

Муниципальное бюджетное общеобразовательное учреждение
"Лебяжьевская средняя общеобразовательная школа"

Рассмотрено на
Педагогическом Совете
школы
Протокол № 2 от «27»
августа 2024 года

Принята
решением Педагогического
Совета школы Протокол №
2 от «27» августа 2024 года
Н.В. Гончарова

Утверждена
приказом №100/4 от «27»
августа 2024 года
Директор школы: _____
Н.В. Гончарова



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
(ОБЩЕРАЗВИВАЮЩАЯ) ПРОГРАММА**
технической направленности
«ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON»
Уровень усвоения программы: базовый
Возраст обучающихся 14-15 лет
Срок реализации 1 год

Составитель: Углова Т.В., учитель
информатики

п. Лебяжье, 2024 г.

ПАСПОРТ ПРОГРАММЫ

Наименование программы	"Основы программирования на Python"
Детское объединение	"Основы программирования на Python"
Тип программы	Дополнительная общеобразовательная (общеразвивающая)
Направленность программы	Техническая
Образовательная область	Познавательное развитие
Вид программы	Модифицированная
Срок обучения	1 год
Возраст учащихся	14-15 лет
Уровень освоения	Базовый
Цель программы	Формирование у обучающихся базовых навыков программирования на языке программирования Python для решения практических задач и разработки программных продуктов.
С какого года реализуется	2024

СОДЕРЖАНИЕ

Паспорт программы.....	2
1. Комплекс основных характеристик программы	
1.1. Пояснительная записка.....	4
1.2. Цель и задачи программы.....	8
1.3. Содержание программы	9
1.4. Планируемые результаты	12
2. Комплекс организационно-педагогических условий	
2.1. Календарный учебный график.....	13
2.2. Условия реализации программы.....	15
2.3. Формы аттестации.....	16
2.4. Оценочные материалы.....	17
2.5. Методические материалы.....	20
3. Список литературы.....	21

1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

1.1. Пояснительная записка

Современное общество переживает активную стадию цифровой трансформации. Все больше сфер жизни людей становятся зависимыми от информационных технологий и электроники.

Одной из составляющих информационной компетентности является владение языком программирования, вследствие чего встает вопрос о выборе языка программирования, который отвечает современным требованиям к написанию программ, служит основой для дальнейшего развития и совершенствования навыков программирования и удобен в освоении подростками.

Язык программирования Python принято считать одним из самых простых в освоении. Обучаться языку возможно имея лишь базовые навыки использования компьютера, установка необходимого ПО не сложнее установки любой другой программы, а синтаксис языка не перегружен и интуитивно понятен. Данные факторы раскрывают Python максимально эффективным инструментом при обучении детей программированию.

При этом Python является современным, востребованным и перспективным языком разработки, позволяющим не только заложить фундамент знаний будущих программистов и инженеров, но получить востребованные и актуальные навыки, необходимые в учебной, проектной и повседневной деятельности.

Направленность программы

Дополнительная общеобразовательная программа «Основы программирования на Python» имеет техническую направленность, что позволяет обучающимся приобщиться к инженерно-техническим знаниям в области инновационных технологий, сформировать техническое мышление.

Основанием для проектирования и реализации данной общеразвивающей программы служит *перечень следующих нормативных правовых актов и государственных программных документов:*

- Федеральный закон от 24.07.1998 № 124-ФЗ «Об основных гарантиях прав ребёнка в РФ»;
- Федеральный закон от 29.12.2012 N 273-ФЗ «Об образовании в Российской Федерации»;
- Распоряжение Правительства РФ от 31.03.2022 №678-р «Об утверждении Концепции развития дополнительного образования детей до 2030 года»;
- Распоряжение Правительства РФ от 29.05.2015 N 996-р «Об утверждении Стратегии развития воспитания в Российской Федерации на период до 2025 года»;
- Приказ Министерства Просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным

программам»;

- Приказ Министерства Просвещения Российской Федерации от 03.09.2019 г. №467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;
- Приказ Министерства труда и социальной защиты Российской Федерации от 05.05.2018 г. № 298н «Об утверждении профессионального стандарта «Педагог дополнительного образования детей и взрослых»;
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 г. № 28 «Об утверждении СанПиН 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
- Письмо Министерства образования и науки РФ от 18.11.2015г. № 09-3242 «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)».

Актуальность программы обусловлена повышенным спросом на изучение языков программирования детьми, в частности языка Python.

Python изучается в школьном курсе информатики, необходим для решения олимпиадных и конкурсных заданий, а также сдачи ОГЭ и ЕГЭ, имеет прикладной характер и может использоваться для решения повседневных задач. Кроме того, Python является востребованным языком программирования, используемым профессиональными инженерами во многих сферах IT-индустрии, поэтому знание данного языка даже на базовом уровне повышает шансы будущих выпускников на трудоустройство.

Отличительная особенность программы заключается в возможности получения обучающимися универсальных компетенций, необходимых при дальнейшем изучении не только информационных технологий, но и предметов гуманитарного и естественно-научного цикла. Также в результате изучения парадигмы объектно-ориентированного подхода к программированию происходит формирование базовых знаний и умений для работы с большинством популярных языков и необходимых при освоении других IT-направлений.

Программа «Основы программирования на Python» является практико-ориентированной. Освоение подростками навыков программирования происходит в процессе практической и самостоятельной работы. Это позволяет обучающимся получать не только теоретические знания в области программирования, но и уверенно овладевать навыками и инструментами разработки продуктов.

Адресат общеразвивающей программы

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на Python» предназначена для детей в возрасте 14–15 лет (9 класс), мотивированных к обучению и проявляющих интерес к IT-технологиям, приобретению навыков программирования.

Формы занятий групповые. Количество обучающихся в группе – 12-15 человек. Состав групп постоянный.

Для успешной реализации дополнительной общеразвивающей программы используются **методы обучения**:

- словесные: объяснение, беседа;
- наглядные: демонстрации;
- практические работы.

Разнообразные методы предъявления учебно-познавательного материала делают содержание доступным, интересным и привлекательным для подростков.

Методы:

- Объяснительно-иллюстративный – предъявление информации различными способами (объяснение, рассказ, беседа, инструктаж, демонстрация, работа с технологическими картами и др.);
- Эвристический – метод творческой деятельности (создание творческих работ и т.д.)
- Проблемный – постановка проблемы и самостоятельный поиск её решения воспитанниками;
- Программированный – набор операций, которые необходимо выполнить в ходе выполнения практических работ (форма: компьютерный практикум, проектная деятельность);
- Репродуктивный – воспроизводство знаний и способов деятельности (по образцу, упражнения по аналогу),
- Частично-поисковый – решение проблемных задач с помощью педагога;
- Поисковый – самостоятельное решение проблем;
- Метод проблемного изложения – постановка проблемы педагогом, решение ее самим педагогом, соучастие обучающихся при решении.
- Метод проектов – технология организации образовательных ситуаций, в которых воспитанник ставит и решает собственные задачи, технология сопровождения самостоятельной деятельности воспитанника.

Методы воспитания: мотивация, убеждение, поощрение, упражнение, стимулирование, создание ситуации успеха и др.

Организационные формы обучения - групповые занятия, индивидуальные или всем составом.

Занятия по Программе проводятся в группе, сочетая принцип группового обучения с индивидуальным подходом.

Занятие делится на две части. Учащиеся сначала знакомятся с теоретическими знаниями, затем 15-20 минут работают за компьютерами в среде программирования, ученики работают индивидуально по 1 человеку, иногда проводится работа в парах, что способствует формированию коммуникативной компетенции учащихся. Во время занятий проводятся профилактические упражнения для зрения (5 мин) и осанки (3 - 7 мин).

Режим занятий, периодичность и продолжительность занятий: длительность одного занятия составляет 1 академический час, периодичность занятий – 1 раз в неделю.

Срок освоения общеразвивающей программы: определяется содержанием программы и составляет 1 год.

Форма обучения: очная, возможна реализация очно с применением электронного обучения и дистанционных образовательных технологий (Закон №273-ФЗ, гл.2, ст.17, п.2.).

Объём общеразвивающей программы: общее количество учебных часов, запланированных на весь период обучения, необходимых для освоения программы: 34 академических часа.

По уровню освоения программа общеразвивающая, **одноуровневая** (стартовый уровень). Обеспечивает возможность обучения детей с любым уровнем подготовки.

Зачисление детей на обучение производится без предварительного отбора (свободный набор).

Стартовый уровень предполагает использование и реализацию общедоступных и универсальных форм организации материала, минимальную сложность предлагаемого материала для освоения содержания программы.

Осваивая программу стартового уровня, обучающиеся изучают базовые принципы программирования, разработки проектов и построения программных продуктов, структуры и устройства компьютеров. В результате освоения программы обучающиеся приобретут навыки программирования, работы с прикладным ПО, применением языка Python в повседневной и учебной деятельности и эффективного анализа информации. Научатся работать в команде, представлять результаты собственной работы.

Знания и умения, приобретённые в результате освоения программы, могут быть использованы обучающимися при сдаче ОГЭ, ЕГЭ, при участии в олимпиадах по программированию, при решении задач по информатике, физике, математике, лингвистике и другим наукам, а также они являются фундаментом для дальнейшего совершенствования мастерства программирования.

Данная программа является базой для перехода на более сложные программы обучения. Так, по итогам успешного освоения программы «Основы программирования на Python», обучающийся может сдать вступительное тестирование и быть зачислен на общеразвивающую программу «Основы программирования на языке Python. «Лицей Академии Яндекса»» (базового уровня), которая представляет собой более углубленное изучение уже освоенного материала в рамках программы.

Обучающиеся повышают уровень компьютерной грамотности и цифровой культуры, также полученные знания и навыки являются актуальными и востребованными при дальнейшем обучении подростков в СПО. Осваивая данную программу, обучающиеся будут овладевать современными навыками, необходимым как в повседневной и учебной деятельности, так для дальнейшего развития в качестве IT-специалистов.

1.2. Цель и задачи программы

Цель программы: формирование у обучающихся базовых навыков программирования на языке программирования Python для решения практических задач и разработки программных продуктов.

Для успешной реализации поставленной цели необходимо решить ряд обучающих, развивающих и воспитательных задач:

Задачи:

Обучающие:

- познакомить с основными предметными понятиями программирования, компьютерных наук и их свойствами;
- познакомить с базовым синтаксисом и инструментарием языка программирования Python, необходимых для решения практических задач и разработки продуктов;
- сформировать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- познакомить с базовыми конструкциями и принципами объектно-ориентированного программирования.

Развивающие:

- развить навыки алгоритмического и критического мышления;
- сформировать и развить навыки работы с различными источниками информации, умения самостоятельно искать, извлекать и отбирать необходимую информацию;
- развить умения планирования, структурирования и разработки проектов, навыков организации и реализации проектной деятельности;
- сформировать трудовые умения и навыки, умение планировать работу, предвидеть результат и достигать его;
- познакомить с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой и оборудованием.

Воспитательные:

- способствовать воспитанию корректного поведения в обществе, социальных норм, ролей и понимания форм социального взаимодействия в группах;
- способствовать воспитанию уважительного и продуктивного учебного сотрудничества и совместной деятельности со сверстниками в процессе проектной и учебно-исследовательской деятельности;
- способствовать формированию понимания необходимости организованного и ответственного отношения к учению, труду, другому человеку, его мнению и деятельности.

1.3. Содержание программы

Тема 1. Знакомство с языком Python (2 ч.)

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программ на языке Python. Комментарии.

- ✓ Практическая работа 1.1. Установка программы Python.
- ✓ Практическая работа 1.2. Режимы работы с Python

Учащиеся должны знать / понимать:

понятие программы; структура программы на Python; режимы работы с Python.

Учащиеся должны уметь:

выполнить установку программы; выполнить простейшую программу в интерактивной среде; написать комментарии в программе.

Тема 2. Переменные и выражения (6 ч.)

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция. Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

- ✓ Практическая работа 2.1. Переменные
- ✓ Практическая работа 2.2. Выражения
- ✓ Практическая работа 2.3. Задачи на элементарные действия с числами

Самостоятельная работа 1 «Решение задач на действия с числами».

Учащиеся должны знать / понимать:

общую структуру программы; типы данных; целые, вещественные типы данных и операции над ними; оператор присваивания; операторы ввода-вывода.

Учащиеся должны уметь:

пользоваться интерфейсом среды программирования Python; использовать команды редактора; организовывать ввод и вывод данных; записывать арифметические выражения.

Тема 3. Условные операторы (6 ч.)

Логический тип данных. Логические выражения и операторы. Сложные

условные выражения (логические операции and, or, not). Условный оператор. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

- ✓ Практическая работа 3.1. Логические выражения
- ✓ Практическая работа 3.2. "Условный оператор"
- ✓ Практическая работа 3.3. Множественное ветвление

Самостоятельная работа 2. Решение задач по теме "Условные операторы".

Учащиеся должны знать / понимать:

назначение условного оператора; способ записи условного оператора; логический тип данных; логические операторы or, and, not;

Учащиеся должны уметь:

использовать условный оператор; создавать сложные условия с помощью логических операторов.

Тема 4. Циклы (9 ч.)

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов. Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

- ✓ Практическая работа 4.1. "Числа Фибоначчи"
- ✓ Практическая работа 4.2. Решение задачи с циклом for.
- ✓ Практическая работа 4.3. Реализация циклических алгоритмов
- ✓ Практическая работа 4.4. Случайные числа
- ✓ Практическая работа 4.5. Решение задач с циклом.

Самостоятельная работа 3 по теме "Циклы"

Учащиеся должны знать / понимать:

циклы с условием и их виды; правила записи циклов с условием; назначение и особенности использования цикла с параметром; формат записи цикла с параметром; примеры использования циклов различных типов.

Учащиеся должны уметь:

определять вид цикла, наиболее удобный для решения поставленной задачи; использовать цикл с условием; определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи;

Тема 5. Функции (5 ч.)

Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций. Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

- ✓ Практическая работа 5.1. Создание функций
- ✓ Практическая работа 5.2. Решение задач с использованием функций
- ✓ Практическая работа 5.3. Рекурсивные функции

Учащиеся должны знать / понимать:

понятие функции; способы описания функции; принципы структурного программирования; понятие локальных переменных подпрограмм; понятие формальных и фактических параметров подпрограмм; способ передачи параметров.

Учащиеся должны уметь:

создавать и использовать функции; использовать механизм параметров для передачи значений.

Тема 6. Строки (4 ч.)

Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

- ✓ Практическая работа 6.1. Строки
- ✓ Практическая работа 6.2. Решение задач со строками.

Учащиеся должны знать / понимать:

назначение строкового типа данных; операторы для работы со строками; процедуры и функции для работы со строками; операции со строками.

Учащиеся должны уметь:

описывать строки; соединять строки; находить длину строки; вырезать часть строки; находить подстроку в строке; находить количество слов в строке.

Тема 7. Итоговое тестирование по курсу (2 ч.)

1.4. Планируемые результаты

Предметные результаты:

- знание основных предметных понятий программирования, компьютерных наук и их свойств;
- знание базового синтаксиса и инструментария языка программирования Python, умение применять язык программирования Python на практике;
- умение применять объектно-ориентированную парадигму в программировании;
- навык разработки эффективных алгоритмов и программ на основе изучения языка программирования Python.

Личностные результаты:

- способность организовывать учебное сотрудничество и совместную деятельность со сверстниками в процессе проектной и учебно-исследовательской деятельности;
- понимание необходимости уважительного, организованного и ответственного отношения к учению, труду, другому человеку, его мнению и деятельности;
- умение алгоритмически и логически мыслить;
- знание правил поведения, социальных норм, ролей и форм социального взаимодействия в группах.

Метапредметные результаты:

- умение работать с различными источниками информации, извлекать и анализировать необходимую информацию из открытых источников;
- способность составлять и изменять план действий, необходимый для достижения цели, предвидеть результат и достигать его;
- умение выполнять проекты в соответствии с техническим заданием;
- знание правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой и оборудованием.

2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

2.1. Календарный учебный график

Учебный план

№	Название раздела, темы	Количество часов		
		Всего	Теория	Практика
1.	Знакомство с языком Python	2	1	1
2	Переменные и выражения	6	2	4
3	Условные операторы	6	2	4
4	Циклы	9	3	6
5	Функции	5	2	3
6	Строки	4	2	2
7	Итоговое тестирование по курсу	2		2
		34	12	22

Календарно-тематический план

№ урока	Дата	Кол-во часов	Тема урока	Формы аттестации/ контроля
Тема 1. Знакомство с языком Python (2 ч.)				
1		1	Общие сведения о языке Python. Практическая работа 1.1. Установка программы Python.	Опрос, установка программы
2		1	Что такое программа. Структура программ на языке Python. Практическая работа 1.2. Режимы работы с Python	Опрос, решение задач
Тема 2. Переменные и выражения (6ч.)				
3		1	Переменные. Практическая работа 2.1. Переменные	решение задач
4		1	Выражения. Ввод и вывод	разбор ситуаций
5		1	Практическая работа 2.2. Выражения	решение задач
6		1	Задачи на элементарные действия с числами действия с числами	разбор ситуаций
7		1	Практическая работа 2.3. Задачи на элементарные действия с числами.	решение задач
8		1	<i>Самостоятельная работа 1 «Решение задач на действия с числами».</i>	решение задач
Тема 3. Условные операторы				
9		1	Логические выражения и операторы	разбор ситуаций
10		1	Практическая работа 3.1. Логические выражения	решение задач

11		1	Условный оператор. Множественное ветвление.	решение задач
12		1	Практическая работа 3.2. "Условный оператор"	решение задач
13		1	Практическая работа 3.3. Множественное ветвление	решение задач
14		1	<i>Самостоятельная работа 2. Решение задач по теме "Условные операторы".</i>	решение задач
Тема 4. Циклы				
15		1	Оператор цикла с условием.	разбор ситуаций
16		1	Практическая работа 4.1. "Числа Фибоначчи"	решение задач
17		1	Оператор цикла for.	решение задач
18		1	Практическая работа 4.2. Решение задачи с циклом for.	решение задач
19		1	Вложенные циклы. Случайные числа.	разбор ситуаций
20		1	Практическая работа 4.3. Реализация циклических алгоритмов.	решение задач
21		1	Практическая работа 4.4. Случайные числа.	решение задач
22		1	Практическая работа 4.5. Решение задач с циклом	решение задач
23		1	<i>Самостоятельная работа 3 по теме "Циклы".</i>	решение задач
Тема 5. Функции				
24		1	Создание функций. Локальные переменные	разбор ситуаций
25		1	Практическая работа 5.1. Создание функций	решение задач
26		1	Практическая работа 5.2. Решение задач с использованием функций	решение задач
27		1	Рекурсивные функции	разбор ситуаций
28		1	Практическая работа 5.3. Рекурсивные функции	решение задач
Тема 6. Строки				
29		1	Строки	разбор ситуаций
30		1	Практическая работа 6.1. Строки	решение задач
31		1	Срезы строк	разбор ситуаций
32		1	Практическая работа 6.2. Решение задач со строками.	решение задач
Тема 7. Итоговое тестирование по курсу				
33-34		2	Итоговый тест по курсу «Основы программирования на Python»	тестирование

2.2. Условия реализации программы

Для реализации программы имеется необходимое материально-техническое обеспечение. Занятия проводятся в хорошо освещенном компьютерном классе, оборудованном компьютерными столами, удобными креслами, а также необходимыми техническими средствами: компьютеры, проектор, подключенный к ПК, МФУ (принтер, сканер, колонки).

Для успешного претворения Программы имеется **программное обеспечение:**

- Операционная система Windows, Astra Linux
- Язык программирования Python 3.7.2

Средства обучения:

- Компьютер, проектор, сканер, принтер, Интернет Wi-Fi роутер
- Раздаточный материал (карточки - задачи)
- Аудиовизуальные средства (презентации, фильмы)
- Готовые файлы с заданиями

Для организации процесса обучения используются следующие формы учебных занятий: теоретические занятия; практические занятия.

Для успешной реализации дополнительной общеразвивающей программы используются методы обучения: словесные: объяснение, беседа, диалог; наглядные: демонстрации; практические работы.

Результаты: по окончании обучения учащиеся должны продемонстрировать сформированные умения и навыки работы в графическом редакторе и применять их в практической деятельности и повседневной жизни.

2.3. Формы аттестации

Промежуточная аттестация учащихся:

Тестовый контроль, фронтальная и индивидуальная беседа, цифровой, графический диктанты, игровые формы контроля. Диагностический инструментарий промежуточной аттестации учащихся представлен тестовыми заданиями, мини-опросами, проводимыми во время занятий-практикумов, цифровыми, графическими и терминологическими диктантами.

Итоговая аттестация учащихся:

Анкетирование, тестирование, практические и самостоятельные работы. Теоретические знания оцениваются через беседы с обучающимися.

Формы отслеживания и фиксации результатов: материал тестирования, файлы практических и самостоятельных работ, грамоты, дипломы, свидетельства (сертификаты). Способами фиксации результатов деятельности являются успехи обучающегося об участиях, победах в конкурсах, соревнованиях, олимпиадах и др. мероприятиях.

Оценка образовательных результатов учащихся по Программе (***текущий контроль***) проводится в следующих формах: беседа, устный опрос, тестирование, самостоятельная работа, практическая работа, контрольное занятие.

Формы промежуточной аттестации: тестирование, самостоятельная работа, практическая работа, контрольное занятие, анализ задач, тестирование.

Формы итоговой аттестации: итоговое тестирование.

Текущий контроль ведется на каждом занятии. Просмотр решенных задач, поиск ошибок. Учащиеся учатся находить ошибки своих товарищей, оценивать их работы.

2.4. Оценочные материалы

Оценочные материалы: вопросы для устного опроса, тесты, задания для самостоятельных, контрольных, практических, проверочных работ.

Приложение 1

Вводная диагностика

(максимум-10 баллов)

ФИО _____ Дата _____

1. К расширениям аудиофайлов относятся: (1 балл)

1) exe, txt, msi 2) flac, aac, mpa 3) iso, odt, ai

2. Опишите своими словами, что такое язык программирования: (2 балла)

3. Сколько килобайт в гигабайте? (1 балл)

4. К архитектурам процессора относятся: (2 балла)

1) X86 2) RISC 3).NET 4)AMD

5. Среда разработки программного обеспечения? (2 балл)

6. Соотношение пикселей по сторонам 4K-разрешения: (1 балл)

1) 2016 x 1080 2) 3960 x 2080 3) 2048 x 1080

7. Элемент компьютера, способный хранить информацию только при включенном компьютере? (1 балл)

1) Процессор 2) Оперативная память 3) Жесткий диск 4) SSD

Пример промежуточной аттестации

(максимум – 50 баллов)

Задача: Ханойская башня (25 баллов)

Есть три стержня. На одном из которых нанизаны диски. Диски располагаются в виде пирамидки (ханойской башни): в самом низу лежит самый большой диск, затем



идёт чуть поменьше диск, затем ещё меньше диск и т. д. Необходимо переместить диски с одного стержня на другой. Можно использовать все три стержня, но при условии: перекладывать можно только по одному диску за ход, складывать диски можно только меньший на больший.

Тестирование:

1. Что будет в выводе данного кода? (3 балла)

```
x = 18
```

```
num = 0 if x > 21 else 26
```

```
print(num)
```

- 1) null 2) 0 3) 26 4) 18

2. Опишите своими словами, чем интерпретируемые языки отличаются от компилируемых? (3 балла)

3. Функция длины строки в Python: (2 балла)

- 1) len('human') 2) get('human') 3) array ['human'] 4) print('human')

8. Язык программирования Python: (2 балла)

- 1) Строго типизированный и интерпретируемый
- 2) Динамически типизированный и компилируемый
- 3) Строго типизированный и компилируемый
- 4) Динамически типизированный и интерпретируемый

9. Опишите своими словами сферы применения языка программирования Python (5 баллов)

10. Опишите своими словами, чем отличаются высокоуровневые языки программирования от низкоуровневых? Приведите примеры. (10 баллов)

Пример итоговой аттестации

(максимум – 25 баллов)

Задача: Вычисление n-го числа ряда Фибоначчи с помощью цикла while (15 баллов) Числа Фибоначчи – это ряд чисел, в котором каждое следующее число равно сумме двух предыдущих. *0, 1, 1, 2, 3, 5, 8, ...*

Тестирование:

1. Что будет в выводе данного кода?

(1 балл) for j in 'Hi! I\'m misses Rosa':

```
    if j == "\":
```

```
        print('Выполнено!')
```

```
    break
```

```
    else:
```

```
        print ('Здорово!')
```

- 1) Здорово
- 2) Ошибку
- 3) Выполнено
- 4) Ничего

2. Что такое аргументы и параметры функции? Приведите примеры (2 балла)

3. Метод Python позволяющий вставить в указанные места указанные аргументы, с выполнением их предварительного форматирования: (1 балл)

1) str.casefold()

2) str.encode(encoding="utf-8", errors="strict")

3) str.endswith(suffix[, start[, end]])

4) str.format(*args, **kwargs)

4. К расширениям архивов относятся: (1 балл)

1) exe, txt, msi

2) rar, 7zip, bin

3) iso, odt, ai

5. Опишите приоритет операций в языке программирования Python. (2 балла)

6. Опишите своими словами принцип работы процессора и оперативной памяти. (3 балла)

2.5. Методические материалы

Для реализации программы используются разнообразные формы и методы проведения занятий. Это рассказ, беседы, лекции, из которых дети узнают много новой информации; практические задания для закрепления теоретических знаний и реализации практических навыков в написании программ. Занятия сопровождаются использованием наглядного материала. Разнообразные занятия дают возможность детям проявить свою индивидуальность, самостоятельность, способствуют гармоничному и духовному развитию личности.

Технические средства обучения: компьютеры, проектор, принтер, сканер.

Дидактические материалы: методические пособия, разработанные преподавателем с учётом конкретных задач, варианты демонстрационных программ, материалы по терминологии ПО, учебная литература.

3. Список литературы

Список литературы, использованной при написании программы:

1. Архитектура компьютера, Таненбаум Эндрю, Остин Тодд – СПб.: Прогресс книга, 2022 – 816 с.;
2. Информатика, Тимофеева Е.В. М.: Эксмо, 2021 – 176 с.;
3. Python, например, Никола Лейси, – СПб.: Питер, 2021 – 192 с.;
4. Ли Воган «Непрактичный» Python занимательные проекты для тех, кто хочет поумнеть. – СПб.: БХВ-Петербург, 2021. – 457 с..

Электронные ресурсы:

1. Питонтьютор. Бесплатный курс по программированию с нуля. //: <https://pythontutor.ru/>
2. Code Basics: обучение базовым аспектам языков программирования от образовательной платформы Hexlet. //: <https://ru.code-basics.com/>
3. Домашняя страница Python www.python.org. Справочные материалы, официальная-документация
4. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>.
5. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>

Литература, рекомендованная обучающимся:

1. Классические задачи Computer Science на языке Python, Дэвид Копец – СПб.: Питер, 2022 – 224 с.;
2. Современные операционные системы, Таненбаум Эндрю, Бос Херберт – СПб.: Питер, 2022 – 1120 с.;
3. Python Быстрый старт, Джейми Чан, 352 стр. 2021 г. – СПб.: Питер, 2022 – 224 с.