

Комитет образования и науки администрации города Новокузнецка
Муниципальное бюджетное учреждение
Дополнительного образования
«Центр детского (юношеского) технического творчества «Меридиан»

РАССМОТРЕНО:
на заседании
методического совета
Протокол № 04
«26» мая 2021 г.

СОГЛАСОВАНО:
на заседании
педагогического совета
Протокол № 03
«04» июня 2021 г.



Дополнительная общеобразовательная общеразвивающая программа

«Основы программирования на языке С#»

технической направленности базового уровня

Возраст учащихся: 14 -17 лет

Срок реализации: 4 месяца (72 часа)

Разработчик: Комаров А. Г.,
педагог дополнительного образования

Новокузнецкий городской округ

2021 год

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа **«Основы программирования на языке С#»** относится к программам **технической направленности базового уровня**.

Нормативные документы, на основании которых разработана программа:

- Федеральный закон от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации»;
- Распоряжение Правительства РФ от 4 сентября 2014 г. № 1726-р «Концепция развития дополнительного образования детей»;
- Приказ Министерства просвещения РФ от 9 ноября 2018 г. № 196 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Приказ Министерства просвещения РФ от 30 сентября 2020 г. № 533 «О внесении изменений в Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденный приказом Министерства просвещения Российской Федерации от 9 ноября 2018 г. № 196»;
- Распоряжение Министерства просвещения РФ от 12 января 2021 г. № Р-5 «Об утверждении методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб» в рамках региональных проектов, обеспечивающих достижение целей, показателей и результата федерального проекта «Цифровая образовательная среда» национального проекта «Образование»;
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Приказ Департамента образования и науки Кемеровской области от 05.05.2019 г. № 740 «Об утверждении Правил персонифицированного финансирования дополнительного образования детей»;
- Устав МБУ ДО «Центр детского (юношеского) технического творчества «Меридиан».

В современных реалиях компьютеры и информационные технологии являются частью жизни практически каждого человека. Навыки компьютерной грамотности помогают любому человеку решать повседневные и профессиональные задачи, поэтому программирование переходит из специального знания в универсальную грамотность, развитием которой лучше заниматься с детства. Становится недостаточным владеть навыками работы на компьютере, умение программировать – это неотъемлемая часть современного образования. Чем раньше ученик начнет овладевать навыками программирования, тем больший запас знаний и технологий он получит к моменту выбора основного рода деятельности.

Обучаясь программированию, дети учатся мыслить по-новому. Обычный результат программирования – это проект. Ребенок сразу видит: работает или не работает программа. Если не работает, он старается понять, что нужно узнать для завершения проекта. Ученик мотивирован, ему интересно искать ответы на вопросы, на которые и учитель не знает ответа, но точно может помочь их найти.

Обучать детей программированию нужно не для того, чтобы они в будущем стали программистами. Задача программирования – развить у детей алгоритмическое мышление, которое поможет им уверенно справляться с задачами в самых разных областях науки и техники, которые не имеют однозначного решения. Если ребенок научился раскладывать большую задачу на маленькие фрагменты, находить сходства в разных элементах, выявлять

и устранять незначимые детали, выстраивать фрагменты в единый алгоритм для получения результата, он сможет решить задачи в любой дисциплине. Обучаясь программированию, ребёнок закладывает фундамент будущего успеха в любой профессии, так как алгоритмическое мышление учит формулировать проблему, искать решение и анализировать его, способствует развитию креативного мышления, дети оказываются не пассивными получателями знаний, а наоборот, сами их конструируют. Кроме того, умение разбираться в сложных системах и взаимодействовать с новыми технологиями ему пригодится в любой сфере, ведь цифровые технологии используются повсеместно.

Актуальность программы. Актуальность программы обусловлена возросшей популярностью объектно-ориентированного программирования в сфере разработки ПО. Одним из самых востребованных языков объектно-ориентированного типа является язык C#. Занятия по данной программе способствуют формированию у детей базового представления о принципах ООП, их практическом применении, а также расширяет знания о синтаксисе и возможностях языка C#.

Педагогическая целесообразность программы заключается в том, что настоящий этап развития общества характеризуется интенсивным внедрением во все сферы человеческой деятельности новых наукоёмких технологий. Привлечение детей к информационному творчеству в процессе разработки программного обеспечения (ПО) способствует развитию их способностей и возможностей, ориентированных на успешную адаптацию будущих выпускников к жизни в высокотехнологичном, конкурентном мире. В процессе моделирования и программирования учащиеся получают дополнительные знания в области математики, алгоритмизации, логики, иностранных языков и информатики. Процесс создания приложений предполагает применение теоретических знаний на практике и осознание прикладного характера компетенций, сформированных на занятиях по программированию. Вне зависимости от того, какую профессию учащиеся выберут в будущем, вероятность того, что она может быть связана с информационными технологиями, велика.

Цель программы: формирование компетенций в области программирования на языке высокого уровня C#, включающей в себя методы проектирования, анализа и создания программных продуктов, основанные на использовании объектно-ориентированной методологии для создания условий личностного развития учащихся и их раннего профессионального определения.

Задачи:

образовательные

- сформировать систему профессиональных понятий в областях информатики, математики, алгоритмизации в процессе создания информационного продукта;
- обучить методам программирования на языке C#, являющемся современным средством разработок программного обеспечения (ПО), а также работе в интегрированной среде разработки Visual Studio 2017;
- формировать базовые понятия ООП для их применения при создании приложений;
- обучить принципам объектно-ориентированного подхода к составлению алгоритмов и программ;

развивающие

- развивать интерес учащихся к программированию;
- способствовать развитию алгоритмического и логического мышления;
- развивать умение применять языки (естественные и формальные) и иные виды знаковых систем, технических средств коммуникаций в процессе передачи информации от одного субъекта общения к другому;

воспитательные

- содействовать профессиональному самоопределению учащихся;
- воспитывать навыки самоорганизации, самостоятельной и командной работы;

- воспитать мотивацию учащихся к творчеству, созданию собственных проектов, ответственности за результат своей работы на компьютере и за возможные ошибки;
- воспитывать социально-значимые качества личности человека: ответственность, коммуникабельность, добросовестность, взаимопомощь, доброжелательность, критичность и самокритичность мышления.

Обучение по данной программе основано на следующих **принципах**: научности, сознательности, доступности, наглядности, последовательности, связи теории с практикой, вариативности.

Отличительная особенность программы. Программа предназначена для учащихся, проявляющих интерес к программированию и направлена на формирование навыков проектной деятельности, ориентирует школьников на создание конкретного результата. Результатом практико-ориентированного обучения становится готовый информационный программный продукт (проект, программа, приложение), созданный учащимися по каждому из разделов. Сформированное представление о методах построения алгоритмов программ, знание методологии структурного программирования и ООП, владение навыками программирования на языке C#, а также навыками построения пользовательского графического интерфейса – всё это является основой, которая создаёт условия для дальнейшего самостоятельного творческого развития учащихся в области программирования и их раннего профессионального самоопределения.

Программа организована по принципу «от простого к сложному» и условно может быть разделена на 2 части, имеющих логическую завершенность. Вначале обучение направлено на освоение основ алгоритмизации, что позволяет обеспечить начальную подготовку детей в области программирования и формирует положительную мотивацию к языкам программирования. Другая часть программы («Основы объектно-ориентированного программирования») предполагает освоение методологии структурирования данных, которая является базой в языке объектно-ориентированного типа C#.

Адресат программы. Программа предназначена для учащихся 14-17 лет, которые имеют минимальный необходимый уровень входных компетенций: уверенный пользователь ПК, проявляют интерес к программированию и информационным технологиям в целом. Количество детей в группе от 7 до 12 человек. Реализация программы допускает разновозрастной состав учащихся, что способствует социальному развитию детей, формированию умения работать в разновозрастном коллективе.

Набор учащихся в объединение осуществляется на добровольной основе в течение всего учебного года по факту комплектования групп. Зачисление в группы производится на основании заполнения родителями (законными представителями) заявления о зачислении в учебное объединение.

Объем и срок освоения программы. Программа «Основы программирования на языке C#» рассчитана на 72 часа, реализуется в течение 4-х месяцев на базе ИТ-Куба в учебном кабинете с необходимым оборудованием, техническим и ресурсным обеспечением в соответствии с перечнем, указанным в методических рекомендациях по созданию и функционированию центров цифрового образования «ИТ-Куб» в рамках федерального проекта «Цифровая образовательная среда» национального проекта «Образование».

Занятия проводятся 4 часа в неделю: 2 раза в неделю по 2 академических часа.

Форма обучения – очная. Особенностью организации образовательной деятельности является возможность проведения занятий с применением электронного обучения и дистанционных образовательных технологий, что обеспечивает освоение учащимися образовательной программы в полном объеме независимо от места их нахождения. При проведении занятий с применением электронного обучения и дистанционных образовательных технологий используются официальный сайт МБУ ДО «Центр «Меридиан», платформы для дистанционного онлайн обучения, социальные сети.

Формы и методы работы: фронтальные, групповые, индивидуальные: беседа, объяснение, практические работы, инструктаж, самостоятельная работа (индивидуально и в малых группах), участие в профильных мероприятиях и соревнованиях, демонстрация наглядного материала, частично-поисковый (эвристический) метод, исследовательский метод, метод проектов; метод проблемного изложения; устный опрос, публичное выступление и др.

Виды занятий определяются в зависимости от целей занятия и его темы:

вводное занятие - педагог знакомит учащихся с техникой безопасности, особенностями организации деятельности и предлагаемым планом работы на текущий год;

ознакомительное занятие - педагог знакомит учащихся с новой темой и методами работы в зависимости от темы занятия;

занятие на конструирование и программирование по образцу – занятие, предоставляющее возможность изучать азы программирования по образцу, схеме;

тематическое занятие – участникам предлагается работать над проектом по определённой теме;

занятие-проект – на таком занятии учащиеся получают полную свободу в выборе направления работы, не ограниченного определённой тематикой, рассказывают о выполненной работе, о ходе выполнения задания и назначении выполненного проекта;

конкурсное игровое занятие строится в виде соревнования для повышения активности учащихся и их коммуникации между собой;

комбинированное занятие проводится для решения нескольких учебных задач;

итоговое занятие служит для подведения итогов работы за учебный год, может проходить в виде защиты проектов.

Содержание занятий дифференцировано, с учетом возрастных и индивидуальных особенностей детей и подростков. В программе предусмотрены условия для индивидуального творчества, а также для раннего личностного и профессионального самоопределения детей, их самореализации и саморазвития. Приведенный в программе перечень практических занятий является примерным и может быть изменен педагогом в зависимости от желаний, интересов учащихся. Теоретические и практические занятия проводятся с использованием наглядного материала (технологические карты, разработки занятий, тестирование, алгоритм выполнения задания, видеоуроки).

Реализация программы предполагает использование здоровьесберегающих технологий. Здоровьесберегающая деятельность реализуется:

- через создание безопасных материально-технических условий;
- включением в занятие динамических пауз, периодической смены деятельности учащихся;
- контролем соблюдения учащимися правил работы на ПК;
- через создание благоприятного психологического климата в учебной группе в целом.

Планируемые результаты

Предметные и предпрофессиональные результаты (hard компетенции)

В результате освоения программы учащиеся

будут знать:

- назначение используемых информационных технологий;
- назначение и основные возможности электронных вычислительных машин;
- основные свойства алгоритма, типы алгоритмических конструкций: следование, ветвление, цикл; основы объектно-ориентированного программирования (ООП): понятие метода, класса и объекта;
- особенности работы с интегрированной средой разработки Visual Studio 2017 для программирования на языке C#;
- основные конструкции языка, способы организации процедур и функций в C#;

- виды компьютерного моделирования;
- правила проектирования пользовательского интерфейса;
- методы реализации пользовательского графического интерфейса посредством использования редактора Windows Form;
- принципы тайм-менеджмента;
- этапы разработки проектов;
- правила презентации и продвижения проектного продукта;
- правила создания и представления мультимедийной презентации;
- технику безопасности при нахождении в IT-Кубе, работе со специальным оборудованием при выполнении практико-ориентированных заданий;
- правила безопасной работы на компьютере;

будут уметь:

- организовывать рабочее место;
- соблюдать технику безопасности, технологически правильно обращаться с оборудованием IT-Куба и инструментами при выполнении практико-ориентированных работ, следовать требованиям гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий;
- создавать простые модели объектов и процессов;
- проводить компьютерные эксперименты с использованием готовых моделей объектов и процессов;
- эффективно использовать интегрированную среду разработки Visual Studio;
- разрабатывать ПО на основе использования языка программирования C# и среды разработки Visual Studio;
- эффективно использовать язык C#;
- использовать информационные ресурсы на русском и на иностранных языках.
- создавать и использовать различные формы представления информации: формулы, графики, диаграммы, таблицы (в том числе динамические, электронные, в частности – в практических задачах), переходить от одного представления данных к другому;
- искать информацию с применением правил поиска в компьютерных сетях, не компьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным темам;
- следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий.

Личностные и метапредметные результаты (soft компетенции)

Личностные

- умение использовать приобретенные знания и умения в практической деятельности и повседневной жизни для:
 - создания простых моделей объектов и процессов в виде программных конструкций;
 - создания информационных объектов, в том числе для оформления результатов учебной работы;
 - организации индивидуального информационного пространства, создания личных коллекций информационных объектов;
 - передачи информации по телекоммуникационным каналам в учебной и личной переписке, использования информационных ресурсов общества с соблюдением соответствующих правовых и этических норм;

- знание техники ведения проектной работы, владение основными универсальными умениями информационного характера (постановка и формулирование проблемы, поиск и выделение необходимой информации, выбор наиболее оптимальных способов решения задач в зависимости от конкретных условий);
- постановка цели собственного развития, соотносить собственные возможности и поставленные задачи, определять способы действий в рамках предложенных условий, осуществлять контроль своей деятельности, объективно оценивать результаты своей работы, соотносить свои действия с планируемыми результатами;
- навыки самопрезентации.

Метапредметные регулятивные

- умение осуществлять целеполагание, планирование, корректировку плана, прогнозирование, контроль, коррекцию, оценку деятельности;
- искать информацию с применением правил поиска в компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным темам;

познавательные

- умение ставить учебную задачу, выбрать способы и найти информацию для её решения;
- умение работать с информацией, структурировать полученные знания;
- умение анализировать и синтезировать новые знания, устанавливать причинно-следственные связи, доказывать свои рассуждения;
- умение сформулировать проблему и найти способы её решения;

коммуникативные

- командные компетенции и умение работать в команде;
- умение слушать и слышать собеседника, аргументировать свою точку зрения;
- умение осуществлять инициативное сотрудничество в поиске и сборе информации;
- навыки публичного выступления и презентации результатов.

Формы контроля и подведения итогов реализации программы

На занятиях используются: входной и текущий контроль, промежуточная и итоговая аттестация.

Входной контроль осуществляется через наблюдение за деятельностью учащихся, предполагает собеседование с учащимися, в ходе которого определяется наличие у них минимального необходимого уровня входных компетенций: уверенный пользователь ПК, проявляет интерес к программированию, информационным технологиям в целом.

Текущий контроль осуществляется посредством наблюдения за деятельностью учащихся на каждом занятии и фиксации их умений во время работы над практическими заданиями/работами по разделам и над проектами. Отмечается активность участия учащихся в мероприятиях, степень самостоятельности при работе над практическими заданиями, самостоятельный поиск и разработка интересных тем для доклада (или мини-проекта).

Промежуточная и итоговая аттестация предполагает разработку и реализацию проектов, представление и защиту индивидуальных и групповых проектов, публичное выступление с демонстрацией результатов работы, участие в профильных конкурсах и мероприятиях.

Оценочные материалы представлены в Приложении.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Тема	Количество часов			Формы контроля/ аттестации
		всего	теория	практика	
1.	Вводное занятие	2	1	1	Собеседование
2.	Основы создания пользовательского графического интерфейса (ПГИ)	6	3	3	Практические работы, тестирование программы, наблюдение
3.	Введение в структурное программирование	14	7	7	Практические работы, опрос, тестирование программы, наблюдение
4.	Информационные проекты	12	2	10	Практические работы, тестирование программы, наблюдение
5.	Основы ООП	34	12	22	Практические работы, тестирование программы, наблюдение
6.	Аттестация	4	-	4	Защита проектов, презентация
	Всего:	72	25	47	

СОДЕРЖАНИЕ ПРОГРАММЫ

Раздел 1. Вводное занятие – 2 часа.

Общая информация об IT-Кубе, актуальность направления. Представление программы, ожиданий участников, правил работы. Обзор профильных мероприятий, конкурсов и соревнований. Вводный инструктаж по технике безопасности. Правила работы в объединении и организации рабочего места. Языки программирования. Знакомство участников в малых группах.

Практическая работа. Экскурсия по IT-Кубу. Собеседование. Установка и настройка ПО, необходимого для разработки на языке.

Раздел 2. Основы создания пользовательского графического интерфейса (ПГИ) - 6 часов.

Обработчик событий, работа с Windows Form. Проблемная ситуация, ее представление в виде графического ограничения и анализ. Простейшие инструменты работы с IDE и команды вывода посредством информационных методических ресурсов. Основы объектно-ориентированного программирования на примере элемента обработки событий «Button». Возможности Visual Studio 2017 и Windows Form на основе кнопки. Компонент/класс «label» и его параметры, вывод информации на форму. Понятие «переменная». Типы переменных и их свойства. Обработка событий. Программа «Счётчик». Компонент/класс «edit». Алгоритм установки значения переменной внешним пользователем.

Практическая работа. Постановка проблемной ситуации и поиск путей её решения. Поиск информации в свободных источниках. Создание формы Windows Form. Проверка работоспособности формы. Создание и настройка компонента «Button», тестирование его возможностей. Демонстрация компонента «label» и его параметров. Обработка событий. Создание приложения «Счётчик». Установка значения переменной извне. Демонстрация работы элемента «edit». Создание приложения «Отражение значений» (отражение значений введённой переменной в компоненте «label»).

Раздел 3. Введение в структурное программирование – 14 часов.

Тема 3.1. Основные типы построения алгоритмов - 12 часов.

Представление проблемной ситуации в виде математико-алгоритмического, синтаксического и пунктуационного ограничения и её анализ. Структуры и основные типы построения алгоритмов. Ветвление. Следование. Условный оператор «if». Механизм работы и синтаксис условного оператора «if». Условный оператор «if else». Логические операторы «И», «ИЛИ», «НЕ». Условный оператор «Switch – case, break». Механизм работы и синтаксис условного оператора «Switch». Цикл. Компонент/класс «Мето». Строчный массив. Оператор цикла «for». Механизм работы и синтаксис оператора цикла «for». Строчный массив. Демонстрация работы элемента «Мето». Написание программы вывода введённого текста. Сумма и произведение в цикле, факториал, степень факториала. Циклы с предусловием/постусловием «while / do while». Алгоритм Евклида, НОД, НОК. Вложенные конструкции. Простые и сложные числа. Функции и методы (локальные, глобальные) «void, return». Модификаторы. Механизм работы, синтаксис и отличия функции от процедуры.

Практическая работа. Постановка проблемной ситуации и поиск путей её решения. Написание программы определения совершеннолетия или определения количества колёс заданного вида транспорта. Написание программы определения високосного года по номеру введённого года, начиная с 0 г.н.э. Обзор механизма работы и синтаксиса условного оператора «if else», а также применяемых логических операторов. Объяснение и демонстрация механизма работы и синтаксиса условного оператора «Switch». Написание программы определения и вывода месяца по его введённому порядковому номеру. Написание программы вычисления факториала. Разбор алгоритма Евклида для вычисления НОД. Обзор механизма работы и синтаксиса оператора цикла «while / do while». Написание программы вычисления НОД (алгоритм Евклида), НОК. Разбор примеров вложенного цикла и механизма его работы. Написание программы выявления простого или сложного числа. Демонстрация работы модификаторов. Модификация любого ранее созданного приложения или разработка игры НИМ.

Тема 3.2. Игра НИМ – 2 часа.

Принцип игры НИМ.

Практическая работа. Разработка игры НИМ.

Раздел 4. Информационные проекты – 12 часов.

Структура и этапы построения информационного проекта. Правила оформления, презентации и защиты проекта.

Практическая работа. Выбор проекта. Работа по созданию индивидуального проекта. Подготовка презентации и защиты проекта.

Список проектов для выбора учащихся:

1. Проект «Быки и Коровы».

Описание проекта. Игра для 2 человек. Один задаёт определённую комбинацию, другой угадывает. Неопределённое количество попыток (зависит от длины загадываемой комбинации). При угадывании только символа выдаётся корова, если же и позиция – бык. Чем меньше коров и больше быков, тем лучше.

Практическая работа. Реализация проекта в формате приложения Windows Form на языке C#.

2. Проект «Кодовый замок».

Описание проекта. Кодовый замок из 9 чисел с блокировкой кнопки изменения пароля. Ознакомить с механизмом сохранения файлов.

Практическая работа. Реализация проекта в формате приложения Windows Form на языке C#.

3. Проект «Простой калькулятор».

Описание проекта. Калькулятор с двумя строками ввода и одной строкой вывода.

Практическая работа. Реализация проекта в формате приложения Windows Form на языке C#.

4. Проект «Викторина».

Описание проекта. Викторина с 4 вариантами выбора радио-кнопок. Ознакомление с компонентом радио-кнопка.

Практическая работа. Реализация проекта в формате приложения Windows Form на языке C#.

5. Проект «Прыгающая кнопка».

Описание проекта. Случайное перемещение кнопки по форме. Суть игры – поймать кнопку. Ознакомление с системой координат и компонентом «таймер».

Практическая работа. Реализация проекта в формате приложения Windows Form на языке C#.

Раздел 5. Основы объектно-ориентированного программирования (ООП) – 34 часа.

Тема 5.1. Классы и объекты – 6 часов.

Классы и объекты классов. Синтаксис и механизм работы классов и его объектов. Оператор «new» в данной теме.

Практическая работа. Разбор синтаксиса и механизма работы классов и его объектов. Применение оператора «new» в данной теме. Создание программы «Зоопарк», где объекты – это животные.

Тема 5.2. Наследование – 6 часов.

Наследование. Применение данного принципа на классах и суперклассах.

Практическая работа. Демонстрация применения данного принципа на классах и суперклассах. Модификация программы «Зоопарк» с добавлением подвидов животных.

Тема 5.3. Инкапсуляция. Конструктор – 6 часов.

Инкапсуляция. Работа с данным термином на примере «конструктора».

Практическая работа. Демонстрация применения данного принципа на классах и суперклассах. Модификация программы «Зоопарк» с использованием конструктора (добавить к классам описания животных).

Тема 5.4. Полиморфизм – 2 часа.

Полиморфизм (псевдополиморфизм). Применение данного принципа на примере переменных.

Практическая работа. Демонстрация работы на примере переменных. Освоение принципа ООП «полиморфизм» (псевдополиморфизм) на примере программы «Зоопарк».

Тема 5.5. Определители `privet`, `public`, `partial`, `abstract`, `static`, `protected` – 6 часов.

Значение определителей и способы их использования Плюсы и минусы подобных определителей.

Практическая работа. Разбор значения и способов использования определителей. Модификация программы «зоопарк» с использованием определителя `abstract` (без запуска).

Тема 5.6. Абстрактные классы и методы – 6 часов.

Отличия между классами, методами и их абстрактными версиями.

Практическая работа. Демонстрация и дифференцирование отличий между классами, методами и их абстрактными версиями. Редактирование и отладка кода с применением абстрактных классов.

Раздел 6. Аттестация – 4 часа.

Практическая работа. Создание презентации. Подготовка публичной речи. Защита проектов и презентации.

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Количество учебных недель – 18.

Количество учебных дней – 36.

Продолжительность каникул – нет.

Даты начала и окончания учебных периодов – группы формируются в течение всего учебного года по факту их комплектования.

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Раздел программы	Формы занятий	Приёмы и методы организации образовательного процесса	Дидактический материал	Техническое оснащение занятий	Формы подведения итогов
Вводное занятие	Лекция, презентация, инструктаж	Словесно-наглядный, проблемное изложение, поиск ответов на поставленные вопросы	Презентация, инструкции, подборка профильных мероприятий	Оборудование ИТ-Куба	Собеседование
Основы создания пользовательского графического интерфейса	Лекция, групповая, индивидуальная, практическая работа, разработка программы	Словесно-наглядный, поисковый, практический, проблемный	Материалы программы, видео-уроки	Оборудование ИТ-Куба	Практические работы, разработка программы, наблюдение
Введение в структурное программирование.	Лекция, групповая, индивидуальная, практическая работа, разработка программы	Словесно-наглядный, поисковый, практический, проблемный	Материалы программы, видео-уроки	Оборудование ИТ-Куба	Практические работы, разработка программы, наблюдение
Проекты.	Групповая, индивидуальная, практическая работа, разработка программ-проектов	Словесно-наглядный, поисковый, практический, проблемный	Материалы программы, видео-уроки, банк проектов	Оборудование ИТ-Куба	Практические работы, решение, разработка проектов
Основы ООП	Лекция, групповая, индивидуальная, практическая работа, разработка программы	Словесно-наглядный, поисковый, практический, проблемный	Материалы программы, видео-уроки	Оборудование ИТ-Куба	Практические работы, тестирование программы, наблюдение

Аттестация	Презентация, демонстрация, обсуждение	Словесно-наглядный, практический, презентация проектов	Презентации, подборка профильных мероприятий	Оборудование IT-Куба	Презентация, защита проектов
------------	---------------------------------------	--	--	----------------------	------------------------------

КАЛЕНДАРНЫЙ ПЛАН ВОСПИТАТЕЛЬНОЙ РАБОТЫ

№ п/п	Содержание, виды, формы деятельности	Сроки проведения
Модуль «Воспитываем и познаём»		
1.	Викторина «Знатоки терминов»	сентябрь
2.	Конкурс «Графический перфекционист»	октябрь
3.	Конкурс «Кодовая ошибка»	ноябрь
4.	Конкурс «Обновлённый код»	декабрь
5.	Соревнования «Командная разработка»	январь
Модуль «Воспитываем, создавая и сохраняя традиции»		
1.	Участие в мероприятии «НАНОВый год» в рамках событий общероссийской образовательной программы «Школьная лига РОСНАНО»	сентябрь
2.	Участие в мероприятии «Инженерные каникулы» в рамках национального проекта «Образование»	октябрь
3.	Участие в городских открытых киберсоревнованиях	ноябрь
4.	Участие в большой проектной неделе Центра «Меридиан»	декабрь
5.	Участие в мероприятии «Инженерные каникулы» в рамках национального проекта «Образование»	январь
Модуль «Профорентация»		
1.	Беседа «Востребованность специалистов с навыками программирования на языке C#»	октябрь
2.	Просмотр и обсуждение видеуроков «C# от новичка к профессионалу».	
3.	Исследовательская работа «Плюсы и минусы объектно-ориентированного программирования»	ноябрь
4.	Ролевая игра «Один день в студии разработки»	декабрь
Модуль «Воспитываем вместе»		
1.	«Дни открытых дверей» в Центре цифрового образования «IT-CUBE»	сентябрь
2.	Родительское собрание «Краткий обзор курса «Основы программирования на языке C#». Профильные мероприятия»	октябрь
3.	Консультация родителей по теме «Как помочь ребёнку поддерживать интерес к занятиям в системе дополнительного образования. Рекомендации по установке ПО на домашний компьютер»	ноябрь
4.	Родительское собрание «Наши достижения. Защита проектов»	январь
Модуль РДШ		
1.	Знакомство с сайтом РДШ. Обзор мероприятий на новый учебный год	сентябрь
2.	Участие в мероприятиях РДШ по выбору в соответствии с направлением учебного объединения	в течение года

МАТЕРИАЛЬНО – ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Условия реализации программы: учебный кабинет, оснащенный оборудованием (стандарт).

Перечень необходимого оборудования и расходных материалов (количество единиц оборудования и материалов указано из расчета на 12 человек):

- компьютеры и ноутбуки, на которых установлено соответствующее программное обеспечение: на каждого учащегося и преподавателя – 12 шт. или 4 шт. на малую группу (должны быть подключены к единой Wi-Fi сети с доступом в интернет);
- презентационное оборудование – 2 шт.;
- интерактивная панель – 1 шт.

ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ПРОГРАММЫ

СПИСОК ЛИТЕРАТУРЫ ДЛЯ ПЕДАГОГА

1. С# для школьников: Учебное пособие / М. Дрейер. Перевод с англ. под ред. В. Биллига – М.: Интернет - Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2009. – 128 с.: ил., табл. – (Лицей информационных технологий).
2. Разработка программных модулей: методические указания / М-во науки и высш. образования Российской Федерации, Сиб. гос. индустр. ун-т, Каф. прикладных информационных технологий и программирования; сост. Е. В. Маслова. – Новокузнецк: Издательский центр СибГИУ, 2019. – URL: <http://library.sibsiu.ru>. – Текст: электронный.
3. Трофимов, В. В. Основы алгоритмизации и программирования: учебник для СПО / В. В. Трофимов, Т. А. Павловская; под ред. В. В. Трофимова. – М. : Издательство Юрайт, 2018. – 137 с. – URL: www.biblio-online.ru/book/0E995B4F-410F-41BD-BB85-23823DBA2F64.

КАДРОВОЕ ОБЕСПЕЧЕНИЕ

Педагог, реализующий данную дополнительную общеобразовательную общеразвивающую программу, должен иметь высшее профессиональное образование или среднее профессиональное образование по направлению подготовки «Образование и педагогика» или в области, соответствующей преподаваемому предмету, без предъявления требований к стажу работы; либо высшее профессиональное образование или среднее профессиональное образование по направлению деятельности в образовательном учреждении без предъявления требований к стажу работы.

При реализации программы наставнику рекомендовано пройти обучение в Академии Министерства просвещения РФ в рамках национального проекта «Образование».