подведения итогов: контрольная работа.

Тема 7. Строки на языке программирования Паскаль.

Теория. Функции и процедуры обработки строк.

Практика. Составление и отладка программ, использующих строки.

Форма подведения итогов: тестирование.

Тема 8. Сведения о файловой системе языка программирования Паскаль.

Теория. Файловый тип. Виды файлов, файловая переменная, доступ к файлам. Организация взаимодействия переменных файлов с основной программой.

Практика. Чтение из файла. Запись в файл.

Форма подведения итогов: контрольная работа.

Тема 9. ООП.

Теория. Объектно-ориентированное программирование.

Практика. Создание и отладка программ ООП.

Форма подведения итогов: защита проекта.

Методическое обеспечение курса «Основы алгоритмизации и программирования»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал. Электронные источники	Техническое оснащение	Форма подведения итогов
Тема 1. Основные понятия языка программирования Паскаль.	Комбинированная	Объяснительно-иллюстративный.	Пособие для ученика «Алгоритмизация и программирование», тема № 1. Дистанционный курс «Алгоритмизация и программирование», уроки 1, 6, 7.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «Pascal».	Тестирование
Тема 2. Операторы ввода-вывода.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Пособие для ученика «Алгоритмизация и программирование», тема № 2. Дистанционный курс «Алгоритмизация и программирование», уроки 2, 8, 9.	Проекционное оборудование. Персональный компьютер. 3) Доступ к сети Интернет. Свободно распространяемое ПО «Pascal».	Контрольная работа
Тема 3. Сведения о файловой системе.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Пособие для ученика «Алгоритмизация и программирование» тема № 3. Дистанционный курс	Проекционное оборудование. Персональный компьютер. Доступ к сети	Тестирование

			«Алгоритмизация и программирование», уроки 3, 10.	Интернет. Свободно распространяемое ПО «Pascal».	
Тема 4. Ветвления на Паскале.	Комбинированная	Объяснительно-иллюстративный.	Пособие для ученика «Алгоритмизация и программирование», тема № 4. Дистанционный курс «Алгоритмизация и программирование», уроки 4, 11.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «Pascal».	Тестирование
Тема 5. Операторы цикла.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Пособие для ученика «Алгоритмизация и программирование», тема № 5. Дистанционный курс «Алгоритмизация и программирование», уроки 12-14.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «Pascal».	Контрольная работа
Тема 6. Массивы.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Пособие для ученика «Алгоритмизация и программирование», тема № 6. Дистанционный курс «Алгоритмизация и программирование»,	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно	Контрольная работа

			уроки 15-19.	распространяемое ПО «Pascal».	
Тема 7. Процедуры и функции.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Пособие для ученика «Алгоритмизация и программирование», тема № 7. Дистанционный курс «Алгоритмизация и программирование», уроки 20-21.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «Pascal».	Контрольная работа
Тема 8. Строки.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Пособие для ученика «Алгоритмизация и программирование», тема № 8. Дистанционный курс «Алгоритмизация и программирование», уроки 21-22.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «Pascal».	Контрольная работа
Тема 9. ООП.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	Практические работы.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «LAZARUS».	Защита творческого проекта.

КУРС « ПРОГРАММИРОВАНИЕ НА PYTHON»

Цель курса:

изучение основных конструкций языка Python (данных, переменных, ветвлениях, циклах и функциях), изучение базовых типов и структур данных (целые и вещественные числа, строки, списки, кортежи, словари). Разработка PyGame.

Задачи курса:

- обучить основам программирования на языке Python;
- развить интерес к информационным технологиям;
- развить алгоритмический стиль мышления;
- обучить основным принципам обработки и хранения числовой, текстовой, графической информации;
- обучить работе со стандартными библиотеками Python;
- научиться подключать и работать с дополнительными модулями языка;
- изучить основы разработки приложений на языке Python.

Форма проведения итоговой аттестации: защита творческого проекта.

Учебно-тематический план курса «Программирование на Python»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Основы ЯП Python и знакомство с IDE.	3	9	12
2	Тема 2. Циклы. Строки. Списки.	7	14	21
3	Тема 3. Словари.	3	9	12
4	Тема 4. Функции.	3	9	12
5	Тема 5. Классы.	5	10	15
6	Тема 6. Стандартные средства ЯП Python.	5	10	15
7	Тема 7. Применение ЯП Python.	3	12	15
Итого:		29	73	102

Содержание курса «Программирование на Python»

Уровень предъявления материала обеспечивает учащимся знакомство с фундаментальными основами технологии объектно-ориентированного программирования.

Учащиеся должны знать:

- основные принципы построения программ на языке;
- основные способы представления структур данных;
- базовые конструкции языка Python (операции присваивания, ветвления, цикла, ввода/вывода, описание данных, запись констант и выражений);
- основные методы работы с контейнерами данных (списками, строками, кортежами, словарями);
- алгоритмы обработки строковых переменных;
- понятие пользовательских функции, способы организации функций, принципы модульной технологии программирования;
- принципы подключения и использования дополнительных модулей языка;
- принципы разработки веб-приложений.

Учащиеся должны уметь:

- формализовать поставленную задачу;
- тестировать и отлаживать задачи с целью повышения надёжности и эффективности;
- использовать современные готовые библиотеки классов, технологии и инструментальные средства.

Формы занятий:

- фронтальная;
- индивидуальная;
- урок-консультация;
- компьютерное тестирование;
- решение задач.

Тема 1. Основы ЯП Python и знакомство с IDE.

Теория. Введение в ЯП Python. Подготовка среды программирования. Интерфейс среды разработки. Структура программы. Ввод и вывод данных. Операции с целыми и вещественными числами. Переменные. Типы данных. Логические операции и операции сравнения. Условия.

Практика. Организация ввода-вывода данных. Решение задач с целочисленными, вещественными и строковыми данными. Использование условных конструкций.

Форма подведения итогов: тестирование.

Тема 2. Циклы. Строки. Списки.

Теория. Цикл for. Цикл while. Операторы break, continue. Строки и символы. Индексы. Упорядочение списка. Работа со списками. Перебор. Создание числовых списков. Работы с частью списка. Кортежи.

Практика. Организация циклических конструкций. Написание программ с использованием списков, строк, кортежей.

Форма подведения итогов: тестирование.

Тема 3. Словари.

Теория. Создание словарей. Перебор словаря. Вложение. Использование циклов со списками и словарями.

Практика. Разработка программ с использованием словарей. Обработка словарей.

Форма подведения итогов: тестирование.

Тема 4. Функции.

Теория. Определение функции. Передача аргументов. Возвращаемые значения. Передача списков. Хранение функций в модулях. Функции и стек вызовов.

Практика. Составление программ с пользовательских функций. Решение задач с использованием функций. Работа со стеками

Форма подведения итогов: тестирование.

Тема 5. Классы.

Теория. Работа с классами и экземплярами. Наследование. Импортирование классов. Полиморфизм. Инкапсуляция.

Практика. Составление и отладка программ с использованием классов. Импорт классов.

Форма подведения итогов: тестирование.

Тема 6. Стандартные средства ЯП Python.

Теория. Модули и импорт. Итераторы и генераторы. Работа с файловой системой и файлами. Исключения. Работа с функциями.

Практика. Составление и отладка программ, использующих итераторы. Подключение файлов. Обработка файловой информации. Работа с библиотеками.

Форма подведения итогов: тестирование.

Тема 7. Применение ЯП Python.

Теория. Создание проекта игры. Добавление изображений. Определение исполняемых частей игры. Рефакторинг. Перемещение персонажей. Добавление кнопок. Ведение счета. Веб-приложения. Подготовка к созданию проекта. Построение страниц. Создание учетных записей. Редактирование данных. Оформление и развертывание приложения. Визуализация данных. Библиотеки для анализа данных. Построение графиков. Загрузка данных. Регулярные выражения. Форматы данных. Работа с API. Git и контроль версий.

Практика. Составление и отладка приложений. Создание веб-приложений. Работа с графикой.

Форма подведения итогов: тестирование.

Методическое обеспечение курса «Программирование на Python»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Основы ЯП Python и знакомство с IDE.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский..	«Изучаем Python. Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 2. Циклы. Строки. Списки.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский..	«Изучаем Python. Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 3. Словари.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	«Изучаем Python. Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 4. Функции.	Комбинированная	Объяснительно-	«Изучаем Python.	Проекционное	Тестирование

		иллюстративный. Частично-поисковый. Исследовательский.	Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	оборудование, ПК. Доступ к сети Интернет.	
Тема 5. Классы.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	«Изучаем Python. Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 6. Стандартные средства ЯП Python.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	«Изучаем Python. Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование
Тема 7. Применение ЯП Python.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	«Изучаем Python. Программирование игр, визуализация данных, веб-приложения». Эрик Мэтиз. – СПб.: Питер, 2017	Проекционное оборудование, ПК. Доступ к сети Интернет.	Тестирование

КУРС «ВЕБ-ПРОГРАММИРОВАНИЕ»

Цель курса:

- научиться создавать сайты с использованием основных веб-технологий: языка гипертекстовой разметки HTML, каскадных таблиц стилей CSS.

Задачи курса:

- рассмотреть структуру веб-страниц;
- познакомить с синтаксисом языка разметки HTML;
- научить применять стилевое оформление с использованием CSS;
- рассмотреть принцип работы веб-сервера;
- научить разрабатывать динамические сайты с использованием системы управления сайтов.

Режим занятий – один раз в неделю по три учебных часа.

Форма реализации курса – очная.

Форма проведения итоговой аттестации – персональный творческий отчёт «Защита проекта».

Учебно-тематический план курса «ВЕБ-ПРОГРАММИРОВАНИЕ»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Введение в веб-программирование.	1	2	3
2	Тема 2. Язык гипертекстовой разметки HTML.	5	10	15
3	Тема 3. Каскадные таблицы стилей CSS.	8	16	24
4	Тема 4. Адаптивная верстка. Построение сеток.	7	14	21
5	Тема 5. Динамические сайты. Работа с CMS.	10	20	30
6	Работа над итоговым проектом.		6	6
7	Защита проекта.	3		3

	Итого:	34	68	102
--	--------	----	----	-----

Содержание курса «ВЕБ-ПРОГРАММИРОВАНИЕ»

Уровень предъявления материала обеспечит учащимся возможность создать собственный динамически генерируемый сайт в сети Интернет.

Учащиеся должны знать:

- язык гипертекстовой разметки HTML;
- каскадные таблицы стилей CSS;
- принцип построения адаптивных сайтов.

Учащиеся должны уметь:

- использовать элементы языка HTML и создавать корректные HTML-документы;
- оформлять HTML-документы, используя CSS;
- разрабатывать сайт с системой управления контентом.
- Формы занятий:
 - фронтальная;
 - индивидуальная;
 - беседа-обсуждение творческих заданий;
 - защита проекта.

Тема1. Введение в веб-программирование.

Теория. World Wide Web, протокол HTTP, типы сайтов, проектирование сайта.

Практика. Практическое применение полученных знаний. Разработка структуры сайта.

Форма подведения итогов. тестирование.

Тема2. Язык гипертекстовой разметки HTML.

Теория. Структура HTML-документа, теги, таблицы, формы.

Практика. Создание HTML-документа.

Форма подведения итогов. тестирование.

Тема3. Каскадные таблицы стилей CSS.

Теория. Назначение и применение CSS, блочные и строковые элементы, свойства элементов, цвет и шрифт, позиционирование, слои.

Практика. Создание HTML-документов, оформленных при помощи CSS.

Форма подведения итогов. тестирование.

Тема4. Адаптивная верстка. Построение сеток.

Теория. Блочная модель документа, сетки, таблицы, flexbox, грид-элементы.

Практика. Создание HTML-документов со сложной сеткой и декоративными элементами.

Форма подведения итогов. тестирование.

Тема5. Динамические сайты. Работа с CMS.

Теория. Система управления сайтом, авторизация, создание шаблонов, размещение сайта на хостинге, загрузка файлов на сайт с использованием форм, постраничное представление информации.

Практика. Практическое применение полученных знаний.

Методическое обеспечение курса «Веб-программирование»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Введение в веб-программирование.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Справочный материал по теме.	Персональные компьютеры. Презентационное оборудование.	Тестирование
Тема 2. Язык гипертекстовой разметки HTML.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Справочный материал по теме.	Персональные компьютеры. Презентационное оборудование.	Тестирование
Тема 3. Каскадные таблицы стилей CSS.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Справочный материал по теме.	Персональные компьютеры. Презентационное оборудование.	Тестирование
Тема 4. Адаптивная верстка. Построение сеток.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Справочный материал по теме;	Персональные компьютеры. Презентационное оборудование.	Тестирование
Тема 5. Динамические сайты. Работа с CMS.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Справочный материал по теме.	Персональные компьютеры. Презентационное оборудование.	творческий проект

КУРС «ЯЗЫК ПРОГРАММИРОВАНИЯ СИ++»

Цели курса:

- на примере языка программирования C/C++ раскрыть суть фундаментальных основ технологии программирования;
- сформировать понимание – с какой целью и каким образом можно создавать программы на языке программирования C/C++, понять суть объектно-ориентированной парадигмы программирования.

Задачи курса:

- обучить технологии работы на персональном компьютере в программной среде GNU C++;
- обучить методам реализации основных алгоритмических конструкций на языке программирования C/C++, создавая программы из различных научных областей.

Режим занятий – один раз в неделю по 4 учебных часа.

Форма реализации курса – очная.

Форма проведения итоговой аттестации – контрольная работа.

Учебно-тематический план курса «Язык программирования C/C++»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
Тема 1. Введение. Управляющие инструкции.		20	28	48
	Введение в программирование на C++. Структура программы на C++.	2		2
	Типы данных языка C++.	2		2
	Организация ввода-вывода.	4		4
	Решение задач линейного программирования.		4	4
	Тест №1 «Решение линейных программ».		4	4

	Операции языка (операция присваивания, операция инкремента и декремента).	2		2
	Структура выбора if/else, тернарная операция.	2		2
	Решение задач с использованием структура выбора if/else.		4	4
	Структура множественного выбора switch? Решения задач.		4	4
	Структура повторения while.	2		2
	Структура повторения for, примеры использования структуры, особенности выражений оператора for.	2		2
	Структура повторения do/while, решение задач. Операторы break и continue пример решения задач.		4	4
	Вложенные циклы. Ошибки использования структур повторений.	4		4
	Вложенные циклы, решение задач.		4	4
	Тест №2. «Структурное программирование».		4	4
Тема 2. «Функции».		4	16	20
	Функции, понятие, определение, прототипы, заголовочные файлы.	4		4
	Генерация случайных чисел.		4	4
	Рекурсия, пример использования рекурсии.		4	4
	Перегрузка функции, решение задач. Решение задач с использованием		4	4

	нескольких функций.			
	Решение задач с использованием функций.		2	2
	Тест №3. «Функции».		2	2
Тема 3. Массивы, указатели, динамические массивы. Строки.		24	32	56
	Введение, объявление одномерных массивов, пример решения задачи.	4		4
	Передача массивов в функции, решение задач.	4		4
	Сортировка массивов, поиск в массивах, виды.	4		4
	Алгоритмы быстрой сортировки.		4	4
	Решение задач на сортировки.		4	4
	Многомерные массивы, пример.	4		4
	Решение задач с использованием многомерных массивов.		4	4
	Тест №4. «Решение задач с использованием массивов».		4	4
	Объявления и инициализация переменных – указателей, операции над указателями.	4		4
	Выражения и арифметические действия с указателями.		4	4
	Массивы указателей, пример использования.	4		4
	Введение в обработку символов и строк.		4	4
	Решение задач с		4	4

	использованием массивов.			
	Тест №5. «Использование указателей».		4	4
Тема 4. Обработка файлов.			12	12
	Произвольная запись, чтение данных из файла, пример использования.		4	4
	Решение задач.		4	4
	Тест №7. «Использование файлов».		2	2
	Итоговая контрольная работа.		2	2
	Анализ итоговой контрольной работы.			
	Резерв времени.			
Итого:		48	88	136

Содержание курса «Язык программирования C++»

Уровень предъявления материала обеспечивает учащимся знакомство с фундаментальными основами технологии программирования.

Учащиеся должны знать:

- базовые конструкции языка программирования C/C++ (операции присваивания, ветвления, цикла, ввода/вывода, описание данных, запись констант и выражений);
- основные методы сортировки массивов (метод простого выбора, метод простого обмена, метод простой вставки);
- алгоритмы обработки символьных переменных;
- понятие подпрограммы, функции пользователя;
- понятие формальных и фактических параметров;
- способы организации процедур (подпрограммы) и функций, принципы модульной технологии программирования;
- алгоритмы обработки массивов (поиск элемента с заданными свойствами, удаление элементов, вставка элементов, замена элементов, обмен местами групп элементов);
- технологию разработки программ методом детализации.

Учащиеся должны уметь:

- читать готовые программы по образцу;
- записать конструкции языка программирования Си;
- составлять, отлаживать и исполнять на ЭВМ программы решения учебных задач по программированию;
- применять методы вычислений для решения задач.

Формы занятий, используемые при изучении данной темы:

- фронтальная;
- индивидуальная;
- урок-консультация;
- компьютерное тестирование;
- контрольная работа.

Тема 1. Введение. Управляющие инструкции.

Теория. Структура программы, типы данных, организация ввода-вывода, операции языка (операция присваивания, операция инкремента и декремента), структура выбора if/else, тернарная операция, структура повторения while, структура повторения for, структура повторения do/while, структура множественного выбора switch.

Практика. Типы операций над типами данных, решение задач линейного программирования и организация форматированного ввода-вывода, решение задач с использованием структура выбора if/else, решение задач с использованием циклических структур while, for, do\while, решение задач с использованием структуры switch.

Форма подведения итогов: контрольная работа.

Тема 2. Функции.

Теория. Понятие и определение функции, Рекурсия, пример использования рекурсии.

Практика. Прототипы функций, заголовочные файлы, генерация случайных чисел, решение задач с использованием рекурсии, перегрузка функции, решение задач, решение задач с использованием нескольких функций.

Форма подведения итогов: контрольная работа.

Тема 3. Массивы, указатели, динамические массивы. Строки

Теория. Введение, объявление одномерных массивов, алгоритмы обработки одномерных массивов, сортировка массивов, поиск в массивах с использованием функций, многомерные массивы, объявления и

инициализация переменных – указателей, операции над указателями, массивы указателей, динамический массив, строки в C++, основы представления строк, библиотечные функции обработки строк.

Практика. Решение задач обработки одномерных массивов, передача массивов в функции, сортировка массивов, поиск в массивах, функции malloc, calloc и realloc, операции new и delete, решение задач на двумерные динамические массивы, Решение задач обработки строк.

Форма подведения итогов: контрольная работа.

Тема 4. Обработка файлов

Теория. Иерархия данных. Файлы и потоки, создание файла последовательного доступа, последовательного и произвольного доступа.

Практика. Произвольная запись, чтение данных из файла, пример использования, решение задач.

Форма подведения итогов: контрольная работа.

Методическое обеспечение курса «Язык программирования C++»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Введение. Управляющие инструкции.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Дистанционный справочный курс (informatics.mccme.ru) . Комплекс интернет-олимпиад (contest.stavliders.ru и contest.stavpoisk.ru). Лицензионное ПО Adobe Flash CS3 Profession.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «MinGW Developer Studio».	Контрольная работа
Тема 2. Функции.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Дистанционный справочный курс (informatics.mccme.ru) . Комплекс интернет-олимпиад (contest.stavliders.ru и contest.stavpoisk.ru). Лицензионное ПО Adobe Flash CS3 Profession.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «MinGW Developer Studio».	Контрольная работа
Тема 3.	Комбинированная	Объяснительно-	Дистанционный	Проекционное	Контрольная

Массивы, указатели, динамические массивы. Строки.		иллюстративный. Частично-поисковый.	справочный курс (informatics.mccme.ru) . Комплекс интернет-олимпиад (contest.stavliders.ru и contest.stavpoisk.ru). Лицензионное ПО Adobe Flash CS3 Profession.	оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «MinGW Developer Studio».	работа
Тема 4. Обработка файлов.	Комбинированная	Частично-поисковый. Исследовательский.	Дистанционный справочный курс (informatics.mccme.ru) . Комплекс интернет-олимпиад (contest.stavliders.ru и contest.stavpoisk.ru). Лицензионное ПО Adobe Flash CS3 Profession.	Проекционное оборудование. Персональный компьютер. Доступ к сети Интернет. Свободно распространяемое ПО «MinGW Developer Studio».	Контрольная работа

КУРС «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Цели курса:

- познакомить учащихся с принципом создания серверных приложений на платформе android;
- привлечь к исследовательской и изобретательской деятельности;
- формировать мотивацию к занятиям программированием и созданию своих приложений.

Задачи курса:

- познакомить с понятием android-приложения, оценить различные среды программирования данных приложений, выявить достоинства и недостатки каждой среды;
- погрузить обучающихся в проектную деятельность для формирования навыков создания собственного проекта;
- научить писать приложения, подключая базы данных для мобильных устройств;
- научить тестировать собственные приложения, находить недостатки в работе и исправлять их;
- научить обучающихся определять уровень потребности создания того или иного проекта;
- дать основные навыки работы с необходимым для программирования программным обеспечением;
- научить создавать приложения различных уровней сложности;
- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- развивать навыки эффективной деятельности в проекте, успешной работы в команде;
- развивать стрессоустойчивость;
- развивать способности к самоанализу, самопознанию;
- формировать навыки рефлексивной деятельности.

Режим занятий: один раз в неделю по два учебных часа.

Форма реализации курса: очная.

Форма проведения итоговой аттестации: по окончании обучения проводится итоговая аттестация в форме публичной защиты проектов.

**Учебно-тематический план курса
«РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»**

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Тема 1. Введение в язык программирования Java	8	6	14
2	Тема 2. Создание простейших Android-приложений	8	8	16
3	Тема 3. Расширение возможностей Android-приложения	8	8	16
4	Тема 4. Базы данных и интернет-взаимодействие	8	6	14
5	Тема 5. Работа над проектом	2	6	8
Итого:		34	34	68

Содержание курса «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Курс знакомит с основными особенностями объектно-ориентированного программирования, основами создания приложений на платформе android, базовыми знаниями основ бытового программирования и подключения баз данных.

Учащиеся должны:

- пользовательский интерфейс профильного ПО, базовых объектов инструментария;
- основы алгоритмизации и объектно-ориентированного программирования.
- принципы работы локальных СУБД на примере SQLite;
- знать основы программирования интернет-взаимодействия на java.

Уметь:

- создавать простые приложения на языке java, используя платформу android;
- свободно использовать среду разработки;
- подключать базы данных;
- организовывать интернет-взаимодействие компьютерных программ.

- генерировать идеи;
- высказываться устно в виде рецензии ответа товарища;
- представлять одну и ту же информацию различными способами;
- слушать и слышать собеседника;
- аргументированно отстаивать свою точку зрения;
- искать информацию в свободных источниках и структурировать ее;
- эффективно работать в команде;
- работать с программами по созданию презентаций (MS PowerPoint или LibreOffice Impress);
- объективно оценивать результаты своей работы.

Формы занятий:

- фронтальная;
- индивидуальная;
- лекционная,
- групповая (командная) работа,
- беседа-обсуждение;
- групповые консультации;
- защита проекта.

Тема 1. Введение в язык программирования Java

Теория. Объектно-ориентированное программирование на Java, среда IntelliJ Idea Community Edition разработки Java-программ.

Практика. Практическое применение полученных знаний. Разработка объектно-ориентированных программ на языке Java.

Форма подведения итогов: тестирование, демонстрация работы компьютерных программ.

Тема 2. Создание простейших Android-приложений

Теория. Архитектура операционной системы Android, среда Android Studio разработки android-приложений, разметка android-приложения, управление кнопками, виджеты ввода-вывода текстовой информации android-приложения, загрузка android-приложения в мобильное устройство.

Практика. Создание простейших android-приложений.

Форма подведения итогов: тестирование, демонстрация работы компьютерных программ.

Тема 3. Расширение возможностей Android-приложения

Теория. Массивы и списки в Java, виджеты обработки списков, адаптеры, многопоточность, управление активностями android-приложения, двумерная графика в мобильном android-приложении.

Практика. Создание android-приложений на языке Java.

Форма подведения итогов: тестирование, демонстрация работы компьютерных программ.

Тема 4. Базы данных и интернет-взаимодействие

Теория. СУБД SQLite, создание и подключение СУБД к android-приложению, простейший интернет-клиент на языке Java, простейший интернет-сервер.

Практика. Создание android-приложений, содержащих СУБД и интернет-взаимодействие.

Форма подведения итогов: тестирование, демонстрация работы компьютерных программ.

Тема 5. Работа над проектом

Теория. Основные стадии создания индивидуального проекта. Техническое задание, компьютерная программа, пояснительная записка, презентация. Приемы защиты проекта.

Практика. Создание технического задания, компьютерной программы и презентации индивидуального проекта. Репетиция процесса защиты проекта.

Форма подведения итогов: защита индивидуального проекта.

Методическое обеспечение курса «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Тема 1. Введение в язык программирования Java	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Электронный учебник с встроенной тестирующей системой	Компьютеры с монитором, клавиатурой, системным блоком и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: ОС Windows; пакет офисных программ MS Office, среда разработки eclips; среда разработки android studio	Тестирование
Тема 2. Создание простейших Android-приложений	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Электронный учебник с встроенной тестирующей системой	Компьютеры с монитором, клавиатурой, системным блоком и мышкой, и доступом к сети Интернет, на	Проект

				которых установлено следующие программное обеспечение: ОС Windows; пакет офисных программ MS Office, среда разработки eclips; среда разработки android studio	
Тема 3. Расширение возможностей Android-приложения	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Электронный учебник с встроенной тестирующей системой	Компьютеры с монитором, клавиатурой, системным блоком и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: ОС Windows; пакет офисных программ MS Office, среда разработки eclips; среда разработки android studio	Проект

Тема 4. Базы данных и интернет-взаимодействие	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Электронный учебник с встроенной тестирующей системой	Компьютеры с монитором, клавиатурой, системным блоком и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: ОС Windows; пакет офисных программ MS Office, среда разработки eclips; среда разработки android studio	Проект
Тема 5. Работа над проектом	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый.	Электронный учебник с встроенной тестирующей системой	Компьютеры с монитором, клавиатурой, системным блоком и мышкой, и доступом к сети Интернет, на которых установлено следующие программное обеспечение: ОС Windows; пакет офисных программ	Защита итогового проекта

				MS Office, среда разработки eclips; среда разработки android studio	
--	--	--	--	---	--

КУРС «РАЗРАБОТКА ИГР НА UNITY»

Цель курса:

организовать учебно-познавательный процесс при помощи проектной деятельности на основе игрового движка Unity, среды для разработки 2D, 3D, VR и AR интерактивного контента, компьютерных и мобильных игр.

Задачи курса:

- познакомить с основными понятиями геймдизайна;
- познакомить с базовыми инструментами движка Unity;
- сформировать навык разработки игр с помощью Unity;
- сформировать навык использования языка программирования C# при разработке игр;
- сформировать навык понимания разных игровых стратегий и умения их проектировать;
- сформировать навык использования итерационного подхода при решении различных задач;
- формировать навык разработки сюжетной линии игр;
- формировать навык продвижения игр на платформе и привлечения новых пользователей;
- формировать навык тестирования и оптимизации созданных проектов;
- формировать навык кроссплатформенной разработки;
- формировать и развивать логическое, алгоритмическое, критическое, креативное и пространственное мышление;
- формировать навык публичного выступления и презентации;
- формировать навык планирования деятельности и основ тайм-менеджмента;
- формировать чувство цвета и стиля при оформлении материалов (эстетического вкуса);
- формировать и расширять словарный запас на английском языке;
- расширять кругозор, развивать память, внимание, творческое воображение, абстрактно-логические и наглядно-образные виды мышления и типы памяти, основные мыслительные операции и свойства внимания;
- совершенствовать диалогическую речь учащихся: уметь слушать собеседника, понимать вопросы, смысл знаний, уметь задавать вопросы, отвечать на них.

Режим занятий: один раз в неделю по два учебных часа.

Форма реализации курса: очная.

Форма проведения итоговой аттестации: персональный творческий проект.

Учебно-тематический план курса «РАЗРАБОТКА ИГР НА UNITY»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Модуль 1. Основы Unity	5	5	10
2	Модуль 2. Дизайн 3D-уровней	4	4	8
3	Модуль 3. Программирование видеоигр на языке C#	10	10	20
4	Модуль 4. Анимация 3D-объектов	4	4	8
5	Модуль 5. Специфика 2D-игр. Мобильная платформа.	4	4	8
6	Модуль 6. Создание гоночной 3D-игры.	4	4	8
7	Модуль 7. Создание ролевой 3D-игры.	4	4	8
Итого:		34	34	68

Содержание курса «РАЗРАБОТКА ИГР НА UNITY»

Уровень предъявления материала обеспечивает учащимся возможность самостоятельно создавать творческие проекты 2D, 3D, VR и AR интерактивного контента, компьютерных и мобильных игр в среде Unity.

Учащиеся должны знать:

- основные объекты среды Unity;
- понятия: геймдизайн, геймдизайнер, дизайн-документ, сценарий игры, скрипт, система координат;
- понятие команды и входных параметров;
- понятие алгоритм и программа;
- правила оформления программы.

Учащиеся должны уметь:

- использует различные инструменты редактора Unity;
- составлять дизайн-документы с описанием игры;
- создавать сценарий для будущей игры;
- создавать описание для каждого уровня и карту переходов между ними;
- продумывать дизайн и оформлять уровни;

- работать по заранее составленному сюжету;
- писать логику видеоигры на языке C#;
- создавать CG-ролики;
- настраивать скрипты для кроссплатформенной разработки.

Формы занятий:

- фронтальная;
- индивидуальная;
- беседа-обсуждение;
- конкурс;
- защита проекта.

Модуль 1. Основы Unity

Теория. Интерфейс редактора Unity. Редактирование положения и формы примитивов — компонент Transform. Игровые объекты и ассеты. 3D-пространства от 2D. Координатная система сцены и игрового объекта. Вложенные объекты. Иерархия игровых объектов. Префабы и экземпляры. Принцип наследования. 3D-рендеринг. Пиксели. Растровый рисунок. 3D-модель объекта. Полигональная сетка. Триангуляция. Низкополигональные и высокополигональные 3D-модели. Параметры материала 3D-объекта. Шейдер. Текстура.

Практика. Создание игровой сцены с несколькими 3D-объектами. Настройка иерархии игровых объектов и её расширение за счёт экземпляров префабов. Создание визуального стиля сцены. Создание плана уровня игры. Импорт ассетов из Unity Asset Store. Тестирование прототипа. Создание WebGL-билда проекта

Форма подведения итогов. Презентация проекта.

Модуль 2. Дизайн 3D-уровней

Теория. Левел-дизайн. Объект Terrain. Формирование ландшафта с помощью карты высот (Heightmap). Настройка кистей для работы с ландшафтом. Terrain Layer. Инструмент Paint Texture, Paint Trees, Paint Details и Terrain Setting. Источника света. Визуальные и звуковые эффекты.

Практика. Создание ландшафта для игрового уровня. Детализация ландшафта. Создание определённой атмосферы видеоигры с помощью средств светового дизайна. Усиление атмосферы видеоигры с помощью световых и звуковых эффектов.

Форма подведения итогов. Презентация проекта.

Модуль 3. Программирование видеоигр на языке C#

Теория. Механики 3D-платформера. Понятия «язык программирования», «алгоритм», «программа», «скрипт». Интегрированная среда разработки Visual Studio. Синтаксис языка C#. Структура шаблона скрипта в Unity. Переменные и типы данных. Классы объектов. Функции и методы. Изменение параметров 3D-объектов. Условные конструкции. Интерфейс игрока. Модульная разработка.

Практика. Разработка скриптов осуществляющих вывод данных, обработку классов. Создание и тестирование скриптов, меняющих свойства 3D-объектов, содержащих условный оператор. *Создание скриптов с учётом модульной разработки.*

Форма подведения итогов. Презентация проекта.

Модуль 4. Анимация 3D-объектов

Теория. 3D-анимация. Этапы создания 3D-анимации в Unity. Настройка анимации с помощью окна Animation. Запись анимации — инструмент Record Mode. Создание плавных переходов в анимации — инструмент Curves Editor. Подготовка 3D-модели для анимации. Скелет (rig) 3D-модели. Система Mecanim. Работа с объектом Animator Controller. Создание переходов между состояниями персонажа в окне Animator. Написание скриптов для анимации объектов в зависимости от игрового события.

Практика. Создание анимации нескольких состояний персонажа видеоигры. Создание анимации игровых объектов под конкретные события в видеоигре. Работа с Timeline

Форма подведения итогов. Презентация проекта.

Модуль 5. Специфика 2D-игр. Мобильная платформа

Теория. Введение в 2D-игры. Особенности физики в 2D-играх. Rigidbody 2D. Коллайдеры 2D-объектов. Механики на основе физики 2D-объектов. Особенности 2D-анимации. Особенности мобильных платформ.

Практика. Создание дизайн-документа и первого прототипа 2D-шутера. Создание анимации спрайтов для 2D-шутера. Создание спрайт-листа (sprite sheet). Работа со слоями спрайтов. Написание логики 2D-шутера. Создание билда под мобильную платформу Android, 2D-шутера для мобильной платформы Android.

Форма подведения итогов. Презентация проекта.

Модуль 6. Создание гоночной 3D-игры

Теория. Знакомство с жанром гоночных игр. Принципы дизайна уровней.

Основные механики гоночной игры. Особенности создания пользовательского интерфейса для гоночной игры. Запись геймплейного ролика.

Практика. Создание дизайн-документа для гоночной 3D-игры. Создание препятствий. Физический движок. Создание скриптов для реализации основных механик гоночной игры. Создание 3D-игры.

Форма подведения итогов. Презентация проекта.

Модуль 7. Создание ролевой 3D-игры

Теория. Игры в жанре RPG. Поворот и атака игрока. Перемещение и атака врага. Особенности жанра «ролевая игра». Планирование разработки ролевой игры. Принципы дизайна уровней для ролевой игры. Основные механики ролевой игры. Особенности создания пользовательского интерфейса для ролевой игры.

Практика. Создание игровых уровней для ролевой игры. Создание скриптов для реализации основных механик ролевой игры. Создание ролевой 3D-игры.

Форма подведения итогов. Презентация проекта.

Выпускной

Теория. Создание портфолио Unity-разработчика. Правила создания портфолио разработчика видеоигр. Инструменты создания цифрового портфолио.

Практика. Создание цифрового портфолио с описанием всех созданных на курсе проектов.

Форма подведения итогов. Презентация проекта.

Методическое обеспечение курса «РАЗРАБОТКА ИГР НА UNITY»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение	Форма подведения итогов
Модуль 1. Основы Unity	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.
Модуль 2. Дизайн 3D-уровней	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.
Модуль 3. Программирование видеоигр на языке C#	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub,	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.

			4) ПО Visual Studio		
Модуль 4. Анимация 3D-объектов	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.
Модуль 5. Специфика 2D-игр. Мобильная платформа.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.
Модуль 6. Создание гоночной 3D-игры.	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.
Модуль 7. Создание ролевой	Комбинированная	Объяснительно-иллюстративный.	1) платформа «Алгоритмика»;	Персональные компьютеры.	Презентация проекта.

3D-игры.		Частично-поисковый. Исследовательский.	2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Презентационное оборудование. Доступ к сети Интернет.	
Выпускной	Комбинированная	Объяснительно-иллюстративный. Частично-поисковый. Исследовательский.	1) платформа «Алгоритмика»; 2) Исходные файлы для создания проектов; 3) ПО Unity Hub, 4) ПО Visual Studio	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Презентация проекта.

КУРС «ОСНОВЫ МОБИЛЬНОЙ РАЗРАБОТКИ»

Цели курса:

- научиться анализировать пользовательские задачи и выбирать алгоритмические решения;
- сформировать понимание ключевых понятий программирования;
- научиться проектировать пользовательский интерфейс.

Задачи курса:

- освоение базовой терминологии в области мобильной разработки;
- формирование представлений об архитектуре мобильных приложений;
- обучение основам алгоритмического мышления;
- развитие навыков проектирования пользовательского интерфейса;
- освоение работы с сенсорами, камерой, геолокацией, мультимедиа.

Режим занятий: шесть раз в неделю по два учебных часа.

Форма реализации курса: очная.

Форма проведения итоговой аттестации: итоговая работа.

Учебно-тематический план курса «Основы мобильной разработки»

№	Наименование раздела, темы	Количество часов		
		Теория	Практика	Всего
1	Введение в мобильную разработку	2		2
2	Интерфейс среды разработки		2	2
3	Переменные и данные. Работа с текстом, числами, списками		2	2
4	Условные конструкции. Логические выражения	2	2	4
5	Циклы. Повторение действий		2	2
6	Работа с мультимедиа: изображения, GIF, звук, видео		2	2

7	Взаимодействие с устройством: камера, датчики		2	2
8	Работа с базами данных	2	2	4
9	Функции	2	2	4
	Итого:	8	16	24

Содержание курса «Основы мобильной разработки»

Курс предназначен для детей, желающих развить свои умственные способности, получить углубленные теоретические и практические знания и навыки в области мобильной разработки.

Учащиеся должны знать:

- основные типы мобильных платформ и принципы их работы;
- интерфейс визуальной среды разработки;
- понятие переменной и ее назначение;
- принцип работы условных операторов;
- понятие цикла и его назначение;
- способы подключения и использования мультимедийных файлов;
- основные возможности мобильных устройств: камера, акселерометр, гироскоп;
- понятие локального хранилища данных в мобильных приложениях;
- понятие функции и ее роль в разработке.

Учащиеся должны уметь:

- создавать простые интерактивные экраны с кнопками и текстовыми полями, используя визуальные инструменты;
- добавлять и настраивать основные компоненты на экране;
- создавать и инициализировать переменные различных типов данных;
- формировать и использовать логические выражения для сравнения чисел и текста;
- применять циклы для выполнения повторяющихся действий;
- подключать и отображать изображения и GIF;
- воспроизводить звуковые эффекты по событию;
- использовать камеру устройства для распознавания QR-кодов;
- реализовывать сохранение и загрузку данных в локальном хранилище приложения;
- создавать собственные функции для выполнения определенных задач.

Формы занятий, используемые при изучении данной темы:

- фронтальная;
- индивидуальная;
- беседа-обсуждение творческих заданий.

Тема 1. Введение в мобильную разработку .

Теория. Основные типы мобильных платформ и принципы их работы. Отличие мобильных приложений от сайтов и программ для ПК. Как создавать простые приложения без написания кода, используя визуальные инструменты.

Практика. Создание первого интерактивного экрана с кнопкой и текстовым полем. Настройка реакции на нажатие кнопки (вывод приветствия).

Форма подведения итогов: устный опрос.

Тема 2. Интерфейс среды разработки.

Теория. Знакомство с интерфейсом визуальной среды разработки. Основные компоненты: кнопка, текстовое поле, метка, переключатель, ползунок. Понятие экрана и навигация между экранами.

Практика. Добавление компонентов на экран. Настройка свойств (цвет, размер, текст). Создание приложения с двумя экранами и кнопками «Далее»/«Назад».

Форма подведения итогов: проверка практического задания.

Тема 3. Переменные и данные. Работа с текстом, числами, списками.

Теория. Понятие переменной. Типы данных: числа, текст, списки. Присваивание и изменение значений переменных. Отображение переменных на экране.

Практика. Создание приложения-счётчика (нажатие на кнопку увеличивает число). Создание простого калькулятора (сложение, вычитание).

Форма подведения итогов: проверка практического задания.

Тема 4. Условные конструкции. Логические выражения.

Теория. Понятие условия. Логические выражения: больше, меньше, равно. Сравнение чисел и текста.

Практика. Создание приложения-угадайки (пользователь вводит число, программа говорит «больше», «меньше», «угадал»). Создание приложения-проверки пароля.

Форма подведения итогов: проверка практического задания.

Тема 5. Циклы. Повторение действий.

Теория. Понятие цикла. Виды циклов: цикл с заданным количеством повторений, цикл с условием.

Практика. Создание приложения, которое выводит числа от 1 до 10. Генератор последовательностей.

Форма подведения итогов: проверка практического задания.

Тема 6. Работа с мультимедиа: изображения, GIF, звук, видео.

Теория. Подключение мультимедийных файлов. Воспроизведение звука по событию.

Практика. Создание приложения с фоновой музыкой (вкл/выкл). Добавление звука на нажатие кнопки. Создание простого аудиоплеера.

Форма подведения итогов: проверка практического задания.

Тема 7. Взаимодействие с устройством: камера, датчики.

Теория. Возможности мобильных устройств: камера, акселерометр, гироскоп. Обработка событий от датчиков.

Практика. Создание приложения, распознающего QR-коды с помощью камеры. Создание приложения, реагирующего на «встряхивание» устройства (выполнение заданного действия).

Форма подведения итогов: проверка практического задания.

Тема 8. Работа с базами данных.

Теория. Понятие локального хранилища данных. Способы сохранения и загрузки информации между сеансами работы. Типы хранилищ: текстовые файлы, встроенные базы данных.

Практика. Создание приложения «Блокнот» или «Список дел» с сохранением данных между запусками. Корректное восстановление содержимого после перезапуска.

Форма подведения итогов: проверка практического задания.

Тема 9. Функции.

Теория. Понятие и назначение функций. Встроенные функции. Пользовательские функции: создание, вызов, передача параметров, возврат результата.

Практика. Создание функции, которая очищает текстовое поле и сбрасывает счётчик (вызов по кнопке «Сброс»). Замена повторяющихся блоков кода на вызов функции. Организация модульной структуры приложения.

Форма подведения итогов: самостоятельная работа.

Методическое обеспечение курса «Основы мобильной разработки»

Раздел, тема	Форма занятия	Приемы и методы организации образовательного процесса	Дидактический материал. Электронные источники	Техническое оснащение	Форма подведения итогов
Тема 1. Введение в мобильную разработку	Комбинированная	Объяснительно-иллюстративный, репродуктивный	Презентация «Что такое мобильное приложение»	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Устный опрос
Тема 2. Интерфейс среды разработки	Практико-ориентированная	Обучение через действие, пошаговое моделирование	Инструкционные листы, визуальная среда разработки	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания
Тема 3. Переменные и данные	Комбинированная	Объяснительно-иллюстративный, репродуктивный	Схемы «Типы данных», практический тренажёр	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания
Тема 4. Условные конструкции	Практико-ориентированная	Проблемное обучение, решение задач по образцу	Блок-схемы условий, карточки с логическими задачами	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания

Тема 5. Циклы	Комбинированная	Частично-поисковый, проектный	Анимированные схемы циклов, шаблоны	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания
Тема 6. Работа с мультимедиа	Практическая лабораторная работа	Обучение через создание продукта	Набор мультимедийных файлов	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания
Тема 7. Взаимодействие с устройством	Практико-исследовательская	Экспериментальный метод, проблемное изложение	Мобильные устройства с сенсорами	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания
Тема 8. Работа с базами данных	Комбинированная	Информационно-рецептивный, репродуктивный	Схема «Как сохраняются данные?», шаблон «Блокнот»	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Проверка практического задания
Тема 9. Функции	Комбинированная	Пошаговое моделирование, частично-поисковый	Таблица «Встроенные и пользовательские функции»	Персональные компьютеры. Презентационное оборудование. Доступ к сети Интернет.	Самостоятельная работа

КУРС «УМНЫЙ ДОМ»

Цели курса:

- обучить принципам построения и программирования системы «умный дом»;
- сформировать мотивацию к занятиям по программированию и созданию своих устройств и систем умного дома.

Задачи курса:

- погрузить обучающихся в проектную деятельность для формирования навыков создания собственного проекта;
- научить обучающихся определять уровень потребности создания того или иного проекта;
- дать основные навыки работы с необходимым для программирования программным обеспечением;
- научить создавать системы умного дома с помощью микроконтроллеров Arduino различных уровней сложности;
- воспитать мотивацию учащихся к изобретательству, созданию собственных программных реализаций;
- привить стремление к получению качественного законченного результата в проектной деятельности;
- развивать навыки эффективной деятельности в проекте, успешной работы в команде;
- развивать стрессоустойчивость;
- развивать способности к самоанализу, самопознанию;
- формировать навыки рефлексивной деятельности.

Режим занятий: один раз в неделю по три учебных часа.

Форма реализации курса: очная.

Форма проведения итоговой аттестации: по окончании обучения проводится итоговая аттестация в форме публичной защиты проектов.

Учебный план курса «Умный дом»

№ темы	Наименование раздела, темы	Контактная работа обучающихся с преподавателем, ч.		
		Теория	Практика	Всего

	Модуль 1. Основы программирования Arduino.	17	49	66
1	Контроллеры и датчики IoT.	14	37	51
2	Основные алгоритмические конструкции.	3	12	15
	Модуль 2. Агро-смарт.	12	24	36
3	Умная теплица.	6	12	18
4	Умная метеостанция.	6	12	18
Итого:		29	73	102

Содержание курса «Умный дом»

В курсе рассматриваются приёмы программирования платы Arduino для создания функциональной умной теплицы и функциональной умной метеостанции.

В результате освоения учебного курса обучающийся должен:

знать:

- значение понятий «умные вещи», «смарт-объекты», «контроллер», «IDE», «облако», «интернет вещей»;
- разницу между IoT и IIoT;
- основные компоненты систем «умная теплица», «умная метеостанция»;
- основные алгоритмические конструкции для программирования платы Arduino;
- набор основных датчиков для организации умной теплицы;
- набор основных датчиков для организации умной метеостанции;

уметь:

- определять задачи, которые должен выполнять умный объект;
- подбирать датчики для эффективного выполнения умным объектом поставленных задач;
- строить основные и сложные алгоритмические конструкции для программирования платы Arduino;
- программировать компоненты умной теплицы и умной метеостанции;
- создавать элементы умной метеостанции с использованием датчиков и элементов управления Arduino, отслеживающей основные показатели погоды;
- создавать элементы умной теплицы для выращивания микрозелени с использованием датчиков и элементов управления Arduino.

Модуль 1. Основы программирования Arduino.

Тема 1. Контроллеры и датчики IoT.

Теория. Датчики освещённости, температуры, влажности, уровня жидкости, влажности почвы, газа/дыма, пламени, вибрации. Инфракрасный и ультразвуковой дальномеры. Инфракрасный датчик движения. Матричная клавиатура. Герконы. Реле. ИК-приёмник. Модуль часов реального времени.

Сервопривод. Водяная помпа. Модуль Wi-Fi. Приемник инфракрасного излучения и встроенный светодиод. Слот для SD-карты.

Практика. Программирование изученных компонентов.

Форма подведения итогов: тестирование.

Тема 2. Основные алгоритмические конструкции.

Теория. Переменные, арифметические операции. Макетная плата, светодиоды, кнопки. Оператор условия IF. Цикл FOR. Оператор CASE. Работа со звуком. Оператор TONE.

Практика. Создание моделей с использованием технологии IoT.

Форма подведения итогов: тестирование.

Модуль 2. Агро-смарт.

Тема 3. Умная теплица.

Теория. Управление освещением в теплице. Ручное управление проветриванием и поливом. Управление тепличной форточкой. Создание системы управления микроклиматом в теплице. Мониторинг ультрафиолетового излучения. Создание бегущей строки. Визуализация данных.

Практика. Программирование макета «Умная теплица».

Форма подведения итогов: защита решения кейса.

Тема 4. Умная метеостанция.

Теория. Осадкомер. Анемометр. Флюгер. Датчик температуры, влажности воздуха и атмосферного давления. Датчик ультрафиолетового излучения. Датчик освещенности.

Практика. Программирование макета «Умная метеостанция».

Форма подведения итогов: публичная защита решения кейса.

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ КУРСА «УМНЫЙ ДОМ»

п/п	№ Название раздела, темы	Формы учебного занятия	Формы, методы, приемы обучения. Педагогические технологии	Материально-техническое оснащение, дидактико- методический материал	Формы контроля/ аттестации
	1 Тема 1. Контроллеры и датчики IoT.	Комбинированная.	Объяснительно- иллюстративный. Частично-поисковый.	Проекционное оборудование. Персональный компьютер. Доступ к сети интернет. ПО LibreOffice, Adobe Reader, Arduino IDE. Контроллеры и датчики.	тестирование
	2 Тема 2. Основные алгоритмические конструкции.	Комбинированная.	Объяснительно- иллюстративный. Частично-поисковый. Исследовательский.	Проекционное оборудование. Персональный компьютер. Доступ к сети интернет. ПО LibreOffice, Adobe Reader, Arduino IDE. Контроллеры и датчики.	тестирование
	3 Тема 3. Умная теплица.	Комбинированная.	Объяснительно- иллюстративный. Частично-поисковый. Исследовательский.	Проекционное оборудование. Персональный компьютер. Доступ к сети интернет. ПО LibreOffice, Adobe Reader, Arduino IDE. Контроллеры и датчики. Макет умной теплицы. Сборник заданий «Конструируем умные вещи».	защита решения кейса
	4 Тема 4. Умная	Комбинированная.	Объяснительно-	Проекционное оборудование.	публичная

	метеостанция.		иллюстративный. Частично-поисковый. Исследовательский.	Персональный компьютер. Доступ к сети интернет. ПО LibreOffice, Adobe Reader, Arduino IDE. Контроллеры и датчики. Макет умной метеостанции. Сборник заданий «Конструируем умные вещи».	защита решения кейса
--	---------------	--	--	---	----------------------------

СПИСОК ЛИТЕРАТУРЫ

Список литературы, использованной при написании программы

1. Цветкова М. С., Хлобыстова И. Ю. Информатика: учебник. — М.: Академия, 2020. — 352 с.
2. Филимонова Е. В. Информатика и информационные технологии в профессиональной деятельности: учебник. — М.: КноРус, 2021. — 408 с.
3. Галатенко В. А. Основы информационной безопасности: курс лекций: учебное пособие. — 5-е изд., перераб. и доп. — М.: Интернет-Университет Информационных Технологий, 2021. — 264 с.
4. Шаньгин В. Ф. Информационная безопасность компьютерных систем и сетей: учебное пособие. — М.: ИД «ФОРУМ»: ИНФРА-М, 2020. — 416 с.
5. Гребнев Е. Облачные сервисы: взгляд из России. — М.: CNews, 2021. — 312 с.
6. Советов Б. Я., Цехановский В. В. Информационные технологии: учебник. — 7-е изд., перераб. и доп. — М.: Юрайт, 2022. — 327 с.
7. Фаронов В. В. Turbo Pascal: учебное пособие. — СПб.: Питер, 2021. — 464 с.
8. Немнюгин С. А., Персикова О. Л. Изучаем Turbo Pascal. — СПб.: БХВ-Петербург, 2021. — 768 с.
9. Вирт Н. Алгоритмы и структуры данных. Новая версия для Оберона / пер. с англ. под ред. Ф. В. Ткачева. — М.: ДМК Пресс, 2021. — 272 с.
10. Седжвик Р., Уэйн К. Алгоритмы на Java. — 4-е изд. — М.: Вильямс, 2021. — 848 с.
11. Мэтиз Э. Изучаем Python: программирование игр, визуализация данных, веб-приложения. — 3-е изд. — СПб.: Питер, 2024. — 512 с.
12. Лутц М. Изучаем Python. — 5-е изд. — М.: Диалектика-Вильямс, 2020. — 1360 с.
13. Дронов В. А. Python 3: самое необходимое. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2022. — 608 с.
14. Прата С. Язык программирования C++. Лекции и упражнения. — 7-е изд. — М.: Вильямс, 2021. — 1248 с.
15. Страуструп Б. Программирование: принципы и практика использования C++. — 2-е изд. — М.: Вильямс, 2020. — 1328 с.

16. Подбельский В. В. Язык C++. — 7-е изд., перераб. и доп. — М.: Финансы и статистика, 2021. — 600 с.
17. Винницкий Ю. А., Григорьев А. Т. Scratch и Arduino для юных программистов и конструкторов. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2021. — 336 с.
18. Патаракин Е. Д. Программирование в среде Scratch. — М.: Лаборатория знаний, 2020. — 264 с.
19. Бонд Дж. Unity и C#. Геймдев от идеи до реализации. — 2-е изд. — М.: Питер, 2023. — 928 с.
20. Хокинг Ф. Unity в действии. — 3-е изд. — СПб.: Питер, 2021. — 384 с.
21. Хорстманн К. С. Java. Библиотека профессионала. — 11-е изд. — М.: Вильямс, 2020. — 864 с.
22. Харди Б., Филлипс Б. Android. Программирование для профессионалов. — 4-е изд. — СПб.: Питер, 2021. — 848 с.
23. Петин В. А. Arduino: от простого к сложному. — СПб.: БХВ-Петербург, 2021. — 320 с.
24. Блум Дж. Изучаем Arduino: инструменты и методы технического волшебства. — 2-е изд. — СПб.: БХВ-Петербург, 2022. — 384 с.
25. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. — 6-е изд. — СПб.: Питер, 2021. — 768 с.
26. Богоявленская Д. Б. Психология одаренности и творчества. — М.: Когито-Центр, 2020. — 416 с.
27. Савенков А. И. Психология детской одаренности. — 2-е изд., перераб. и доп. — М.: Юрайт, 2021. — 334 с.

Список литературы, рекомендованной обучающимся

1. Мэтиз Э. Изучаем Python: программирование игр, визуализация данных, веб-приложения. — 3-е изд. — СПб.: Питер, 2024. — 512 с.
2. Фаронов В. В. Turbo Pascal: учебное пособие. — СПб.: Питер, 2021. — 464 с.
3. Петин В. А. Arduino: от простого к сложному. — СПб.: БХВ-Петербург, 2021. — 320 с.
4. Винницкий Ю. А., Григорьев А. Т. Scratch и Arduino для юных программистов и конструкторов. — 2-е изд. — СПб.: БХВ-Петербург, 2021. — 336 с.

5. Хокинг Ф. Unity в действии. — 3-е изд. — СПб.: Питер, 2021. — 384 с.
6. Седжвик Р., Уэйн К. Алгоритмы на Java. — 4-е изд. — М.: Вильямс, 2021. — 848 с.

Перечень литературы, рекомендованной родителям

1. Савенков А. И. Психология детской одаренности: учебник для вузов. — 2-е изд., перераб. и доп. — М.: Юрайт, 2021. — 334 с.
2. Щебланова Е. И. Неуспешные одаренные школьники: проблемы и перспективы. — 2-е изд., перераб. — М.: Лаборатория знаний, 2020. — 245 с.
3. Кови С. 7 навыков высокоэффективных людей. Мощные инструменты развития личности. — М.: Альпина Паблишер, 2021. — 396 с.
4. Пинтусевич И. Действуй! 10 заповедей успеха. — М.: Эксмо, 2020. — 256 с.
5. Бабаева Ю. Д., Войскунский А. Е. Психология одаренности детей и подростков. — М.: Академия, 2020. — 336 с.

СПИСОК ЭЛЕКТРОННЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ

1. Онлайн-платформа «Сириус.Курсы» — курсы повышения квалификации и олимпиадная подготовка по информатике, математике, программированию. — Режим доступа: <https://edu.sirius.online>
2. Stepik.org — образовательная платформа с курсами по Python, C++, Java, Scratch, Unity. — Режим доступа: <https://stepik.org>
3. Code.org — международная некоммерческая платформа для обучения основам программирования детей с 4 лет. — Режим доступа: <https://code.org>
4. Arduino Official Documentation — официальная документация, руководства, библиотеки и форум по Arduino. — Режим доступа: <https://docs.arduino.cc>
5. Python.org Documentation — официальная документация языка Python (актуальная версия). — Режим доступа: <https://docs.python.org/3>
6. Онлайн-школа «Алгоритмика» — программы обучения программированию для детей. — Режим доступа: <https://alg.academy>
7. Платформа «Яндекс.Учебник» — интерактивные курсы по информатике и программированию для школьников. — Режим доступа: <https://education.yandex.ru>
8. Unity Learn — официальная платформа обучения работе в среде Unity и языку C#. — Режим доступа: <https://learn.unity.com>
9. КиберЛенинка — научная электронная библиотека открытого доступа (статьи по педагогике, информатике, ИТ). — Режим доступа: <https://cyberleninka.ru>
10. Учи.ру — образовательная онлайн-платформа с курсами по программированию и информатике для школьников. — Режим доступа: <https://uchi.ru>
11. Дистанционный курс «Алгоритмизация и программирование» <http://moodle.stavdeti.ru>.
12. Тесты. Зачем они нам нужны? Их реальная необходимость в нашей жизни [Электронный ресурс]. - 2010. - Режим доступа: <http://dnevnik.bigmir.net/article/966919>.
13. Сайт дистанционной подготовки по информатике *informatics.mccme.ru*
14. Электронная школа программиста *astp.ru*.

15. Сайт для скачивания свободно распространяемого программного обеспечения Free Basic: freebasic.net.
16. Сайт «Федеральный институт педагогических измерений» fipi.ru
17. Сайт «Успешно сдать ЕГЭ по информатике». Подготовка школьников к сдаче ЕГЭ по предмету "Информатика" <http://infoegehelp.ru>
18. Образовательный портал для подготовки к экзаменам Дмитрия Гущина: Решу ЕГЭ; Сдам ГИА <http://reshuege.ru>
19. Образовательный ресурс: преподавание, наука и жизнь; автор Поляков Константин Юрьевич <http://kpolyakov.spb.ru>
20. Открытый банк заданий ЕГЭ. – Режим доступа: <http://www.fipi.ru/content/otkrytyy-bank-zadaniy-ege>
21. Открытый банк заданий ОГЭ – Режим доступа: <http://www.fipi.ru/content/otkrytyy-bank-zadaniy-oge>
22. Дуванов А. А., Шумилина Н. Д. Алгоритмы и программы Азбуки Роботландии // Информатика, ИД Первое сентября. – 2014., электронное приложение на CD