

Администрация городского округа Тольятти
Департамент образования
Муниципальное бюджетное образовательное учреждение
дополнительного образования
«Гуманитарный центр интеллектуального развития»
городского округа Тольятти

Программа принята к реализации
решением педагогического
совета. Протокол № 4
от «28» июня 2024г.



УТВЕРЖДАЮ.
Директор МБОУ ДО ГЦИР
А.В. Хаирова
« 28 » июня 2024г. Приказ № 58

**Дополнительная общеобразовательная
общеразвивающая программа
«ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ
PYTHON»**

Направленность техническая

Возраст детей – 12-17 лет

Срок реализации – 1 год

Разработчики:

Лихоманенко Николай Иванович,
Тарасова Маргарита Сергеевна,
педагоги дополнительного образования.

Методическое сопровождение:

Клюева Юлия Викторовна, методист
центра цифрового образования «IT-
куб»

Тольятти

2024

Паспорт дополнительной общеобразовательной программы

Название программы	Дополнительная общеобразовательная общеразвивающая программа «Программирование на языке Python»
Краткое название программы	Программирование на Python
Изображение (логотип)	
Место реализации программы (адреса)	МБОУ ДО ГЦИР: 445045, Самарская область, г.Тольятти, ул. Чайкиной, 87. МБУ школа «Образовательный центр «Галактика». 445047, РФ, Самарская обл., г. Тольятти, ул. Тополиная, 5
Разработчик(и) программы	Лихоманенко Николай Иванович, Тарасова Маргарита Сергеевна, педагоги дополнительного образования.
Методическое сопровождение	Клюева Юлия Викторовна, методист центра цифрового образования «IT-куб»
Краткое описание	Дополнительная программа «Программирование на языке Python» реализуется в рамках центра цифрового образования «IT-куб» и предназначена для школьников 12-17 лет, проявляющих интерес к программированию. Выбор языка PYTHON обусловлен востребованностью языка, развитостью его библиотек для современных парадигм программирования и применением его для разработки веб-приложений и машинного обучения. В процессе обучения основам алгоритмизации и программированию на языке Python обучающиеся получают представление о специфике работы программиста и получают навыки самостоятельной работы над проектом. Программа предполагает углубленное изучение алгоритмов и развитие умений использования языка программирования Python в объектно-ориентированной методологии. Большое внимание уделяется практической работе на компьютере, самостоятельному решению задач и написанию кода. Программа предоставляет возможность для обучающихся освоить её содержание на одном из трех уровней: ознакомительном, базовом, продвинутом
Ключевые слова для поиска	программирование на Python, Питон, язык программирования, алгоритмика, язык программирования Python, алгоритм, дополнительное образование в сфере IT, IT- образование, дополнительная программа , обучение программированию на Python

Цели и задачи	Освоение практических умений программирования на языке Python, умений построения алгоритмов, решения простых задач и задач повышенной сложности, использования переменных различных типов и операций с ними
Результаты освоения	Выпускники будут уметь решать задачи по программированию на языке Python различной степени сложности, писать небольшие собственные программы, создавать проекты
Материальная база	Мультимедийное проекционное оборудование, персональный компьютер для каждого обучающегося, ПО Python, IDLE версия 3.9 и выше
Год создания программы. Где, когда и кем утверждена программа	2021 год. Решение педагогического совета МБОУ ДО ГЦИР от 18.06.2021 г. Протокол № 4
Тип программы по функциональному назначению	общеразвивающая
Направленность программы	Техническая
Направление (вид) деятельности	Информационные технологии
Форма обучения по программе	Очная
Используемые образовательные технологии (перечислить кратко)	Информационно-коммуникационные технологии, проектный метод обучения, модульно-рейтинговая технология оценивания
Уровень освоения содержания программы	Разноуровневая программа: ознакомительный уровень, базовый уровень, продвинутый уровень
Охват детей по возрастам	12 – 17 лет
Вид программы по способам организации содержания	Модульная
Срок реализации программы	1 год
Взаимодействие программы с различными учреждениями и профессиональными сообществами	
Финансирование программы	Реализуется в условиях ПФДО и на бюджетной основе в рамках муниципального финансирования. За рамками муниципального финансирования – на платной основе.
Итоги экспертизы программы на соответствие требованиям ПФДО	Итоговое заключение ОМЭС-36 от 28.02.2024. Итоговое заключение ОМЭС №16 от 27.12.2021 г.
Итоги участия программы в конкурсах	

ОГЛАВЛЕНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	4
Введение	4
Актуальность и педагогическая целесообразность программы	4
Новизна, отличительные особенности данной программы от уже существующих образовательных программ	5
Цель и основные задачи программы	6
Педагогические принципы, определяющие теоретические подходы к построению образовательного процесса.....	6
Основные характеристики образовательного процесса	7
Отбор и структурирование содержания, направления и этапы образовательной программы, формы организации образовательного процесса	9
Воспитательная деятельность в рамках программы	11
Планируемые результаты освоения программы	12
Педагогический мониторинг результатов образовательного процесса	14
УЧЕБНЫЙ ПЛАН ПРОГРАММЫ.....	16
СОДЕРЖАНИЕ ПРОГРАММЫ.....	17
Ознакомительный уровень	17
Базовый уровень	22
Продвинутый уровень	27
ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ.....	33
Кадровое обеспечение.....	33
Методическое обеспечение	33
Информационное обеспечение.....	35
Материально-техническое обеспечение программы	36
СПИСОК ЛИТЕРАТУРЫ	37
ПРИЛОЖЕНИЯ	39
Календарный учебный график программы.....	39
Оценочные материалы	40
Методические материалы	45

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Введение

Дополнительная общеобразовательная общеразвивающая программа «Программирование на языке Python» является неотъемлемой частью образовательной программы муниципального бюджетного образовательного учреждения дополнительного образования «Гуманитарный центр интеллектуального развития» городского округа Тольятти и дает возможность каждому ребенку получать дополнительное образование исходя из его интересов, склонностей, способностей и образовательных потребностей, осуществляемых за пределами федеральных государственных образовательных стандартов и федеральных государственных требований.

Программа «Программирование на языке Python» реализуется в рамках центра цифрового образования «IT-куб» и предназначена для школьников 12-17 лет, проявляющих интерес к программированию. В процессе обучения основам алгоритмизации и программированию на языке Python обучающиеся получают представление о специфике работы программиста и получают навыки самостоятельной работы над проектом. Программа предполагает углубленное изучение алгоритмов и развитие умений использования языка программирования Python в объектно-ориентированной методологии. Большое внимание уделяется практической работе на компьютере, самостоятельному решению задач и написанию кода. Выбор языка PYTHON обусловлен востребованностью языка, развитостью его библиотек для современных парадигм программирования и применением его для разработки веб-приложений и машинного обучения.

По своему функциональному назначению программа является *общеразвивающей*, поскольку она обеспечивает удовлетворение потребностей обучающихся в интеллектуальном, технико-технологическом совершенствовании, в организации их свободного времени.

Программа имеет *техническую направленность*, так как занятия по ней формируют элементарную грамотность в области информационных технологий, развивают умение работать с прикладным программным обеспечением и дают знания основ алгоритмизации.

Актуальность и педагогическая целесообразность программы

Актуальность предлагаемой программы заключается в том, что она ориентирована на приоритетные направления социально-экономического и территориального развития Самарской области, определенных в Стратегии социально-экономического развития Самарской области на период до 2030 года (утверждена постановлением Правительства Самарской обл. от 12.07.2017 г. № 441), в которой поставлена задача качественного изменения структуры направленностей дополнительного образования и увеличения кружков и секций технического профиля.

Актуальность программы обусловлена потребностью общества в технически грамотных специалистах и полностью отвечает социальному заказу по подготовке квалифицированных кадров в области программирования, а также высоким интересом подростков к IT-сфере. Важнейшей характеристикой подрастающего поколения является активность в информационном пространстве, интернет-коммуникации. Программирование развивает математическое и алгоритмическое мышление, способность к абстракции и умение логически связывать объекты разной природы в сочетании представить всё, что угодно, из нуля и единиц. Процессы автоматизации труда требуют написания программ. Программа – это результат коллективного мыслительного процесса проектной команды, материализованный на одном из языков программирования. В сочетании с конкретной предметной областью часть решения задач, возможно, перенести на специалистов, не занимающихся профессионально программированием, но имеющих определенную подготовку, способность к алгоритмическому мышлению и простой инструмент для записи алгоритма. Кроме того, данная программа направлена на профессиональную ориентацию для

тех, кто хочет связать свою трудовую деятельность с информационными технологиями и с развитием «цифровой экономики» и «цифрового общества».

Актуальность дополнительной общеобразовательной общеразвивающей программы соотносится с тенденциями развития дополнительного образования и согласно Концепции развития дополнительного образования способствует формированию и развитию имеющихся навыков операционного и логического стиля мышления, исследовательских и творческих качеств. Актуальность программы отвечает современным потребностям детей и родителей в получении необходимых теоретических знаний и навыков в сфере программирования для личностного развития и успешной адаптации к требованиям жизни в современном информационном обществе. Предлагаемая программа «Программирование на языке Python» способствует формированию математического склада ума, развитию аналитического мышления посредством изучения вопросов программирования и алгоритмизации.

Педагогическая целесообразность программы заключается в том, что содержание программы, используемые технологии, формы и методы обучения создают и обеспечивают необходимые условия для личностного развития и творческого труда обучающихся и позволяют удовлетворить индивидуальные потребности обучающихся в интеллектуальном развитии. Методы, применяемые в процессе обучения, такие как проблемное обучение, проектная деятельность, способствуют формированию мотивации учащихся к углубленному изучению программирования как одной из компьютерных наук. У детей формируется познавательный интерес, самостоятельность мышления, стремление к самопознанию. Программа «Программирование на языке Python» является разноуровневой и модульной, что позволяет более вариативно организовать образовательный процесс, оперативно подстраиваясь под интересы и способности обучающихся.

Таким образом, дополнительная программа «Программирование на языке Python» актуальна и педагогически целесообразна: она удовлетворяет потребности школьников в решении актуальных для них задач – освоении актуальных и значимых знаний и умений, развитии интеллектуальных способностей, воспитание творческой личности, способной реализовать свой потенциал в условиях современного общества.

Новизна, отличительные особенности данной программы от уже существующих образовательных программ

Целью создания программы «Программирование на языке Python» является изменение содержания, организационно-педагогических основ и методов обучения, поддерживающих деятельностный подход к организации обучения в центре цифрового образования «IT-куб».

Особенностью программы от уже существующих образовательных программ является:

- разработка с учетом современных тенденций в образовании по принципу модульного освоения материала, что максимально отвечает запросу социума, т.к. она погружает учащихся в процесс программирования на языке Python, который уже многие годы находится среди первых в списке самых востребованных и популярных языков программирования;
- построение содержания обучения ознакомительного уровня обеспечивает достаточно медленный темп прохождения учебных тем, при этом акцент делается на практическую отработку изучаемого материала и решение задач;
- содержание обучения базового уровня охватывает только базовые основы языка Python, основное внимание при этом концентрируется на детальной практической отработке базы;
- содержание продвинутого уровня охватывает сложные структуры и типы данных языка, что больше приближенно к реально работе специалиста, акцент делается на практическую отработку материала, а также на индивидуальное создание проектов;
- формирование и развитие алгоритмического и логического мышления учащихся, которые являются основой для продолжения изучения любого другого языка программирования и смежных дисциплин (разработка игр, мобильная разработка и т.п.).

Новизна программы заключается в отборе такого содержания обучения, которое погружает в изучение востребованного языка на рынке труда программирования Python, что задаёт необходимую базу и в будущем облегчает переход к изучению других языков программирования.

Цель и основные задачи программы

Цель программы - формирование и развитие алгоритмического и логического мышления учащихся 12-17 лет в процессе изучения основ алгоритмизации и программирования на языке Python.

Основные задачи:

Обучающие:

- 1) познакомить обучающихся с возможностями, синтаксисом, технологией языка Python, с принципами и методами структурного, функционального и объектно-ориентированного программирования, с основными структурами данных и типовыми методами обработки этих структур;
- 2) обеспечить освоение основных базовых алгоритмических конструкций и методов программирования для решения прикладных математических и информационных задач;
- 3) формировать умения работы на компьютере с использованием интегрированной среды IDLE Python и умения разработки эффективных алгоритмов и программ на основе изучения языка программирования Python.

Воспитательные:

- 1) воспитывать культуру безопасного труда при работе за компьютером;
- 2) воспитывать целеустремленность и упорство в достижении результата в процессе решении учебных задач;
- 3) формировать самостоятельность и творческий подход к решению задач с использованием средств вычислительной техники;
- 4) расширять кругозор обучающихся в области программирования.

Развивающие:

- 1) развивать познавательный интерес к программированию на Python;
- 2) развивать умение работать с компьютерными программами и дополнительными источниками информации;
- 3) приобщить к проектно-творческой деятельности, развивать умения планирования проекта, умение работать в группе;
- 4) формировать умения поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использование информации при решении задач.

Педагогические принципы, определяющие теоретические подходы к построению образовательного процесса

Реализация программы «Программирование на языке Python» основывается на общедидактических принципах научности, системности, связи теории с практикой, доступности; принципе модульности, продуктивности, проектности, творчества.

В целях раскрытия педагогического и развивающего потенциала учебно-воспитательного процесса по программе акцент в ней делается на следующих принципах.

1. Принцип научности обучения предполагает соответствие содержания образования уровню развития современной науки и техники, опыту, накопленному мировой цивилизацией. Принцип научности требует, чтобы содержание образования, реализуемое как в учебное, так и во вне учебное время, было направлено на ознакомление обучаемых с объективными научными фактами, явлениями, законами, основными теориями и концепциями той или иной отрасли, приближаясь к раскрытию ее современных достижений и перспектив развития. Принцип научности формирует у учащихся понятия через раскрытие причинно-следственных связей явлений, процессов, событий; проникновение в сущность

явлений и событий; раскрытия истории развития культуры, борьбы тенденций; ориентации на междисциплинарные научные связи.

2. Принцип связи обучения с жизнью реализуется через использование на занятиях жизненного опыта учащихся, приобретенных знаний в практической деятельности, раскрытие практической значимости знаний.

3. Принцип последовательности заключается в последовательном усвоении социального опыта человеком в процессе своего развития с учётом возрастных и индивидуальных особенностей. Существует ряд правил для реализации данного принципа:

- поэтапное усвоение теоретического материала — от простого к сложному, от понятного к непонятному, от реальных форм к абстрактным;
- последовательное овладение технологическими приемами и операциями;
- создание в процессе учения затруднения, проблемной ситуации, которое ставит обучающегося в необходимость соотношения нового и предшествующего опыта;
- работа в «зоне ближайшего развития» ребенка, которая характеризуется решением учащимся учебной (технологической, конструкторской) задачи на повышенном уровне усилий, в т. ч. с дифференцированной помощью педагога.

4. Принцип продуктивности деятельности состоит в обязательности получения продукта самостоятельной деятельности, что является одним из важных условий дополнительного образования. Продуктом деятельности могут быть научно-исследовательская работа, произведения художественного, технического, прикладного творчества (картина, скульптура, вышивка, техническая модель), спортивные достижения и т.п. Самореализация сопровождается созданием лично значимого продукта, позволяющего ребенку самоутвердиться в социальной среде, а также состоянием удовлетворенности от результатов деятельности. Принцип продуктивности предполагает не только получение какого-либо продукта деятельности детьми, но и изменение социума под культурным влиянием дополнительного образования.

5. Проектный принцип предполагает последовательную ориентацию всей деятельности педагога на подготовку и выведение ребенка в самостоятельное проектное действие, развертываемый в логике замысел — реализация — рефлексия. В ходе проектирования перед человеком всегда стоит задача представить себе еще не существующее, но то, что он хочет, чтобы появилось в результате его активности. В логике действия данного принципа в программе предусматриваются практико-ориентированные и исследовательские проекты обучающихся.

Основные характеристики образовательного процесса

Возраст детей, участвующих в реализации программы, - 12 – 17 (6-11 класс). Для овладения программированием необходимы знания по математике, умение логически мыслить, поэтому в группы принимаются школьники, начиная с 6 класса, у которых уже сформирована минимальная математическая база, а также минимальное знание английского языка, который также существенно облегчает процесс обучения программированию.

Условия набора детей в объединение.

В группы ознакомительного уровня принимаются учащиеся 6 класса, а также учащиеся 7-11 классов, кто пока не имеет четкого представления о процессе программирования и о том, отвечает ли занятия программированием их интересам. При приеме в группу ознакомительного уровня предварительная диагностика не проводится. Обучающиеся, закончившие обучение на ознакомительном уровне, могут впоследствии пройти программу базового уровня.

Группа базового уровня рекомендуется обучающимся не младше 7 класса с уже сформированным желанием и запросом к изучению программирования. Прием в группу базового уровня осуществляется по результатам предварительной диагностики, которая проводится до начала учебного года (август). Диагностическая работа включает в себя задачи на логическое мышление, на работу с процентами и на работу с математическими операторами для решения задач.

Обучающиеся, ранее проходившие обучение в центре цифрового обучения «It-куб и освоившиеся дополнительные программы «Лаборатория компьютерных игр», «Программирование это интересно: С++», «Инженерная компьютерная графика», «Программирование роботов», «Программирование на Java», «Мобильная разработка», «Инженериум», «системное администрирование» на уровне не менее 80%, принимаются для обучения по программе базового уровня без предварительной диагностики и имеют преимущество при зачислении.

Группа продвинутого уровня рекомендуется учащимся не младше 8 класса с уже имеющейся базой программирования, знакомых с основными конструкциями языка и способными решать базовые задачи программирования. Прием в группу продвинутого уровня осуществляется по результатам предварительной диагностики, которая проводится до начала учебного года (август). Диагностическая работа включает в себя задачи на использование ввода-вывода в программе, использование базовых конструкций и функций.

Объявление результатов зачисления в группы базового и продвинутого уровня осуществляется путем размещения пофамильного списка-рейтинга с указанием оценок, полученных каждым поступающим.

При наличии свободных мест возможен дополнительный приём обучающихся в течение учебного года. Поступающий зачисляется в группу базового или продвинутого уровня, если на предварительной диагностике он показал результат не менее 80%.

Поступающие, не зачисленные на программу по результатам предварительной диагностики, приглашаются на другие программы центра цифрового образования «IT-куб» в соответствии со своим возрастом.

Характеристика учебных групп по возрастному принципу. Группы формируются с учетом возраста и результатов предварительной диагностики. В группу ознакомительного уровня приглашаются учащиеся 12-13 лет (6 класс), в группу базового уровня - учащиеся 13-15 лет (7-8 классов), в группы продвинутого уровня – учащиеся 15- 17 лет (9-11 класс). Но группы могут быть и разновозрастными. Для обучающихся, разных по возрасту, предусматривается дифференцированный подход при определении индивидуального образовательного маршрута и назначении учебных заданий в процессе обучения.

Категория детей, для которых предназначена программа: любой ребенок, проявляющий интерес к компьютерным технологиям, в том числе одаренные дети и дети с ограниченными возможностями здоровья.

Форма обучения очная.

Срок реализации программы – 1 год.

Количество обучающихся в группе – 10 - 12 человек.

Уровень освоения содержания. Программа является разноуровневой и предоставляет обучающимся возможность освоения учебного содержания с учетом их уровней общего развития, способностей, мотивации. Разный уровень программы реализуется с помощью разного уровня задач и тем проектов. Программа включает в себя три уровня освоения:

- ознакомительный уровень предполагает использование и реализацию общедоступных и универсальных форм организации материала и минимальную сложность предлагаемого для освоения содержания программы (знакомство с тремя основными алгоритмическими конструкциями);
- базовый уровень освоения программы предполагает освоение обучающимися специализированных знаний, обеспечение трансляции общей и целостной картины тематического содержания программы;
- продвинутый уровень предполагает углубленное изучение содержания программы и доступ обучающегося к сложным разделам в рамках содержательно-тематического направления программы.

Вид программы по способам организации содержания модульная.

Взаимодействие данной программы с другими программами МБОУ ДО ГЦИР. Объединение «Программирование на языке Python» является одним из комплекса объединений центра цифрового образования детей «IT-Куб». Внутри центра «IT-Куб»

организована собственная воспитательная система (конкурсные мероприятия, соревнования, открытые защиты проектов, воспитательные мероприятия и праздники). Поэтому объединение «Программирование на языке Python» взаимодействует со всеми другими объединениями центра «IT-Куб».

Обучающимся рекомендуется параллельно с изучением программы «Программирование на языке Python» освоить дополнительные учебные модули «Английский для IT-специалиста» и «4К: компетенции современного IT-специалиста», которые реализуются в рамках Центра цифрового образования «IT-куб».

Возможность продолжения обучения по программам близкого вида деятельности. В соответствии с принципами непрерывности и преемственности образования по окончании обучения по программе «Программирование на языке Python» дальнейшее образование ребенка может быть продолжено по дополнительным программам Центра цифрового образования «IT-Куб», подходящим ему по возрасту. Обучающиеся, освоившие программу «Программирование на языке Python» на уровне не менее 80%, имеют преимущество при зачислении на все программы IT-куба.

Взаимодействие с другими учреждениями. Поскольку центр «IT-Куб» - это сетевой центр, объединение «Программирование на Python» может взаимодействовать с другими центрами «IT-Куб» в других регионах в виде участия в конкурсах, соревнованиях, хакатонах. Также центр напрямую сотрудничает с Детским технопарком «Кванториум 63 регион».

Режим занятий по программе зависит от выбора уровня усвоения программы:

- на ознакомительном и базовом уровне один раз в неделю по 2 учебных часа;
- на продвинутом уровне два раза в неделю по 2 учебных часа.

В соответствии с СП 2.4.3648-20 длительность одного учебного часа для детей среднего и старшего школьного возраста – 40 мин.

Продолжительность образовательного процесса: 36 учебных недель. Начало занятий 15 сентября, завершение 31 мая. *Календарный учебный график см. в Приложении 1.*

Объем учебных часов по программе ознакомительного и базового уровня – 72 часа, по программе продвинутого уровня – 144 часа.

Отбор и структурирование содержания, направления и этапы образовательной программы, формы организации образовательного процесса

Программное содержание, методы, формы, средства обучения отбирались с учетом выше обозначенных принципов и основных направлений развития дополнительного образования, отраженных в Концепции развития дополнительного образования детей до 2030 года (распоряжение Правительства Российской Федерации от 31 марта 2022 г. № 678-р).

Содержание программы направлено на создание условий для вовлечения детей в создание искусственно-технических и виртуальных объектов, построенных по законам природы, в приобретение навыков в области освоения языков программирования, машинного обучения, автоматизации и робототехники, в формирование у обучающихся современных знаний, умений и навыков в области технических наук, технологической грамотности и инженерного мышления.

Содержание программы структурировано следующим образом. Программа реализуется в течение одного учебного года, и может быть реализована на ознакомительном, базовом или продвинутом уровне.

Осваивая программу **на ознакомительном уровне**, обучающиеся получают первичные умения алгоритмизации, работы в интегрированной среде разработки на языке Python, изучают основные конструкции языка программирования.

Содержание программы **на ознакомительном уровне представлено** 4 модулями:

- 1-модуль «Введение в программирование»;
- 2-й модуль «Базовые конструкции в Python»;
- 3-й модуль «Знакомство с графикой в Python»;
- 4-й модуль «Разработка проекта».

Осваивая программу *на базовом уровне*, обучающиеся получают углубленные навыки в алгоритмизации, изучают продвинутую интегрированную среду разработки на языке Python; повторяют основные конструкции языка программирования, изучают новые типы данных в рамках функционального программирования.

Содержание программы на базовом уровне представлено 4 модулями:

- 1-модуль «Введение в программирование»;
- 2-й модуль «Базовые конструкции в Python»;
- 3-й модуль «Процедурное программирование»;
- 4-й модуль «Разработка проекта».

Осваивая программу *на продвинутом уровне*, обучающиеся нацеливаются на углубление и структурирование знаний основ языка программирования Python, формируют умение на практике использовать сложные структуры данных; работать со специальными средствами и библиотеками языка Python, а также осваивают объектно-ориентированное программирование.

Содержание программы на базовом уровне представлено 5 модулями:

- 1-модуль «Структурное программирование. Целочисленная арифметика. Линейные алгоритмы, алгоритмы с ветвлением, повторяемые алгоритмы»;
- 2-й модуль «Структуры данных. Решение задач»
- 3-й модуль «Объектно-ориентированное программирование. Проектирование классов. Иерархия классов Python. Прикладные задачи»
- 4-й модуль «Коллективный проект»;
- 5-й модуль «Коллективный проект».

К концу обучения подростки способны будут самостоятельно определять профессиональные задачи и пути решения; писать грамотный, красивый код; находить и обрабатывать ошибки в коде; разрабатывать эффективные алгоритмы и программы на основе изученного языка программирования Python; способны самостоятельно изучать новые технологии.

Методы и формы организации образовательного процесса

Для решения поставленных задач на всех этапах обучения каждая тема модуля начинается с постановки задачи – характеристики предметной области или конкретной программы на языке Python, которую предстоит изучить. С этой целью педагог проводит демонстрацию презентации или показывает саму программу, а также готовые работы, выполненные в ней. Закрепление знаний проводится с помощью практики отработки умений самостоятельно решать поставленные задачи, соответствующие минимальному уровню планируемых результатов обучения. Основные задания являются обязательными для выполнения всеми обучающимися группы. Задания выполняются на компьютере с использованием интегрированной среды разработки. При этом обучающиеся не только формируют новые теоретические и практические знания, но и приобретают новые технологические навыки.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый обучающийся получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы на домашнем компьютере. Для самостоятельной работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться. В ходе обучения проводится промежуточное тестирование по темам для определения уровня знаний обучающихся. Выполнение контрольных заданий способствует активизации учебно-познавательной деятельности и ведёт к закреплению знаний, а также служит индикатором успешности образовательного процесса. По типу организации взаимодействия педагогов с обучающимися при реализации программы используются личностно-ориентированные технологии, технологии сотрудничества.

Изучение содержания программы осуществляется в разнообразных **формах**:

- групповых: коллективные игры по решению задач, обсуждение итогов, практические занятия;

- индивидуальных: разработка и выполнение творческих проектов, подготовка к конкурсным мероприятиям.

Воспитательная деятельность в рамках программы

Обучение и воспитание представляют собой единый процесс целенаправленного формирования личности ребёнка. Образование по программе не сводится исключительно к передаче знаний, оно выполняет и такие функции, как формирование ряда новых личностных качеств, новых ценностных ориентаций, гибкого мышления, установок на диалог и сотрудничество.

Воспитательные задачи программы (см. подраздел «Цели и основные задачи») реализуются в процессе педагогического общения, в использовании активных методов обучения, побуждающих детей проявить себя в совместной деятельности, принять оценочное решение. Проектная деятельность, другие используемые технологии способствуют приобретению навыков работы в команде, расширению кругозора обучающихся в области программирования, формированию культуры безопасного труда при работе за компьютером, целеустремленности и упорству в достижении результата в процессе решения учебных задач, самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники. Важное актуальное значение имеет и самостоятельная работа, вырабатывающая способность принимать решение и навыки самоконтроля.

В течение всего обучения планируется участие детей в досуговых, социально-значимых и творческих мероприятиях.

Примерный план воспитательных, досуговых мероприятий в объединении

<i>№</i>	<i>Название мероприятия</i>	<i>Примерные сроки</i>	<i>Цели проведения мероприятия</i>
1.	День открытых дверей в центре цифрового образования «It-куб». Презентация объединения	Сентябрь	Привлечение в объединение новых учащихся. Формирование мотивации к творческой деятельности
2.	Командная игра «Безопасная цифра в сети»	Ноябрь (каникулы)	Формирование навыков кибербезопасности
3.	Мероприятие в рамках реализации Федерального календарного плана воспитательной работы «День народного единства»: тематическое занятие	4 ноября	Патриотическое воспитание.
4.	Новогодний праздник «It-ёлка» в центре цифрового образования «It-куб»	Январь	Организация досуга
5.	Мероприятие в рамках реализации Федерального календарного плана воспитательной работы «День защитников Отечества»: тематическое занятие	23 февраля	Патриотическое воспитание
6.	Мероприятие в рамках реализации Федерального календарного плана воспитательной работы «День космонавтики»: тематическое занятие	12 апреля	Воспитание патриотизма, чувства гордости за достижения нашей страны
7.	Мероприятие в рамках реализации Федерального календарного плана воспитательной работы «День Победы»: тематическое занятие	9 мая	Воспитание патриотизма, чувства гордости за подвиг народа в Великой Отечественной войне

8.	Праздник окончания учебного года	Май	Подведение итогов года. Формирование сплоченного детского коллектива
9.	Участие в итоговом мероприятии МБОУ ДО ГЦИР Фестивале интеллекта и творчества «Мы в Центре»	май	Презентация достижений объединения. Формирование сплоченного детского коллектива

Программа предполагает, что обучающиеся представляют результаты своей индивидуальной или групповой работы на конкурсные и неконкурсные мероприятия различного уровня.

***Перечень мероприятий,
в которых могут принять участие обучающиеся по программе***

- 1) Хакатон по программированию «IT-идея» (python) учащихся 8-11 классов (Февраль -март);
- 2) Всероссийский диктант по информационным технологиям «IT-диктант» учащихся 6-11 классов (13 сентября);
- 3) Всероссийский Технологический диктант учащихся 6-11 классов (Ноябрь – декабрь) .

Взаимодействие педагога с родителями

Воспитательная деятельность по программе осуществляется во взаимодействии с семьей. Работа с родителями на протяжении учебного года включает в себя:

№	Вид работы	Цели проведения данных видов работ
1.	Индивидуальные и коллективные консультации для родителей	Совместное решение задач по воспитанию и развитию детей
2.	Организация общения в чате мессенджеров Viber, WhatsApp	Решение организационных вопросов. Реализация единых требований к ребёнку семьи и объединения дополнительного образования
3.	Работа группы «IT-куб Тольятти» в социальной сети «ВКонтакте»: https://vk.com/itcube_tlt	Демонстрация детских достижений, информирование о текущих мероприятиях в объединении, в центре цифрового объединения «IT-куб»
4.	Привлечение родителей к посильному участию в жизни детского коллектива (помощь в обеспечении контроля при выполнении самостоятельных заданий дома, приглашение на открытые занятия и воспитательные мероприятия)	Совместное решение задач по воспитанию, развитию детей и организации образовательного процесса
5	Анкетирование «Удовлетворённость результатами посещения ребёнком занятий объединения»	Изучение потребностей родителей, степени их удовлетворения результатами УВП

Планируемые результаты освоения программы

Требования к уровню подготовки выпускников направлены на овладение обучающимися знаниями и умениями, востребованными в повседневной жизни, значимыми для социальной адаптации личности, ее приобщения к современным It-технологиям.

1. Предметные результаты

Завершив обучение по программе **на ознакомительном уровне**, обучающийся **будет знать**:

- понятийный аппарат программирования: «блок-схема», «переменная», «информация», «алгоритм», «система команд», «программа», «среда разработки», «типы данных»;

- простейшие алгоритмические структуры—линейные, условные и циклические;
- основные понятия и этапы проектной деятельности;

будет уметь:

- грамотно составлять алгоритмы, рассчитанные для конкретного исполнителя;
- пошагово выполнять алгоритмы управления исполнителями;
- анализировать числовые и текстовые данные;
- осуществлять операции как вручную, так и с использованием компьютера.

Дополнительно **на базовом уровне**, обучающийся

будет знать:

- понятийный аппарат программирования: «интерпретатор», «компилятор», «функция», «процедура», «файловая структура проекта», «библиотека»;
- алгоритмы работы вложенных конструкций;
- парадигму процедурного программирования;

будет уметь:

- составлять модульные программы;
- выстраивать файловую структуру проекта;
- устанавливать библиотеки в проект;

Дополнительно **на продвинутом уровне**, обучающийся

будет знать:

- понятийный аппарат программирования: «объект», «инкапсуляция», «наследование», «полиморфизм», «рекурсия», «генератор», «множества»;
- парадигму объектно-ориентированного программирования;

будет уметь:

- интерпретировать и анализировать полученные результаты на предмет их соответствия или несоответствия реальному объекту или процессу;
- выполнять созданные программы, осуществлять их разработку, тестирование и отладку, используя язык программирования Python;
- использовать базовые управляющие конструкции объектно-ориентированного программирования;
- определять результат выполнения алгоритма при заданных исходных данных;
- объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- искать и обрабатывать ошибки в коде;
- разбивать решение задачи на подзадачи, строить абстракцию реальных объектов;
- писать грамотный, красивый код, анализировать как свой, так и чужой код;
- работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения проектных задач.

Более конкретные диагностические признаки по овладению предметными знаниями и умениями приведены в программах каждого из модулей.

2. Метапредметные результаты

Завершив обучение по программе **на ознакомительном и базовом уровне**, обучающийся будет:

- самостоятельно планировать последовательность своих действий для достижения поставленных целей;
- распределять свое время и ресурсы для получения максимально эффективного результата;
- работать в паре и в группе, выстраивая совместную деятельность как с педагогом, так и со сверстниками;
- осуществлять самостоятельный поиск информации, анализировать и обобщать её;
- формулировать, аргументировать и отстаивать своё мнение.

Завершив обучение по программе **на продвинутом уровне**, обучающийся будет:

- формулировать, аргументировать и отстаивать своё мнение;
- критически оценивать правильность решения учебно-исследовательской задачи;
- корректировать свои действия, планировать последовательность шагов для достижения

целей;

- вносить необходимые изменения в процессе работы над программой, исходя из изменяющихся условий;
- выбирать наиболее эффективные способы решения задач на компьютере в зависимости от конкретных условий;
- владеть приемами проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, результат деятельности соотносить с целью, делать выводы и заключения, доказывать, защищать свои идеи, оценивать результаты своей работы.

3. Личностные результаты

Завершив обучение по программе *на ознакомительном и базовом уровне*, обучающийся будет:

- использовать принципы индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;
- уважительно относиться к другому человеку, его мнению, своему и чужому труду;
- бережно относиться к используемому оборудованию;
- работать в команде сверстников в процессе образовательной, учебно-исследовательской и проектной деятельности.

Завершив обучение по программе *на продвинутом уровне*, обучающийся будет:

- формулировать и обосновывать собственное мнение о возможностях, преимуществах и недостатках предлагаемого программного продукта;
- ответственно относиться к обучению, подбору и комбинированию имеющихся умений программиста для решения учебных задач;
- владеть техническим мышлением, соответствующим современному уровню развития информационных технологий;
- иметь опыт распределения задач между членами группы и совместного принятия решений внутри данной группы;
- иметь опыт участия в социально значимых проектах (опыт создания и оценки проекта, повышении уровня самооценки благодаря реализованным проектам).

Педагогический мониторинг результатов образовательного процесса

Педагогический мониторинг освоения программного содержания складывается из следующих компонентов.

- 1) **Входная диагностика знаний.** В начале учебных занятий педагогом проводится вводный контроль для определения начального уровня знаний учащихся в форме беседы «Что я знаю о программировании?».
- 2) **Оперативный контроль** – это контроль за процессом и результатом обучения и коррекция ошибок и пробелов в знаниях и умениях. Оперативный контроль уровня усвоения материала осуществляется по результатам выполнения обучающимися практических заданий на занятиях. Знания теоретического материала проверяются с помощью тестовых заданий.
- 3) **Промежуточный контроль.** В конце каждого модуля проводится контрольная работа.
- 4) **Итоговый контроль** проводится по завершении года обучения в форме презентации результатов индивидуального проекта.

Результаты педагогического мониторинга образовательных результатов группы заносятся педагогом в электронный журнал критериальных оценок. Для оценивания учащихся используется модульно-рейтинговая технология, в которой учитывается количество решенных задач каждым обучающимся. Итоговая оценка выставляется по сумме баллов за все тесты и зачетные работы по схеме: «Н» (низкий уровень) - менее 40% от общей суммы баллов; «С» (средний уровень) - от 40 до 79% от общей суммы баллов; «В» (высокий уровень) - от 80 до 100% от общей суммы баллов.

Подведение итогов реализации программы.

В соответствии с календарным учебным графиком (*Приложение 1*) в конце учебного года проводится итоговая аттестация обучающихся в форме итоговой контрольной работы (*см. Приложение 2 «Оценочные материалы»*).

Сведения о проведении и результатах итоговой аттестации фиксируются педагогом в электронном журнале АСУ РСО, в котором затем создается отчет об освоении программы каждой группой.

Презентация достижений детей (защита проектов) проводится в конце каждого учебного года на учрежденческом Фестивале интеллекта и творчества «Мы в Центре» в номинации «Ярмарка проектов «Время познания».

УЧЕБНЫЙ ПЛАН ПРОГРАММЫ

Вариант 1. Ознакомительный уровень

№	Название модуля	Количество часов		
		теория	практика	Всего
1	Модуль «Введение в программирование»	5	7	12
2	Модуль «Ввод и вывод информации»	6	10	16
3	Модуль «Базовые конструкции в Python»	10	18	28
4	Модуль «Разработка проекта»	4	12	16
	Итого часов по программе:	25	57	72

Вариант 2. Базовый уровень

№	Название модуля	Количество часов		
		теория	практика	Всего
1	Модуль «Введение в программирование»	4	8	12
2	Модуль «Базовые конструкции в Python»	8	16	24
3	Модуль «Процедурное программирование»	5	11	16
4	Модуль «Разработка проекта»	3	17	20
	Итого часов по программе:	20	52	72

Вариант 3. Продвинутый уровень

№	Название модуля	Количество часов		
		теория	практика	Всего
1	Модуль «Структурное программирование. Целочисленная арифметика. Линейные алгоритмы, алгоритмы с ветвлением, повторяемые алгоритмы»	9	23	32
2	Модуль «Структуры данных. Решение задач»	10	20	30
3	Модуль «Объектно-ориентированное программирование. Проектирование классов. Иерархия классов Python. Прикладные задачи»	12	22	34
4	Модуль «Коллективный проект»	5	19	24
5	Модуль «Индивидуальный проект»	5	19	24
	Итого часов по программе:	41	103	144

СОДЕРЖАНИЕ ПРОГРАММЫ

Ознакомительный уровень

МОДУЛЬ 1 «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»

Цель модуля – знакомство со сферой программирования, основными понятиями и основными операторами.

Задачи модуля:

- 1) познакомить с историей и профессиями в сфере программирования;
- 2) познакомить с понятием «переменная», «исполнитель», «система команд», «алгоритм», «программа»
- 3) познакомить с математическими операторами.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятия программирования «исполнитель», «система команд», «алгоритм», «программа», «переменная»;

- математические операторы и как с ними работать;

будут уметь:

- решать базовые уравнения и задачи.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Знакомство с языком программирования Python	1	1	2
2	Знакомство с основными понятиями	2	2	4
3	Математические операторы	2	4	6
	Итого по модулю:	5	7	12

Содержание учебного модуля

Тема 1. Знакомство с языком программирования Python.

Теория. История создания языка, его плюсы и минусы. Сферы использования и перспективы.

Практика. Знакомство с редактором кода IDEL Python. Пример простой программы. Разбор интерфейса. Подсветка синтаксиса.

Входная диагностика. Тестирование «Что я знаю о программировании?».

Тема 2. Знакомство с основными понятиями.

Теория. Понятия «переменная», «исполнитель», «система команд», «алгоритм», «программа».

Практика. Построение линейных текстовых алгоритмов. Решение задач с использованием переменных. Учебный тест «Базовые понятия программирования».

Тема 3. Математические операторы.

Теория. Математические операторы и их применение на практике.

Практика. Решение задач с использованием математических операторов. Решение логических и математических задач.

Подведение итогов модуля. Контрольная работа по модулю. Решение логических и математических задач.

МОДУЛЬ 2 «ВВОД И ВЫВОД ИНФОРМАЦИИ»

Цель модуля – освоение базовых конструкций в программировании и освоение интегрированной среды разработки IDLE для изучения Python.

Задачи модуля:

- 1) сформировать умение работы с функциями ввода и вывода информации;
- 2) сформировать умение работы с циклами в Python.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятия, относящиеся к построению алгоритмов: «последовательный алгоритм»;
- команды ввода-вывода информации;
- базовые типы данных и их методы.

будут уметь:

- писать программы с использованием функций ввода и вывода информации;
- строить последовательный алгоритм;
- писать программы, используя разные типы данных;
- писать форматированный вывод для вывода результата программы;
- уметь работать с такими типами данных, как строка и список, и использовать их методы.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Функция вывода информации	1	1	2
2	Форматированный вывод информации	2	2	4
3	Функции ввода информации	1	3	4
4	Типы данных: строка, список	2	4	6
Итого по модулю:		6	10	16

Содержание учебного модуля

Тема 1. Изучение среды разработки IDLE .

Теория. Базовый вывод информации.

Практика. Практическая работа с выводом результатов простейших примеров и текстовой информации.

Тема 2. Форматированный вывод информации.

Теория. Встроенная функция вывода и её параметры. Операторы форматированного вывода. Метод format. Строковый метод f-строки. Поля заданной ширины.

Практика. Разбор решения задач на вывод форматированного текста. Самостоятельная практическая работа с выводом форматированного текста.

Тема 3. Функция ввода информации.

Теория. Встроенная функция ввода информации. Типы данных. Блок-схема: последовательность.

Практика. Практическая работа на построение линейных алгоритмов. Практическая работа на ввод и расчет данных определенного типа данных.

Тема 3. Типы данных строка, список.

Теория. Тип данных строка. Методы строк. Тип данных список. Методы списков

Практика. Создание списка, операции со списками и элементами списков, перебор списка. Применение методов списков. Применение методов строк.

Подведение итогов модуля. Контрольная работа по модулю. Решение задач с использованием ввода и вывода информации, работа со строками и списками.

МОДУЛЬ 3 «БАЗОВЫЕ КОНСТРУКЦИИ В PYTHON»

Цель модуля – познакомить с основными алгоритмическими структурами (линейная, ветвление, цикл), основными типами данных и с основами программирования на языке Python.

Задачи модуля:

- 1) сформировать умение в построении алгоритмов ветвления и циклов;
- 2) сформировать умение построения программы с использованием операторов в python, для алгоритмов ветвления и циклов.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятия, относящиеся к построению алгоритмов, «алгоритм ветвление», «циклический алгоритм с параметром», «циклический алгоритм с предусловием»;
- операторы базовых конструкций,

будут уметь:

- строить алгоритмы ветвления и циклов;
- использовать базовые конструкции для построения программного кода;
- будут уметь работать с наборами данных, используя алгоритмы ветвления и циклов.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Алгоритм ветвления. Построение блок-схем	2	2	4
2	Условная конструкция	1	3	4
3	Цикл с параметром. Построение блок-схем	2	2	4
4	Цикл for. Перебор данных в программировании.	1	3	4
5	Цикл с предусловием. Построение блок-схем	2	2	4
6	Цикл while. Цикл с условием.	1	3	4
7	Операторы циклов, условие выхода из цикла	1	3	4
Итого по модулю:		10	18	28

Содержание учебного модуля

Тема 1. Алгоритм ветвления. Построение блок-схем.

Теория. Изучение блоков алгоритма ветвления. Изучение логики построения алгоритма с ветвления.

Практика. Разбор примеров на построение алгоритмов ветвления. Решение задач на построение алгоритмов ветвления.

Тема 2. Условная конструкция.

Теория. Операторы условной конструкции. Построение кода с использованием условной конструкции.

Практика. Разбор примеров на использование условной конструкции в программировании. Решение задач с использованием условной конструкции.

Тема 3. Цикл с параметром. Построение блок-схем.

Теория. Изучение блоков цикла с параметром. Изучение логики построения алгоритма цикл с параметром.

Практика. Разбор примеров на построение алгоритмов цикла с параметром. Решение задач на построение алгоритмов цикла с параметром.

Тема 4. Цикл for. Перебор данных в программировании.

Теория. Изучение операторов цикла for. Построение кода с использованием циклической конструкции с параметром.

Практика. Разбор примеров на использовании циклической конструкции в программировании. Решение задач на построение алгоритмов цикла с параметром.

Тема 5. Цикл с предусловием. Построение блок-схем.

Теория. Изучение блоков цикла с предусловием. Изучение логики построения алгоритма цикла с предусловием.

Практика. Разбор примеров на построение алгоритмов цикла с предусловием. Решение задач на построение алгоритмов цикла с предусловием.

Тема 6. Цикл while. Цикл с условием.

Теория. Изучение операторов конструкции while. Построение кода с использованием конструкции цикла с предусловием.

Практика. Разбор примеров на использование циклической конструкции в программировании. Решение задач на построение алгоритмов цикла с предусловием.

Подведение итогов модуля. Контрольная работа по модулю: решение задач на построение основных конструкций.

МОДУЛЬ 4 «РАЗРАБОТКА ПРОЕКТА»

Цель модуля – формирование умений проектной деятельности.

Задачи модуля:

- 1) познакомить с проектной деятельностью, ее аспектами;
- 2) коллективно разработать проект, с внесением изменений в проект обучающимися;
- 3) сформировать умения публичного выступления.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятие «проект»,
- как использовать изученные инструменты для создания проекта,
- способы визуализации проектной идеи;

будут уметь:

- определять цель и задачи проекта,
- создавать и визуализировать проект,
- создавать презентацию.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Введение в проектную деятельность. Разбор основных аспектов создания проекта	2	2	4
2	Разработка проекта. Определение идеи проекта. Определение цели и задач. Разработка структуры проекта. Разработка программы.	1	7	8
3	Оформление проекта. Подготовка презентации	1	1	2
4	Защита проекта	0	2	2
Итого по модулю:		4	12	16

Содержание учебного модуля

Тема 1. Введение в проектную деятельность. Разбор основных аспектов создания проекта. Работа над проектом.

Теория. Понятие «проект». Цели, задачи. Планирование проекта. Реализация.

Практика. Творческая работа на определение идеи проекта. Визуализация проектной идеи.

Тема 2. Разработка проекта.

Теория. Визуализация проектной идеи (схемы, блок-схемы). Разбор примера проекта.

Практика. Формулирование цели и задач. Разработка структуры проекта (схема). Программирование проекта. Тестирование получившегося продукта.

Тема 2. Оформление проекта. Подготовка презентации.

Теория. Требования к оформлению проекта, презентации.

Практика. Оформление проекта. Подготовка презентации.

Тема 3. Защита проекта.

Теория. Основные требования к докладу проекта.

Практика. Подготовка доклада.

Подведение итогов модуля. Защита проекта.

Подведение итогов учебного года. Итоговая аттестация обучающихся по методу модульно-рейтинговой оценки. Защита индивидуальных проектов на итоговом мероприятии МБОУ ДО ГЦИР Фестивале интеллекта и творчества «Мы в Центре» в номинации «Ярмарка проектных идей «Время познания». Праздник окончания учебного года. Коллективное обсуждение итогов освоения программы и индивидуальное осмысление своей деятельности.

Базовый уровень

МОДУЛЬ 1 «ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ»

Цель модуля – знакомство с интегрированной средой разработки IDLE для изучения Python.

Задачи модуля:

- 1) познакомить с понятием «переменная», «исполнитель», «система команд», «алгоритм», «программа», «среда разработки», «переменная», «ввод-вывод в программе», «условный оператор»;
- 2) познакомить с математическими операторами;
- 3) познакомить с командами ввода и вывода;
- 4) познакомить базовыми типами данных;
- 5) познакомить с интегрированной средой разработки IDLE для изучения Python.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятия программирования «исполнитель», «система команд», «алгоритм», «программа», «среда разработки», «переменная», «условный оператор», «ввод-вывод в программе»;
- математические операторы и как с ними работать;
- типы данных и как с ними работать;
- как работать со средой разработки IDLE.

будут уметь:

- решать базовые и усложненные уравнения и задачи.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Знакомство с языком программирования Python. Знакомство с основными понятиями. Работа с IDLE, примеры и вывод данных	2	2	4
2	Переменные. Математические операторы, операторы сравнения, логические операторы, операторы тождественности.	1	3	4
3	Типы данных. Ввод данных	1	3	4
Итого по модулю:		4	8	12

Содержание учебного модуля

Тема 1. Знакомство с языком программирования Python. Знакомство с основными понятиями. Работа с IDLE, примеры и вывод данных.

Теория. История создания языка, его плюсы и минусы. Сферы использования и перспективы. Понятия кода, интерпретатора, программы. Простейшие программы с выводом на экран.

Практика. Решение примеров и вывод на экран.

Тема 2. Переменные и операторы Python.

Теория. Переменные. Математические операторы, операторы сравнения, логические операторы, операторы тождественности.

Практика. Решение уравнений с применением математических операторов.

Тема 3. Типы данных. Ввод данных.

Теория. Ввод в программе. Типы данных, примеры работы с типами данных. Метод `map`.

Практика. Укрепление навыка работы с переменными на примерах. Выполнение практических заданий.

Подведение итогов модуля. Контрольная работа по модулю: решение задач на использование операторов `python`, ввод и вывод данных.

МОДУЛЬ 2 «БАЗОВЫЕ КОНСТРУКЦИИ В PYTHON»

Цель модуля – формирование умений использования в работе условных операторов, циклов, списков, строк и функций.

Задачи модуля:

- 1) сформировать умение работы с условными операторами в `Python`, с вложенными конструкциями;
- 2) сформировать умение работы с циклами в `Python`;
- 3) сформировать умение работы со списками в `Python`, познакомить с методами списков;
- 4) сформировать умение работы со строками в `Python`, познакомить с методами строк;
- 5) сформировать умение работы со словарями.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятия «условная конструкция», «цикл», «срез», «список», «строка», «функция»;

будут уметь:

- писать программы с использованием условных операторов, циклов, списков, строк и функций.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Условный оператор. Построение алгоритма для задачи	2	4	6
2	Циклы. Построение алгоритма для задачи.	2	4	6
3	Строки. Списки. Методы списков и строк. Срезы	2	4	6
4	Словарь. Методы словарей	2	4	6
Итого по модулю:		8	16	24

Содержание учебного модуля

Тема 1. Условный оператор. Построение алгоритма для задачи.

Теория. Понятие о языке высокого уровня `Python`. Основные управляющие конструкции алгоритмов с ветвлением в `Python`.

Практика. Решение задач с условиями, построение алгоритмов решения условных задач.

Тема 2. Циклы. Построение алгоритма для задачи.

Теория. Устройство циклов `for` и `while`. Основные управляющие конструкции циклического алгоритма в `Python`. Алгоритм Евклида.

Практика. Решение циклических задач. Совмещение конструкции цикла с условной конструкцией для решения задач с циклами и условиями.

Тема 3. Строки. Списки. Методы списков и строк. Срезы.

Теория. Списочные выражения. Методы `split`, `join`. Другие методы списков и строк. Работа со списками, строками, срезами.

Знакомство со срезами и диапазонами.

Практика. Создание списка, операции со списками и элементами списков, перебор списка. Применение методов списков. Применение методов строк. Применение срезов к спискам и строкам.

Тема 4. Функции.

Теория. Знакомство с функциями. Области видимости переменных. Понятие подпрограммы, процедуры, функции. Функции в Python.

Практика. Рассмотрение примера разделения задачи на подзадачи, написание функций для подзадач. Самостоятельное выполнение похожих задач.

Подведение итогов модуля. Контрольная работа: решение задач по темам модуля.

МОДУЛЬ 3 «ПРОЦЕДУРНОЕ ПРОГРАММИРОВАНИЕ»

Цель модуля – сформировать навыки по работе с графическими объектами.

Задачи модуля:

- 1) познакомить с модульностью в программировании;
- 2) познакомить с синтаксисом построения функций;
- 3) познакомить с понятиями «параметр», «аргумент», «возврат значения»;
- 4) сформировать умение создавать пользовательские функции и выстраивать небольшие проекты с вызовом функций.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- модульность в программировании,
- разницу между функциями и процедурами,
- понятия «параметр», «аргумент», «возврат значения», «функция», «процедура»;

будут уметь:

- писать свои пользовательские функции,
- тестировать написанную функцию.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Функции. Абстракция, возможность повторного использования, модульность. Синтаксис функции	1	1	2
2	Область видимости функций	1	3	4
3	Виды аргументов в функции	1	3	4
4	Возвращаемые значения (return). Функции, процедуры	1	3	4
5	Lambda функции. Анонимные функции в программировании	1	1	2
Итого по модулю:		5	11	16

Содержание учебного модуля

Тема 1. Функции. Абстракция, возможность повторного использования, модульность. Синтаксис функции.

Теория. Абстракция, возможность повторного использования, модульность. Параметры и аргументы.

Практика. Написание простейших задач с использованием синтаксиса функции.

Тема 2. Область видимости функций.

Теория. Локальная, область объемлющих функций, глобальная область видимости.

Практика. Решение задач для отработки областей видимости.

Тема 3. Виды аргументов в функции.

Теория. Позиционные. Именованные. Необязательные параметры. Аргументы переменной длины. Передача по значению и по ссылке. Словарь в качестве аргументов.

Практика. Разбор примеров работы с аргументами. Решение задач.

Тема 4. Возвращаемые значения. Функции, процедуры.

Теория. Что можно возвращать? Распаковка возвращаемых значений. Пустая функция. Отличие функции от процедуры.

Практика. Разбор примеров работы с возвращением значений, примеры процедур. Решение задач.

Тема 5. Lambda функции. Анонимные функции в программировании.

Теория. Lambda функции. Для чего используются, как перевести обычную функцию в lambda функцию.

Практика. Решение задач на построение анонимных функций.

Подведение итогов модуля. Контрольная работа по темам модуля: решение задач с построением функции, процедур и анонимных функций.

МОДУЛЬ 4 «РАЗРАБОТКА ПРОЕКТА»

Цель модуля – формирование умений проектной деятельности.

Задачи модуля:

- 1) познакомить с проектной деятельностью, ее аспектами;
- 2) коллективно разработать проект, с внесением изменений в проект обучающимся;
- 3) сформировать умения публичного выступления.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- понятие «проект»,
- как использовать изученные инструменты для создания проекта,
- способы визуализации проектной идеи;

будут уметь:

- определять цель и задачи проекта,
- создавать и визуализировать проект,
- создавать презентацию.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Введение в проектную деятельность. Разбор основных аспектов создания проекта	2	2	4
2	Коллективный проект. Определение идеи проекта. Определение цели и задач. Разработка структуры проекта. Разработка программы	2	4	6
3	Оформление проекта. Разбор примера защиты проекта. Подготовка презентации. Пробные выступления	1	1	2

4	Разработка индивидуального проекта	1	5	6
5	Защита проекта	0	2	2
	Итого по модулю:	6	14	20

Содержание учебного модуля

Тема 1. Введение в проектную деятельность. Разбор основных аспектов создания проекта.

Теория. Понятие «проект». Цели, задачи. Планирование проекта. Реализация.

Практика. Творческая работа на определение идеи проекта. Визуализация проектной идеи.

Тема 2. Коллективный проект. Определение идеи проекта. Определение цели и задач. Разработка структуры проекта. Разработка программы.

Теория. Визуализация проектной идеи (схемы, блок-схемы). Разбор примера проекта.

Практика. Формулирование цели и задач. Разработка структуры проекта (схема). Программирование проекта. Тестирование получившегося продукта.

Тема 3. Оформление проекта. Разбор примера защиты проекта. Подготовка презентации. Пробные выступления.

Теория. Требования к оформлению проекта, презентации. Разбор примера защиты проекта

Практика. Подготовка презентации. Пробные выступления

Тема 4. Разработка индивидуального проекта.

Теория. Специфика разработки индивидуального проекта.

Практика. Формулирование цели и задач. Разработка структуры проекта (схема). Программирование проекта. Тестирование получившегося продукта.

Тема 5. Защита проекта.

Теория. Основные требования к докладу проекта.

Практика. Подготовка доклада.

Подведение итогов модуля. Защита проекта.

Подведение итогов учебного года. Итоговая аттестация обучающихся по методу модульно-рейтинговой оценки. Защита индивидуальных проектов на итоговом мероприятии МБОУ ДО ГЦИР Фестивале интеллекта и творчества «Мы в Центре» в номинации «Ярмарка проектных идей «Время познания». Праздник окончания учебного года. Коллективное обсуждение итогов освоения программы и индивидуальное осмысление своей деятельности.

Продвинутый уровень

МОДУЛЬ 1 «СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ. ЦЕЛОЧИСЛЕННАЯ АРИФМЕТИКА. ЛИНЕЙНЫЕ АЛГОРИТМЫ, АЛГОРИТМЫ С ВЕТВЛЕНИЕМ, ПОВТОРЯЕМЫЕ АЛГОРИТМЫ»

Модуль предназначен для знакомства со структурной парадигмой в программировании и применении этой методологии при решении практических задач. В модуле особо рассматриваются задачи на целочисленную арифметику и на основные конструкции языка.

Цель модуля – знакомство со структурным подходом при написании программ, синтаксисом и стандартными библиотеками.

Задачи модуля:

- 1) познакомить со структурным подходом в программировании;
- 2) сформировать умение решать задачи с целочисленными характеристиками;
- 3) сформировать умение использовать основные конструкции для решения задач;
- 4) познакомить с автоматической системой проверки программ.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- принципы и методы структурного подхода в программировании,
- алгоритмы целочисленного решения задач,

будут уметь:

- пользоваться автоматической системой проверки задачи,
- использовать основные конструкции для записи алгоритмов.
- интерпретировать и анализировать полученные результаты на предмет их соответствия или несоответствия реальному объекту или процессу;
- выполнять созданные программы, осуществлять их разработку, тестирование и отладку, используя язык программирования Python.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	Всего
1	Введение. Реализации Python	1	1	2
2	Структурное программирование. Целочисленная арифметика. Линейные алгоритмы	2	4	6
3	Алгоритмы с ветвлением	2	6	8
4	Словари	2	6	8
5	Множества	2	6	8
Итого по модулю:		9	23	32

Содержание учебного модуля

Тема 1. Введение. Реализации Python.

Теория. История, область применения. Реализации Python. Установки на различные платформы. IPython, IDLE, PyCharm. Байт-код и интерпретация Python и вычисления (Python и математика). Python и программы cGUI.Python, web и xml. Инструктаж по технике безопасности при работе с оборудованием в аудитории и на рабочем месте ученика.

Практика. Примеры простых программ на Python.

Тема 2. Структурное программирование. Целочисленная арифметика. Линейные алгоритмы.

Теория. Синтаксис языка. Целочисленная арифметика.

Практика. Примеры задач на целочисленную арифметику. Решение задач.

Тема 3. Алгоритмы с ветвлением.

Теория. Синтаксис языка: управляющие структуры. Повторяемые алгоритмы

Практика. Примеры задач с управляющими структурами. Решение задач.

Тема 4. Словари.

Теория. Типы данных. Словари. Генератор словарей. Определение Словаря. Понятия ключ, значение. Получение данных из словаря. Добавление и обновление ключей. Удаление ключей. Методы.

Практика. Решение задач на создание и применение словарей.

Тема 5. Множества.

Теория. Типы данных: Множества. Генератор множеств. Определение множества. Создание множеств. Доступ к элементам множеств. Добавление элементов во множество. Удаление элементов из множеств. Объединение множеств. Пересечение множеств. Разница множеств. Сравнение множеств. Методы множеств.

Практика. Решение задач на создание и применение множеств.

Подведение итогов модуля. Контрольная работа по темам модуля. Решение задач по темам повторение конструкций с использованием новых типов данных (словарей и множеств).

Запись в файл.

МОДУЛЬ 2 «СТРУКТУРЫ ДАННЫХ»

Модуль предназначен для знакомства с массивами, списками, таблицами, стеком, очередью и алгоритмами их обработки. Кроме того, рассматривается класс задач для решения, которых используется динамическое программирование и рекурсии.

Цель модуля – знакомство с разнообразными структурами данных и с динамическим подходом решения задач.

Задачи модуля:

- 1) познакомить с простейшими структурами данных;
- 2) сформировать умение использовать в программах алгоритмы поиска данных и алгоритмы упорядочивания данных в структурах;
- 3) познакомить с динамическим программированием.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- метод динамического программирования,
- устройство структур данных,
- алгоритмы поиска и упорядочивания наборов данных;

будут уметь:

- проектировать и использовать структуры данных в контексте задачи,
- применять метод динамического программирования при решении задач.
- определять результат выполнения алгоритма при заданных исходных данных;
- объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- искать и обрабатывать ошибки в коде;
- разбивать решение задачи на подзадачи, строить абстракцию реальных объектов;
- писать грамотный, красивый код, анализировать как свой, так и чужой код.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	Всего
1	Контекстный менеджер	2	4	6
2	Упорядочивание данных	2	4	6
3	Рекурсия, итераторы, генераторы, декораторы методов	2	4	6
4	Поиск данных	2	4	6
5	Динамическое программирование	2	4	6
	Итого по модулю:	10	20	30

Содержание учебного модуля

Тема 1. Контекстный менеджер.

Теория. Определение контекстного менеджера. Разбор эффективных и не эффективных примеров работы с файлами. Знакомство с операторами контекста. Методы для работы с файлами.

Практика. Разбор аффективных и не эффективных примеров работы с файлами на практике. Решение задач на запись в файл, чтение из файла, структурирование данных в файле. .

Тема 2. Упорядочивание данных.

Теория. Методы упорядочивания данных. Виды сортировок.

Практика. Решение задач на упорядочивание данных.

Тема 3. Рекурсия, итераторы, генераторы, декораторы методов.

Теория. Рекурсивные функции. Знакомство с классическими задачами рекурсии. Линейная и каскадная рекурсия. Сравнение итератора с рекурсией. Работа с итераторами. Знакомство с декораторами.

Практика. Решение задач с использованием рекурсивных алгоритмов.

Тема 4. Поиск данных.

Теория. Методы поиска данных. Алгоритм бинарного поиска и остальные виды поиска данных.

Практика. Решение задач на поиск данных.

Тема 5. Динамическое программирование.

Теория. Задача. Подзадача. Метод динамического программирования.

Практика. Решение задач с использованием динамического программирования.

Подведение итогов модуля. Контрольная работа: решение задач по темам модуля.

МОДУЛЬ 3 «ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ. ПРОЕКТИРОВАНИЕ КЛАССОВ. ИЕРАРХИЯ КЛАССОВ PYTHON. ПРИКЛАДНЫЕ ЗАДАЧИ»

Модуль предназначен для знакомства с объектно-ориентированной методологией в программировании и применении этой методологии при решении практических задач. Рассматриваются задачи проектирования классов, создание и взаимодействие объектов, иерархия классов Python.

Цель модуля — знакомство с объектно-ориентированной парадигмой программирования.

Задачи модуля:

- 1) познакомить с объектно-ориентированным подходом в программировании;
- 2) научить проектировать классы;

- 3) научить применять классы для решения задач;
- 4) познакомить с иерархией классов Python.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- принципы и методы объектно-ориентированного программирования,

будут уметь:

- использовать базовые управляющие конструкции объектно-ориентированного программирования;
- проектировать и применять классы для решения практических задач.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Объектно-ориентированное программирование	2	4	6
2	Проектирование классов. Иерархия классов Python. Библиотеки Python	4	6	10
3	Генераторы, декораторы методов и классов. Itertools. Listcomprehensions	6	12	18
Итого по модулю:		12	22	34

Содержание учебного модуля

Тема 1. Объектно-ориентированное программирование.

Теория. Абстракция. Инкапсуляция. Наследование. Полиморфизм. Классы.

Практика. Примеры создания и использования класса.

Тема 2. Проектирование классов. Иерархия классов Python. Библиотеки Python.

Теория. Иерархия классов Python. Магические методы классов.

Практика. Пример проектирования класса. Решение задач.

Тема 3. Генераторы, итераторы, декораторы классов. Itertools. Listcomprehensions.

Теория. Синтаксис языка: Генераторы, декораторы методов и классов. Itertools. ListComprehensions.

Практика. Работа с генераторами, итераторами, декораторами.

Подведение итогов модуля. Самостоятельная работа, решение задач по темам модуля.

МОДУЛЬ 4 «КОЛЛЕКТИВНЫЙ ПРОЕКТ»

Модуль предназначен для формирования приёмов коллективной проектной деятельности, включая умения коллективного обсуждения и формирования темы и цели проекта, составлять план коллективной деятельности, распределять обязанности, оценивать результаты коллективной работы.

Цель модуля – формирование умения работать в команде в ходе создания коллективного проекта.

Задачи модуля

- 1) сформировать умение создавать проект используя объектно-ориентированное программирование;
- 2) формировать умения выбирать наиболее эффективные способы решения задач на компьютере в зависимости от конкретных условий;
- 3) формировать способы и приемы коллективной проектной деятельности и коммуникативные умения, умения обсуждать и договариваться в ходе решения задачи;

- 4) формировать умения составлять план коллективной деятельности, распределять обязанности, оценивать результаты коллективной работы.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- принципы, приемы работы с проектом;
- приемы коллективного обсуждения проблем и формулирования темы и цели проекта,
- этапы создания коллективного практико-ориентированного проекта;

будут уметь:

- проектировать и разрабатывать проект;
- выбирать наиболее эффективные способы решения задач на компьютере в зависимости от конкретных условий;
- формулировать тему и цель проекта, составлять план своей деятельности,
- распределять обязанности внутри команды, распределять время на работу по проекту;
- осуществлять командную презентацию проекта.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Формирование команд и выбор темы проекта	2	2	4
2	Разработка проекта. Коллективный практико-ориентированный проект	2	14	16
3	Презентация проекта	1	3	4
Итого по модулю:		5	19	24

Содержание учебного модуля

Тема 1. Формирование команд и выбор темы проекта.

Теория. Определение направлений для разработки проекта. Разбор примеров.

Практика. Формирование команд. Формулирование темы и цель проекта, составление плана деятельности.

Тема 2. Разработка проекта. Коллективный практико-ориентированный проект.

Теория. Поэтапная разработка проекта.

Практика. Проектирование и реализация проекта.

Тема 3. Презентация проекта.

Теория. Требования к презентации проекта.

Практика. Создание презентации.

Подведение итогов модуля. Презентация выполненных коллективных практико-ориентированных проектов.

МОДУЛЬ 5 «ИНДИВИДУАЛЬНЫЙ ПРОЕКТ»

Модуль предназначен для формирования приёмов индивидуальной проектной деятельности, включая умения видеть проблему, формулировать тему и цель проекта, составлять план своей деятельности, осуществлять действия по реализации плана, результат деятельности соотносить с целью, делать выводы и заключения, доказывать, защищать свои идеи, оценивать результаты своей работы.

Цель модуля – формирование проектных компетенций обучающихся.

Задачи модуля:

- 1) Научить разрабатывать приложение «калькулятор».

- 2) Формировать умения работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения проектных задач.
- 3) Формировать целеустремленность, усидчивость в процессе проектной деятельности.

Ожидаемые предметные результаты освоения модуля

По окончании модуля обучающиеся

будут знать:

- принципы, методы и приемы разработки приложения «Калькулятор»;
- обратная польская запись;
- методы работы с информацией;

будут уметь:

- видеть и формулировать проблему, формулировать тему и цель проекта;
- составлять план своей деятельности,
- осуществлять действия по реализации плана,
- делать выводы и заключения,
- оценивать результаты своей работы;
- осуществлять презентацию собственного проекта;
- работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения проектных задач.

Учебно-тематический план модуля

№	Наименование тем	Количество часов		
		теория	практика	всего
1	Подготовительный этап проектирования. Формирование замысла проекта	2	2	4
2	Практический этап индивидуального проекта	2	14	16
3	Презентационный этап проектирования	1	3	4
	Итого по модулю:	5	19	24

Содержание учебного модуля

Тема 1. Подготовительный этап проектирования. Формирование замысла проекта.

Теория. Этапы создания собственных проектов. Функциональные возможности проекта. Особенности разработки проекта.

Практика. Формирование замысла проекта, формулирование темы проекта. Составление плана реализации проекта.

Тема 2. Практический этап индивидуального творческого проекта.

Теория. Инструменты проектирования. Методы сбора информации. Методы анализа информации.

Практика. Поэтапное создание творческого проекта в соответствии со своим планом. Сбор и обобщение информации. Анализ полученных материалов. Определение основных понятий. Структурирование материалов. Презентация вариантов проектов. Определение оптимального варианта. Доработка проекта.

Тема 3. Презентационный этап проектирования.

Теория. Этапы защиты проекта. Требования к мультимедийной презентации.

Практика. Рассмотрение примера презентаций, анализ выжимки проекта. Формирование презентации. Выделение основных аспектов индивидуального проекта.

Подведение итогов модуля. Презентация выполненных проектов.

Подведение итогов учебного года. Итоговая аттестация обучающихся по методу модульно-рейтинговой оценки. Защита индивидуальных проектов на итоговом мероприятии МБОУ ДО ГЦИР Фестивале интеллекта и творчества «Мы в Центре» в номинации «Ярмарка проектных идей «Время познания». Коллективное обсуждение итогов освоения программы и индивидуальное осмысление своей деятельности.

ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Кадровое обеспечение

Реализовывать программу может педагог, имеющий среднее специальное или высшее педагогическое образование по специальностям технического профиля, имеющий специальную подготовку в области ComputerScience и опыт работы с детьми старшего школьного возраста.

Для осуществления научного руководства исследовательскими работами детей или для консультирования по определенным темам к работе по программе могут привлекаться научные сотрудники высшей школы.

Методическое обеспечение

1. Педагогические технологии, методы, приемы и формы организации образовательного процесса

При реализации программы используются следующие педагогические технологии:

№	Педагогические технологии	Применение в программе
1-	Метод проектов	Отдельный модуль программы предназначен для организации деятельности обучающихся по методу проектов. Проектная технология используется при работе с отдельными группами детей или индивидуально с одаренным ребенком при подготовке к мероприятиям
2-	Технология обучения в сотрудничестве (обучение в малых группах)	Обучение в малых группах. Доклад малых групп.
3-	Информационные технологии. Использование программных средств и компьютеров для работы с информацией	Поиск, сбор и систематизация информации с использованием Интернет Использование специальных компьютерных программ и сред программирования для выполнения практических и проектных заданий

2. Методический кейс программы

Для реализации программы «Программирование на языке Python» сформирован методический кейс, который имеет следующие разделы и включает следующие материалы.

1) Нормативные и организационные документы, обеспечивающие эффективную организацию процесса освоения программы обучающимися:

- 1.1. Комплексы оздоровительно-профилактических упражнений, предотвращающих и снижающих утомление обучающихся (для среднего и старшего школьного возраста).
- 1.2. Инструкции по охране труда и технике безопасности.
- 1.3. Положение о проведении итогового мероприятия МБОУДОГЦИР Фестиваля интеллекта творчества «Мы в Центре».
- 1.4. Положения, приказы, информационные письма о проведении мероприятий различного уровня по профилю объединения.
- 1.5. Рекламные материалы объединения для организации набора (печатные, видео, в социальных сетях).

2) Методические материалы для педагога:

- 2.1. Сценарий проведения презентации объединения на Дне открытых дверей центра

цифрового образования «IT-куб».

2.2. Сценарий тематического занятия «Велика Россия» (мероприятие в рамках реализации Федерального календарного плана воспитательной работы «День народного единства»).

2.3. Сценарий тематического занятия в рамках реализации Федерального календарного плана воспитательной работы «День защитников Отечества».

2.4. Сценарий тематического занятия в рамках реализации Федерального календарного плана воспитательной работы «День Победы».

2.5. Сценарий праздника окончания учебного года в объединении.

2.6. Методические рекомендации по реализации коллективного учебного проекта.

2.7. Коллекция авторских медиапрезентаций по темам занятий (пример см. в Приложение 3 «Методические материалы»).

2.8. Материалы консультаций по следующей тематике:

2.8.1. Планирование деятельности в рамках проекта;

2.8.2. Как работать в команде?

2.8.3. Алгоритм работы со справочной литературой и Интернетом;

2.8.4. Оформление проектной папки;

2.8.5. Анализ информации;

2.8.6. Структура письменного отчета по проекту;

2.8.7. Анализ качества выполнения проекта.

3). Учебно-дидактические материалы, обеспечивающие реализацию содержания дополнительной образовательной программы:

№	Название материалов	Где используется: год обучения, модуль, тема	Цель использования
3.1.	Учебники и справочники «Изучаем Python» — Марк Лутц «Python для детей» — Джейсон Бриггс	Ознакомительный, базовый уровень Знакомство с базовыми конструкциями языка программирования	Теоретическая база, разбор примеров из учебника или справочника
3.2.	Интерактивные платформы LeetCode: Python Problems Codecademy: Learn Python	Все модули	Подбор задач для практики, самостоятельных и контрольных работ
3.3.	Проектные задания «100 Days of Code: The Complete Python Pro Bootcamp for 2023» — курс на Udemу от Angela Yu «Python Projects for Beginners» — книга от Дэна Беннисона	Коллективные, индивидуальные проекты	Разбор готовых проектов, поиск идеи для проекта

4) Оценочно-диагностические материалы для мониторинга освоения дополнительной образовательной программы обучающимися:

4.1. Контрольно-диагностические материалы для входной диагностики теоретических знаний (беседа «Что я знаю о программировании»).

4.2. Контрольно-диагностические материалы для оперативного контроля практических умений по каждой теме (пример компьютерного теста см. в приложении 2 «Оценочные материалы»).

4.3. Контрольно-диагностические материалы для промежуточного контроля освоения программы модуля (см. Приложение 2 «Оценочные материалы»).

4.4. Контрольно-диагностические материалы для проведения итоговой аттестации обучающихся (см. Приложение 2 «Оценочные материалы»).

4.5. Электронный журнал критериальных оценок.

4.6. Анкета для родителей «Удовлетворённость результатами посещения ребёнком занятий объединения».

Информационное обеспечение

1. Литература для обучающихся:

- 1) Пэйн, Б. Python для детей и родителей./ БрайсонПэйн; Пер. с англ. М.А.Райтман. – М.: Издательство «Э», 2017. – 352 с.
- 2) Бэрри, П. Изучаем программирование на Python. / Пол Бэрри; Пер. с англ. М.А. Райтман - М. : Издательство «Э», 2017. – 624 с. – (Мировой компьютерный бестселлер). [Электронный ресурс] / CoderNet – Режим доступа: https://codernet.ru/books/python/izuchaem_programmirovanie_na_python/

2. Литература для педагога:

Общепедагогическая, психологическая и методическая литература

- 1) Гин, А.А. Приёмы педагогической техники. Свобода выбора. Открытость. Деятельность. Обратная связь. Идеальность: Пособие для учителей / А.А. Гин. – М. : Вита-Пресс, 2022. – 112 с. – (Школа креативного мышления).
- 2) Золотарева, А.В. Методика преподавания по программам дополнительного образования детей. Учебник и практикум / А.В. Золотарева, Г.М. Криницкая, А.Л. Пикина – М. : Юрайт, 2023. — 315 с. – (Профессиональное образование).
- 3) Конасова, Н.Ю. Оценка результатов дополнительного образования детей. ФГОС. / Н.Ю. Конасова. - Волгоград: Учитель, 2016. – 121с. – (Образовательный мониторинг).
- 4) Леонтович, А.В. Исследовательская и проектная работа школьников. 5-11 класс / А.В.Леонтович, А.С. Саввичев – М.: ВАКО, 2018. – 160 с. – (Современная школа).
- 5) Михелькевич, В.Н. «Метод проектов» и его использование в средней общеобразовательной и высшей инженерной школах: учебное пособие / В.Н. Михелькевич, Н.В. Охтя. – Самара : Изд. Самарского государственного технического университета, 2004. – 48с.
- 6) Пинская, М.А. Оценивание для обучения: Практическое руководство / М.А. Пинская – М. : Чистые пруды, 2009. – 32с. – (Библиотечка «Первого сентября». Серия «Управление школой». Вып 28).
- 7) Селевко, Г.К. Энциклопедия образовательных технологий. / Г.К. Селевко. В 2-х томах. – М. : НИИ Школьные технологии, 2019. - 818с. – (Энциклопедия образовательных технологий).
- 8) Шаульская, Н.А. 2500 вопросов для школьных викторин. / Н.А. Шаульская. - Ростов-на-Дону : Феникс, 2013. – 256 с. – (Серия «Здравствуй, школа!»).

Специальная литература по программированию и методике информатики

- 1) Бриггс, Дж. Python для детей. Самоучитель по программированию. /Джейсон Бриггс; пер. с англ. Станислава Ломакина. – М. : Манн, Иванов и Фербер, 2017. – 320 с.
- 2) Васильев, А.Н. Программирование на Python в примерах и задачах. / А.Н. Васильев – М. : Бонбора, 2021. – 616 с. – (Российский компьютерный бестселлер).
- 3) Информатика. Задачник - практикум в 2т. / Под ред. И.Г. Семакина, Е.К. Хеннера; Том 1. – М. : Бином. Лаборатория Знаний, 2002. – 136 с.
- 4) Коглиати, Д. Python для непрограммистов. Самоучитель в примерах. /Джош Коглиати - М. : АСТ, 2024. – 96с. - (Учимся программировать).
- 5) Крылов, С.С., Ушаков, Д.М.. Информатика. Решение сложных задач. / ФИПИ. - М. : Интеллект-Центр, 2010. - 152с. - (Отличник ЕГЭ).
- 6) Окулов, С.М. Программирование в алгоритмах. / С.М. Окулов. – М. : Бингм. Лаборатория знаний, 2002. – 190 с.

3. Используемые интернет-ресурсы:

№	Интернет-адрес	Название ресурса	Где используется и для чего
1.	https://metanit.com/python/	Руководство по языку программирования Python	При изучении теории и в практической деятельности при решении задач

2.	http://informatics.msk.ru/	Центр педагогического мастерства. МЦНМО. Система автоматической проверки решений задач	В индивидуальной работе при изучении алгоритмов и решении задач
3.	https://lihomanenko.blogs.pot.com/	Блог преподавателя	При изучении теории в индивидуальной и коллективной формах обучения

Материально-техническое обеспечение программы

1. Компьютерный класс, удовлетворяющий санитарно-гигиеническим требованиям, для занятий группы 12 человек (рабочее место, укомплектованное компьютером; шкафы для УМК и оборудования). Комната для занятий должна быть хорошо освещена (естественным и электрическим светом). В кабинете должны быть созданы условия для безопасной работы за компьютерами (изолированные провода, система хранения компьютеров, отсутствие проводов на полу).

2. Для организации учебного процесса в рамках реализации программы согласно распоряжению «Об утверждении методических рекомендаций по созданию и функционированию центров цифрового образования «IT-куб» от 12 02 2021г. необходимо следующее оборудование:

Рабочее место преподавателя и каждого обучающегося:

- ноутбук с жёсткой неотключаемой клавиатурой;
- процессор: не менее 4-ёх ядер с частотой не менее 1 ГГц;
- объём установленной оперативной памяти должен быть не менее 8 Гбайт (до 24 Гбайт); объём поддерживаемой оперативной памяти (для возможности расширения): не менее 24 Гбайт;
- объём накопителя SSD: не менее 240 Гбайт;
- время автономной работы от батареи: не менее 6 часов;
- вес ноутбука с установленным аккумулятором: не более 1,8 кг;
- внешние интерфейсы: USB стандарта не ниже 3.0: не менее трёх свободных штук; сетевые и беспроводные интерфейсы: LAN, Wi-Fi (с поддержкой стандарта IEEE 802.11n или современнее);
- web-камера;
- предустановленная операционная система с графическим пользовательским интерфейсом, обеспечивающая работу распространённых образовательных и общесистемных приложений.

Дополнительное оборудование:

- МФУ (принтер черно-белый, цветной; сканер, ксерокс);
- интерактивный моноблочный дисплей с диагональю экрана не менее 65 дюймов и разрешением не менее 3840x2160 пикселей;
- Wi-Fi роутер;
- Мультимедийная проекционная установка;
- Флипчарт или магнитная доска;
- Цифровой фотоаппарат.

Прикладные программы:

- среда программирования Python 3.4 и выше.

Канцелярские принадлежности:

ручки, карандаши, маркеры, корректоры; блокноты, тетради; бумага разных видов и формата (А 3, А 4); клей; ножницы, степлеры, файлы, папки.

СПИСОК ЛИТЕРАТУРЫ, использованной при составлении программы

- 1) Баркова, И.В. Компьютерное программирование для подростков: Дополнительная программа. [Электронный ресурс] / Дворец творчества детей и молодежи. В помощь педагогу. – Режим доступа : <http://doto.ucoz.ru/metod/38-1-0-2090>.
- 2) Буйлова, Л.Н. Современные тенденции обновления содержания дополнительных общеобразовательных общеразвивающих программ. [Электронный ресурс] / Научная электронная библиотека КиберЛенинка. – Режим доступа: <https://cyberleninka.ru/article/n/sovremennye-tendentsii-obnovleniya-soderzhaniya-dopolnitelnyh-obscheobrazovatelnyh-obscherazvivayushih-programm/viewer>
- 3) Закон Российской Федерации «Об образовании» № 273-ФЗ от 26.12.2012 г. [Электронный ресурс] / Закон об образовании РФ. – Режим доступа : <http://zakon-ob-obrazovanii.ru/>
- 4) Концепция развития дополнительного образования детей. Распоряжение Правительства Российской Федерации от 31 марта 2022 г. № 678-р. [Электронный ресурс] / Интернет-портал «Правительство Российской Федерации» – Режим доступа : <http://static.government.ru/media/files/3fIgkklAJ2ENBbCFVEkA3cTOsiypicBo.pdf>
- 5) Методические рекомендации по подготовке дополнительных общеобразовательных общеразвивающих программ к прохождению процедуры экспертизы (добровольной сертификации) для последующего включения в реестр образовательных программ, включенных в систему ПФДО. [Электронный ресурс] / Региональный модельный центр дополнительного образования детей в Самарской области. Методические материалы. Проектирование дополнительных общеобразовательных программ. - Режим доступа: <http://surl.li/shwfz>
- 6) Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы). Письмо Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ № 09-3242 от 18.11.2015 г. [Электронный ресурс] / Региональный модельный центр дополнительного образования детей в Самарской области. Методические материалы. Проектирование дополнительных общеобразовательных программ. – Режим доступа: <http://surl.li/shwfz>
- 7) Методические рекомендации по проектированию разноуровневых дополнительных общеобразовательных общеразвивающих программ. /РМЦ ГБОУ ДО СО СДДЮТ – Самара, 2021 [Электронный ресурс] / Региональный модельный центр дополнительного образования детей в Самарской области. Методические материалы. Проектирование дополнительных общеобразовательных программ. - Режим доступа: <http://surl.li/shwfz>
- 8) Методические рекомендации по разработке дополнительных общеобразовательных общеразвивающих программ. Письмо Министерства образования и науки Самарской области № МО-1141-ТУ от 12.09.2022 года. [Электронный ресурс] / Региональный модельный центр дополнительного образования детей в Самарской области. Методические материалы. Проектирование дополнительных общеобразовательных программ. – Режим доступа: <http://surl.li/shwfz>
- 9) Методические рекомендации по реализации образовательных программ начального общего, основного общего, среднего общего образования, образовательных программ среднего профессионального образования и дополнительных общеобразовательных программ с применением электронного обучения и дистанционных образовательных технологий. Письмо Министерства просвещения РФ № ГД-39/04 от 19.03.2020 года. [Электронный ресурс] / Министерство просвещения Российской Федерации. Банк документов - Режим доступа: <https://docs.edu.gov.ru/document/26aa857e0152bd199507ffaa15f77c58/>

- 10) Положение о порядке разработки, экспертизы и утверждения дополнительной общеобразовательной общеразвивающей программы МБОУ ДО ГЦИР (утверждено приказом директора МБОУ ДО ГЦИР № 62 от 24.08.2020 г.) [Электронный ресурс] / Гуманитарный центр интеллектуального развития. Документы. – Режим доступа: <https://clck.ru/VXrd4>
- 11) Положение о проведении педагогического мониторинга, промежуточной и итоговой аттестации обучающихся (утверждено приказом директора МБОУ ДО ГЦИР № 88 от 07.12.2020 г.). [Электронный ресурс] / Гуманитарный центр интеллектуального развития. Документы. – Режим доступа: <https://clck.ru/VXrRg>
- 12) Положение об организации образовательного процесса с использованием дистанционных образовательных технологий (утверждено приказом директора МБОУ ДО ГЦИР № 78 от 28.08.2019 г.). [Электронный ресурс] / Гуманитарный центр интеллектуального развития. Документы. – Режим доступа: http://cir.tgl.ru/sp/pic/File/nast/Polozhenie_o_distante_2020_na_sayt.pdf
- 13) Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 г. № 28 "Об утверждении санитарных правил СП 2.4. 3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи"[Электронный ресурс] / Интернет-портал «Российская газета» - Режим доступа: <https://rg.ru/2020/12/22/rospotrebnadzor-post28-site-dok.html>
- 14) Постановление Главного государственного санитарного врача РФ от 28 января 2021 г. N 2 "Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания" [Электронный ресурс] / Информационно-правовой портал «Гарант.РУ» - Режим доступа: https://base.garant.ru/400274954/#block_1000
- 15) Приказ Министерства образования и науки РФ от 23 августа 2017 г. № 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ». [Электронный ресурс] / Информационно-правовой портал «Гарант.РУ» - Режим доступа: <https://base.garant.ru/71770012/>
- 16) Приказ Министерства образования и науки РФ от 27 июля 2022 г. № 629 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам». [Электронный ресурс] / Официальный интернет-портал правовой информации. Государственная система правовой информации. – Режим доступа : <http://publication.pravo.gov.ru/Document/View/0001202209270013>
- 17) Приказ Министерства просвещения РФ от 2 декабря 2019 г. № 649 «Об утверждении Целевой модели цифровой образовательной среды». [Электронный ресурс] / Информационно-правовой портал «Гарант.РУ» - Режим доступа: <https://www.garant.ru/products/ipo/prime/doc/73235976/>
- 18) Центры цифрового образования детей «It-куб». Банк документов [Электронный ресурс] / Академия Минпросвещения России - Режим доступа: <https://apkpro.ru/natsproektobrazovanie/bankdokumentov/>

ПРИЛОЖЕНИЯ

Приложение 1

Календарный учебный график программы

Календарный учебный график программы составлен в соответствии с локальным актом «Календарный учебный график МБОУ ДО ГЦИР городского округа Тольятти на 2024-2025уч.г.», принятым решением педагогического совета от 29 мая 2024 г., протокол № 3.

<i>Месяц</i>	<i>Содержание деятельности</i>	<i>Промежуточная и итоговая аттестация</i>
Сентябрь	Занятия по расписанию: 3 учебные недели. Начало занятий 09сентября	Входная диагностика знаний и практических навыков
Октябрь	Занятия по расписанию 5 учебных недель	
Ноябрь	Занятия по расписанию 4 учебные недели. Дополнительный день отдыха (государственный праздник) - 4 ноября	
Декабрь	Занятия по расписанию 4 учебные недели. В период школьных каникул с 30 декабря по 07 января: Новогодний праздник в объединении	
Январь	Занятия по расписанию 4 учебные недели. Дополнительные дни отдыха, связанные с государственными праздниками (выходные дни): 1-7 января	
Февраль	Занятия по расписанию 4 учебные недели. Дополнительный день отдыха (государственный праздник) - 23 февраля	
Март	Занятия по расписанию 4 учебные недели. Период школьных каникул с 22-31 марта. Дополнительный день отдыха (государственный праздник) - 8 марта	
Апрель	Занятия по расписанию 4 учебные недели	
Май	Занятия по расписанию 5 учебных недель. Участие в учрежденческом итоговом Фестивале интеллекта и творчества «Мы в Центре». Завершение учебных занятий 31 мая. Дополнительные дни отдыха, связанные с государственными праздниками –1 мая, 9 мая	Итоговая аттестация обучающихся
Июнь	Продолжение занятий по программе летней профильной смены Центра цифрового образования «It-куб» (по выбору обучающегося). Дополнительный день отдыха (государственный праздник) – 12июня	
Июль	Самостоятельные занятия учащихся	
Август	Формирование учебных групп до 10 сентября Предварительное тестирование вновь поступающих на программу	
Итого учебных недель	36 учебных недель	

Оценочные материалы**1. Контрольно-диагностические материалы
для проведения текущего контроля (пример компьютерного теста)****Тест по теории основных конструкций**

Основные конструкции Python (Баллов: 14)

1. Напишите вашу Фамилию Имя

Введите ответ

2. Как выглядит функция ввода с клавиатуры? (Баллов: 1)

☐ print()

☐ if

☒ input() ✓

☐ else

☐ int(input())

3. Как правильно записать в программе команду ввода для вещественного числа? (Баллов: 1)

☐ float(a)

☒ a = float(input()) ✓

☐ b = int(input())

☐ int(input(a))

**2. Контрольно-диагностические материалы
для проведения промежуточного контроля по итогам модуля****Пример задачи базового уровня**

1. Определить, введенное с клавиатуры число положительное, или отрицательное? Используйте условную конструкцию.
2. Является ли введенное число кратным пяти. Напишите программу по данному тексту
Ввод числа с клавиатуры
Если введенное число деленное на 5 дает остаток ноль, то
 вывести значение этой переменной
иначе
 вывести фразу “Данное число не кратно 5”

3. Контрольно-диагностические материалы для проведения итоговой аттестации

Итоговая контрольная работа

Итоговая аттестация проводится в конце учебного года. По результатам итоговой аттестации определяется уровень освоения учащимся содержания программы:

- 80% – 100% - высокий уровень.
- 60% -79% - средний уровень
- 59% и ниже – низкий уровень.

Параметры оценки:

Обучающийся должен **иметь представление:**

- о модулях, входящих в состав среды Python,
- о процессе исполнения каждого из операторов

Обучающийся должен **знать**

- особенности структуры программы и синтаксис языка программирования Python,
- назначение, устройство и свойства основных конструкций языка Python
- правила описания процедур и функций в Python
- правила описание классов и объектов (продвинутый уровень)

Обучающийся должен **уметь:**

- разрабатывать программы линейные, разветвляющиеся и с циклами
- уметь разрабатывать программы обработки числовой и символьной информации
- уметь работать в процедурной парадигме (базовый, продвинутый)
- уметь работать в объектно-ориентированной парадигме (продвинутый уровень)
- использовать сложные конструкции языка, упорядочивать данные, использовать динамическое программирование (продвинутый уровень)

Для ознакомительного уровня (5 задач):

1. Пирожок в столовой стоит, а рублей и b копеек. Определите, сколько рублей и копеек нужно заплатить за n пирожков.

Входные данные: Программа получает на вход три числа: a,b,n

Выходные данные: Программа должна вывести два числа: Стоимость покупки в рублях и копейках.

Входные данные:	Выходные данные:
10 15 2 (через enter)	20 30
2 50 4 (через enter)	10 0

2. Коля каждый день ложится спать ровно в полночь и недавно узнал, что оптимальное время для его сна составляет X минут. Коля хочет поставить себе будильник так, чтобы он прозвенел ровно через X минут после полуночи, однако для этого необходимо указать время сигнала в формате часы, минуты. Помогите Коле определить, на какое время завести будильник.

Входные данные:	Выходные данные:
480	8 0
512	8 32

3. Напишите программу, которая по введенному числу определит, какое сейчас время суток. Будем считать, что если введено число от 5 до 10, то это утро, от 11 до 17, то день,

с 18 до 22 – вечер, с 23 до 4 – ночь. Предусмотрите, что пользователь может ввести неверное число или вообще слово. Тогда ему стоит написать: Ошибка.

Входные данные: Вводится число.

Выходные данные: Если введено число от 5 до 10, то вывести Утро, от 11 до 17, то День, с 18 до 22 – Вечер, с 23 до 4 – Ночь. Если введено неверное число или слово, то вывести: Ошибка.

Входные данные:	Выходные данные:
5	Утро
24	Ошибка
Час дня	Ошибка

4. Цикл `for`. По данному действительному числу **a** и натуральному **n** вычислите сумму $1 + a + a^2 + \dots + a^n$, не используя формулу суммы геометрической прогрессии. Время работы программы должно быть пропорционально **n**.

Входные данные: Вводятся 2 числа **a** и **n**.

Выходные данные: Значение суммы.

Входные данные:	Выходные данные:
2 2	7

5. Цикл `while`. В первый день спортсмен пробежал **x** километров, а затем он каждый день увеличивал пробег на 10% от предыдущего значения. По данному числу **y** определите номер дня, на который пробег спортсмена составит не менее **y** километров.

Входные данные: Вводятся 2 числа **x** и **y**.

Выходные данные: Значение суммы.

Входные данные:	Выходные данные:
10 20	9

Контрольная работа обучающихся оценивается по следующим критериям:

- задания 1-2 по 1 баллу
- задания 3-4 по 2 балла
- задания 5 по 3 балла

Максимум– 9 баллов.

Критерии определения уровня освоения программы

<i>Низкий</i>	<i>Средний</i>	<i>Высокий</i>
0-2 балла	2-6 балла	6-9баллов

Для базового уровня (5 задач):

1. Реализуйте функцию **pair_sum(a, b)**, в которую поступают два целых числа и которая возвращает их сумму.

Входные данные:	Выходные данные:
<code>print(pair_sum(2, 5))</code>	7
<code>print(pair_sum(2, -2))</code>	0

2. Реализуйте функцию **min_int(a, b, c, d)**, в которую поступают 4 целых числа, из которых она находит наименьшее число.

Входные данные:	Выходные данные:
<code>print(min_int (4, 5, 6, 7))</code>	4

3. Напишите функцию, которая по введенному числу определит, какое сейчас время суток. Будем считать, что если введено число от 5 до 10, то это утро, от 11 до 17, то день, с 18 до 22 – вечер, с 23 до 4 – ночь. Предусмотрите, что пользователь может ввести неверное число или вообще слово. Тогда ему стоит написать: Ошибка.

Не забудьте, что в данной задаче числа – это тоже строки.

Входные данные: Вводится строка.

Выходные данные: Если введено число от 5 до 10, то вывести Утро, от 11 до 17, то День, с 18 до 22 – Вечер, с 23 до 4 – Ночь. Если введено неверное число или слово, то вывести: Ошибка.

Входные данные:	Выходные данные:
5	Утро
24	Ошибка
Час дня	Ошибка

4. Дано целое число n. Найти квадрат данного числа, используя для его вычисления следующую формулу: $n^2 = 1 + 3 + 5 + \dots + (2 \cdot N - 1)$ После добавления к сумме каждого слагаемого выводить текущее значение суммы

Контрольная работа обучающихся оценивается по следующим критериям:

- задания 1-2 по 1 баллу
- задания 3-4 по 2 балла
- задания 5 по 3 балла

5. Даны положительные числа A и B ($A > B$). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Не используя операции умножения и деления, найти длину незанятой части отрезка A

Входные данные:	Выходные данные:
17	2
5	

Максимум – 9 баллов.

Критерии определения уровня освоения программы

Низкий	Средний	Высокий
0-2 балла	2-6 балла	6-9баллов

Для продвинутого уровня (5 задач):

1. На вершине лесенки, содержащей N ступенек, находится мячик, который начинает прыгать по ним вниз, к основанию. Мячик может прыгнуть на следующую ступеньку, на ступеньку через одну или через (То есть, если мячик лежит на 8-ой ступеньке, то он может переместиться на 5-ую, 6-ую или 7-ую.) Определить число всевозможных «маршрутов» мячика с вершины на землю

Вводится одно число $0 < N < 31$.	Ввод	4
Выведите одно число — количество маршрутов	Вывод	7

2. Необходимо написать программу, в которой сортировка выполняется «методом камня» – самый «тяжёлый» элемент опускается в конец массива. Нельзя использовать встроенные функции: **исходный массив:** [8, 2, 9, 5, 6, 7, 0, 4], **результат:** [0, 2, 4, 5, 6, 7, 8, 9]
3. Слова-палиндромы — это строки, которые одинаково читаются с обеих сторон. Например: «aba», «abba», «abcba», «aaaa», «ababa» и так далее.

Напишите рекурсивную функцию `is_palindrome(string)`, которая будет принимать строку и возвращать «True», если строка является палиндромом, и «False» во всех других случаях.

4. Система учета студентов**

1. Создайте текстовый файл "students.txt", в котором каждая строка содержит информацию о студенте в формате: "Имя Фамилия, Возраст, Средний балл".

2. Напишите программу на Python, которая будет выполнять следующие действия:

- Чтение данных из файла "students.txt" и вывод списка студентов на экран.
- Добавление нового студента в файл.
- Поиск студента по имени и вывод его информации.
- Удаление студента из файла.

5. Класс "Банковский счет". Создать класс "Банковский счет" с атрибутами для номера счета, баланса, и методами для пополнения счета, снятия денег и вывода информации о счете.

1. Инкапсуляция:

Добавьте приватные атрибуты для номера счета и баланса, и методы для доступа к ним через геттеры и сеттеры.

2. Наследование:

Создайте дочерний класс "Сберегательный счет", который унаследует функциональность класса "Банковский счет" и добавит свои уникальные методы, например, для начисления процентов.

3. Полиморфизм:

Реализуйте методы для снятия денег как в классе "Банковский счет", так и в классе "Сберегательный счет", но с разной логикой (например, наличие комиссии при снятии денег со сберегательного счета).

Контрольная работа обучающихся оценивается по следующим критериям:

- задание 1-3 по 3 балла
- задания 4-5 по 5 баллов

Максимум – 19 баллов.

Критерии определения уровня освоения программы

<i>Низкий</i>	<i>Средний</i>	<i>Высокий</i>
0-3балла	3-9балла	9-19 баллов

Методические материалы

1. Наглядные материалы по теме «Списки», модуль «Базовые конструкции в python» (базовый уровень)



Базовые типы данных

Списки в Python - упорядоченные изменяемые коллекции объектов произвольных типов (почти как массив, но типы могут отличаться).

Чтобы использовать списки, их нужно создать. Создать список можно несколькими способами. Например, можно обработать любой итерируемый объект (например, строку) встроенной функцией `list`:

Пример:

```
>>> list('список')
['с', 'п', 'и', 'с', 'о', 'к']
```



Списки

Для хранения таких данных можно использовать структуру данных, называемую в Питоне список. Список представляет собой последовательность элементов, пронумерованных от 0, как символы в строке. Список можно задать перечислением элементов списка в квадратных скобках, например, список можно задать так:

```
Primes = [2, 3, 5, 7, 11, 13]
Rainbow = ['Red', 'Orange', 'Yellow', 'Green', 'Blue', 'Indigo', 'Violet']
```

В списке `Primes` — 6 элементов, а именно,

```
Primes[0] == 2, Primes[1] == 3, Primes[2] == 5, Primes[3] == 7,
Primes[4] == 11, Primes[5] == 13.
```



Функции и методы списка

```
a = [5, 3, 1]
```

```
length = len(a)
```

```
print(length)
```

```
sort = sorted(a)
```

```
print(sort)
```

Встроенная функция	Описание
<code>len()</code>	Возвращает длину списка
<code>sorted()</code>	Возвращает отсортированный список (по возрастанию)
<code>min()</code> и <code>max()</code>	Возвращают наименьший и наибольший элемент списка

```
max_a = max(a)
```

```
min_a = min(a)
```

```
print(max_a, min_a)
```



2. Примеры задач

2.1. Задача на работу со списком

Задача 1

Дан список некоторых целых чисел, найдите значение 20 в нем и, если оно присутствует, замените его на 200. Обновите список только при первом вхождении числа 20.

Мы можем использовать метод `index()`, который позволит получить индекс первого вхождения некоторого объекта (в нашем случае числа 20). Затем просто изменим один элемент списка с этим индексом до нужного нам значения (то есть 200).

```
list1 = [5, 10, 15, 20, 25, 50, 20] #список
# переменная индекс использующая метод index()
index = list1.index(20)
list1[index] = 200
print(list1)
```

