

**МУНИЦИПАЛЬНОЕ АВТОНОМНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«ЛИЦЕЙ ИМЕНИ АКАДЕМИКА ИГОРЯ АЛЕКСЕЕВИЧА БАКУЛОВА»
ПОСЕЛКА ВОЛЬГИНСКИЙ
ПЕТУШИНСКОГО РАЙОНА ВЛАДИМИРСКОЙ ОБЛАСТИ**

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

Центра естественно-научной и технологической направленности

«Точка Роста»

«Программирование на Python»

Направленность: техническая

Уровень программы: базовый

Возраст учащихся: 12-17 лет

Срок реализации: 10 месяцев (72 часа)

**Автор составитель:
Раковская Татьяна Александровна
Педагог дополнительного образования
Высшей квалификационной категории**

п.Вольгинский, 2024 год

Нормативно-правовые документы, регламентирующие разработку и реализацию общеобразовательных общеразвивающих программ дополнительного образования:

1. Федеральный закон от 29.12.2012 № 273-ФЗ (ред. от 08.12.2020) «Об образовании в Российской Федерации» (с изм. и доп., вступ. в силу с 01.01.2021)
2. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. N 996-р «Стратегия развития воспитания в Российской Федерации на период до 2025 года»
3. Приказ Министерства просвещения Российской Федерации от 27 июля 2022 г. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»
4. Приказ Министерства просвещения Российской Федерации от 03.09.2019 № 467 «Об утверждении Целевой модели развития региональных систем дополнительного образования детей»;
5. Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»
6. Концепция развития дополнительного образования детей до 2030 года (Распоряжение Правительства РФ от 31 марта 2022 года № 06-1172)
7. Концепция духовно-нравственного развития и воспитания личности гражданина России (ФГОСООО)
8. Примерные требования к программам дополнительного образования детей в приложении к письму Департамента молодежной политики, воспитания и социальной поддержки детей Минобрнауки России от 11.12.2006 г. № 06-1844
9. Письмо Министерства образования и науки РФ N 09-3242 от 18 ноября 2015 г. «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)»
10. Положение о дополнительной общеобразовательной общеразвивающей программе педагога МАОУ «Лицей им. ак. И.А. Бакулова» пос. Вольгинский

Раздел 1. Комплекс основных характеристик программы

1. 1. Пояснительная записка

1. 1.1. Направленность программы – техническая.

1.1.2. Актуальность программы

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.

1.1.3. Значимость программы

Значимость 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 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.

1.1.4. Отличительные особенности программы.

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 «Программирование на Python» является практикоориентированной. 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.

Также в программе отдельный модуль выделен на развитие soft- компетенций обучающихся и обучение методикам командного взаимодействия, работы над проектами, поскольку данные навыки приобретают все большее значение в современном обществе, культуре и профессиональной среде.

1.1.5. Новизна программы

Новизна программы заключается в комплексном изучении предметов и дисциплин, не входящих в школьную программу, а также в том, что она построена на обучении в процессе практики, т.е. практикоориентирована. 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.

1.1.6. Адресат программы:

Адресат общеразвивающей программы дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» предназначена для детей в возрасте 12-17 лет, мотивированных к обучению и проявляющих интерес к IT-технологиям, приобретению навыков программирования.

1.1.7. Сроки реализации программы

Общее количество часов освоения программы составляет - 72 ч. Программа рассчитана на один год обучения.

Группа занимается два раза в неделю по одному часу.
Количество учебных часов в год – 72 ч.

1.1.8. Уровень программы: базовый

1.1.9. Особенности организации образовательного процесса:

Дополнительная общеобразовательная общеразвивающая программа базового уровня технической направленности «Программирование на Python» ориентирована на детей в возрасте от 12 до 17 лет. Отбор в группы – свободный. Численность учащихся в группе – 10-20 человек.

1.10. Форма обучения и режим занятий

Реализация программы осуществляется в очной форме. Занятия проходят 2 раза в неделю по 1 академическому часу. В год 72 часа.

1.1.11 Педагогическая целесообразность 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 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. Обучающиеся повышают уровень компьютерной грамотности и цифровой культуры, готовятся выступать на чемпионатах WorldSkills. Также полученные знания и навыки являются актуальными и востребованными при дальнейшем обучении подростков в СПО.

Осваивая данную программу, обучающиеся будут овладевать актуальными и 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 прикладной разработки на языке программирования Python для решения практических задач и разработки 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;
- обучить методикам проектной работы;
- 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

1.3.1. Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы организации	Формы аттестации
		Всего	Теория	Практика		
1	Знакомство с языком программирования Python	6	2	4	Групповая	Практическое задание
2	Вычисления в Python	6	2	4	Групповая	Практическое задание
3	Условные операторы в Python	6	2	4	Групповая	Практическое задание
4	Циклы в Python. Цикл for	6	2	4	Групповая	Практическое задание, Контрольные вопросы
5	Циклы в Python. Цикл while	6	2	4	Групповая	Практическое задание
6	Строки и списки. Функции в Python	6	2	4	Групповая	Практическое задание
7	Список и кортеж	6	2	4	Групповая	Практическое задание

8	Работа с файлами и строками	6	2	4	Групповая	Практическое задание, Контрольные вопросы
9	Сложные типы данных. Словари	6	2	4	Групповая	Практическое задание
10	Классы в Python	6	2	4	Групповая	Практическое задание
11	Работа с классами и функциями. Закрепление материала	6	2	4	Групповая	Практическое задание
12	Решение практических задач. Итоги курса. Презентация проектов	6	2	4	Групповая	Практическое задание. Контрольные вопросы. Защита проекта.
Итого: 72 ч.						

1.3.2. Содержание учебного плана

1. Знакомство с языком программирования Python (6 ч.)
 - Теория : (2ч)Что такое программирование;
 - Преимущества и простота программирования на Python;
 - Практика(4ч): Обзор IDE (Интегрированная среда разработки);
 - Простые переменные в Python
 - Практика написания первых команд.

Практическое задание: написание первых программ на языке Python (6 ч.).
Форма контроля Практическое задание
2. Вычисления в Python (6ч)
 - Теория (2ч)Типы данных в Python;
 - Арифметические и логические операции;
 - Практика (4ч)Преобразование типов данных;
 - Логические операторы.

Практическое задание: написания программ по решению математических задач на Python.
Форма контроля Практическое задание
3. Условные операторы в Python (6 ч.)
 - Теория (2ч)Что такое условный оператор;
 - Понятие блока кода;
 - Операторы if, else, elif;
 - Логические операторы and, or и not.

Практика(4ч): написать не менее 5 программ с использованием переменных, условных конструкций и логических операторов.
Форма контроля Практическое задание
4. Циклы в Python. Цикл for (6 ч.)
 - Теория (2Ч)Циклы в программировании. Цикл for;
 - Функция range;
 - Комментирование кода;
 - Библиотека turtle()
 - Вложенные циклы.

Практика (4ч): решение задач с применением циклов, функции range(),рисован не фигур с использованием библиотеки turtle.

Форма контроля Практическое задание

5. Циклы в Python. Цикл while (6 ч.)

- Теория (2ч) Знакомство с циклом while;
- Отличия конструкций циклов и сравнений;
- Прерывание цикла while;
- Практика программирования.

Практика (4ч): написание программ Python с использованием различных видов циклов и условных конструкций.

Форма контроля Практическое задание

6. Строки и списки. Функции в Python (6 ч.)

- Теория (2ч) Тип данных строка. Базовые операции со строками;
- Списки и операции над списками;
- Понятие "функции" в программировании;
- Примеры функций. Команда return.

Практика (4ч): написания программ с применением списков, строк, функций.

Создание программы "камень-ножницы-бумага".

Форма контроля Практическое задание

7. Список и кортеж (6 ч.)

- Теория (2ч) Концепция массива в программировании;
- Списки как массивы;
- Понятие кортежа. Операции над кортежами;
- Методы кортежа.

Практика (4ч): написания программ с применением различных видов списков.

Форма контроля Практическое задание

8. Работа с файлами и строками (6 ч.)

- Теория (2ч) Способы форматирования строк. f-строки;
- Чтение и запись в файл;
- Команды для r;
- Логические операторы and, or и not.

Практика (4ч): написать не менее 5 программ на работу со строками и действия с файлами.

Форма контроля Практическое задание

9. Сложные типы данных. Словари (6 ч.)

- Теория (2ч) Словари и множества в Python;
- Отличие списков и словарей;
- Методы работы со словарями;
- Практика программирования (4ч) - решение задач с использованием словарей.

Практика: поиск алгоритмов и написание кода для решения задач по теме "словари" и "списки".

- Форма контроля Практическое задание

10. Классы в Python (6 ч.)

- Теория (2ч) Введение в объектно-ориентированное программирование;
- Концепция классов. Методы классов;
- Атрибуты объектов;
- Практика (4ч): написание программ с использованием классов и библиотеки turtle(). Практическое задание: создание программ с использованием библиотеки turtle и применением классов.

- Форма контроля Практическое задание

11. Работа с классами и функциями. Закрепление материала (6 ч.)

- Теория (2ч) Повторение свойств class и def;
- Библиотеки в программировании;

- Основные этапы создания программы;
 - Поэтапное создание программы "Прыгающий мяч".
- Практика(4ч): написание программы "Прыгающий мяч" с использованием библиотеки turtle.

Форма контроля Практическое задание

12. Решение практических задач. Итоги курса (6 ч.)

- Теория (2ч) Знакомство с библиотеками tkinter и gttts;
- Практика (4ч) Подготовка к презентации созданных программ;
- Дальнейшее развитие и обучение в сфере Python-разработки;

Подведение итогов курса.

Практическое задание: подготовка выступления и презентация своего проекта.

Форма контроля Практическое задание

1.4 Планируемые результаты

	Планируемые результаты		
	Развивающие	Воспитательные	Обучающие

<p>Чего достигаем?</p>	<ul style="list-style-type: none"> • развиты навыки алгоритмического и критического мышления; • сформированы и развиты навыки работы с различными источниками информации, умения самостоятельно искать, извлекать и отбирать необходимую информацию; • развиты умения планирования, структурирования и разработки проектов, навыков организации и реализации проектной деятельности; • освоены методики проектной работы; • сформированы трудовые умения и навыки, умение планировать работу, предвидеть результат и достигать его; • ознакомлены с правилами индивидуального и коллективного безопасного поведения при работе с компьютерной техникой и оборудованием 	<ul style="list-style-type: none"> • сформированы навыки корректного поведения в обществе, социальных норм, ролей и понимания форм социального взаимодействия в группах • сформировано уважительное и продуктивное учебное сотрудничество и совместная деятельность со сверстниками в процессе проектной и учебно-исследовательской деятельности; • понимания необходимости организованного и ответственного отношения к учению, труду, другому человеку, его мнению и деятельности 	<ul style="list-style-type: none"> • Учащиеся познакомились с основными предметными понятиями программирования, компьютерных наук и их свойствами; • Познакомились с базовым синтаксисом и инструментарием языка программирования Python, необходимых для решения практических задач и разработки продуктов • сформированы навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python; • познакомить с базовыми конструкциями и принципами объектно-ориентированного программирования.
------------------------	--	--	--

Раздел № 2 Комплекс организационно-педагогических условий

2.1. Календарный учебный график.

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Количество часов	Тема занятия	Место проведения	Форма контроля
1	Сентябрь	04	14-50	Лекция	1	Что такое программирование. Преимущества и простота программирования на Python	Школьный кабинет	Входной, опрос, учебное тестирование
2	Сентябрь	06	15-00	Лекция	1	Обзор IDE (Интегрированная среда разработки). Простые переменные в Python	Школьный кабинет	Текущий, опрос, учебное тестирование
3	Сентябрь	11 13 18 20	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика написания первых команд	Школьный кабинет	Текущий, правильность выполнения практических заданий
4	Сентябрь	25	14-50	Лекция	1	Типы данных в Python. Арифметические и логические операции	Школьный кабинет	Текущий, опрос, учебное тестирование
5	Сентябрь	27	15-00	Лекция	1	Преобразование типов данных. Логические операторы	Школьный кабинет	Текущий, опрос, учебное тестирование
6	Октябрь	02 04 09 11	14-50 15-00 14-50 15-00	Практическое занятие	4	Практикум: «Вычисления в Python»	Школьный кабинет	Текущий, опрос, учебное тестирование

7	Октябрь	16	14-50	Лекция	1	Что такое условный оператор. Понятие блока кода	Школьный кабинет	Текущий, опрос, учебное тестирование
8	Октябрь	18	15-00	Лекция	1	Операторы if, else, elif. Логические операторы and, or и not.	Школьный кабинет	Текущий, опрос, учебное тестирование
9	Октябрь Ноябрь	23 25 30 01	14-50 15-00 14-50 15-00	Практическое занятие	4	Практикум: «Условные операторы в Python»	Школьный кабинет	Текущий, опрос, учебное тестирование
10	Ноябрь	06	14-50	Лекция	1	Циклы в программировании. Цикл for. Функция range.	Школьный кабинет	Текущий, опрос, учебное тестирование
11	Ноябрь	08	15-00	Лекция	1	Комментирование кода. Библиотека turtle(). Вложенные циклы.	Школьный кабинет	Текущий, опрос, учебное тестирование
12	Ноябрь	13 15 20 22	14-50 15-00 14-50 15-00	Практическое занятие	4	Практикум: «Циклы в Python. Цикл for»	Школьный кабинет	Текущий, опрос, учебное тестирование
13	Ноябрь	27	14-50	Лекция	1	Знакомство с циклом while.	Школьный кабинет	Текущий, опрос, учебное тестирование

14	ноябрь	29	15-00	Лекция	1	Отличия конструкций циклов и сравнений. Прерывание цикла while.	Школьный кабинет	Текущий, опрос, учебное тестирование
15	Декабрь	04 06 11 13	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование
16	Декабрь	18	14-50	Лекция	1	Тип данных строка. Базовые операции со строками. Списки и операции над списками.	Школьный кабинет	Текущий, опрос, учебное тестирование
17	Декабрь	20	15-00	Лекция	1	Понятие "функции" в программировании. Примеры функций. Команда return.	Школьный кабинет	Текущий, опрос, учебное тестирование
18	Декабрь Январь	25 27 15 17	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование
19	Январь	22	14-50	Лекция	1	Концепция массива в программировании. Списки как массивы	Школьный кабинет	Текущий, опрос, учебное тестирование
20	Январь	19	15-00	Лекция	1	Понятие кортежа. Операции над кортежами. Методы кортежа.	Школьный кабинет	Текущий, опрос, учебное тестирование
21	Январь Февраль	24 29 31 05	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование

22	Февраль	07	14-50	Лекция	1	Способы форматирования строк. f-строки. Чтение и запись в файл.	Школьный кабинет	Текущий, опрос, учебное тестирование
23	Февраль	12	15-00	Лекция	1	Команды для р. Логические операторы and, or и not.	Школьный кабинет	Текущий, опрос, учебное тестирование
24	Февраль	18 21 25 28	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование
25	Март	05	14-50	Лекция	1	Словари и множества в Python. Отличие списков и словарей	Школьный кабинет	Текущий, опрос, учебное тестирование
26	Март	07	15-00	Лекция	1	Методы работы со словарями.	Школьный кабинет	Текущий, опрос, учебное тестирование
27	Март	12 14 19 21	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование
28	Март	26	14-50	Лекция	1	Введение в объектно-ориентированное программирование. Концепция классов. Методы классов	Школьный кабинет	Текущий, опрос, учебное тестирование
29	Март	28	15-00	Лекция	1	Атрибуты объектов. Практика: написание программ с использованием классов и библиотеки turtle().	Школьный кабинет	Текущий, опрос, учебное тестирование

30	Апрель	02 04 09 11	14-50 15-00 14-50 15-00	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование
31	Апрель	16	14-50	Лекция	1	Повторение свойств class и def. Библиотеки в программировании.	Школьный кабинет	Текущий, опрос, учебное тестирование
32	Апрель	18	15-00	Лекция	1	Основные этапы создания программы. Поэтапное создание программы "Прыгающий мяч"	Школьный кабинет	Текущий, опрос, учебное тестирование
33	Апрель Май	23 25 07 14	14-50 15-00 15-00 14-50	Практическое занятие	4	Практика программирования	Школьный кабинет	Текущий, опрос, учебное тестирование
34	Май	16	14-50	Лекция	1	Знакомство с библиотеками tkinter и gttts	Школьный кабинет	Текущий, опрос, учебное тестирование
35	Май	21	15-00	Лекция	1	Подготовка к презентации созданных программ	Школьный кабинет	Текущий, опрос, учебное тестирование
36	Май	23 28	14-50 15-00	Практическое занятие	2	Решение практических задач. Итоги курса	Школьный кабинет	Текущий, опрос, учебное тестирование
37	Май июнь	30 04	14-50 15-00	Практическое занятие	2	Презентация своего проекта	Школьный кабинет	Текущий, опрос, учебное тестирование
Итого: 72 ч.								

2.2. Условия реализации программы:

2.2.1. Материально-техническое обеспечение

-помещение, соответствующий санитарно -гигиеническим нормам:

- 1) температурный режим в учебном кабинете для теоретических занятий - 20 - 22 °С;
- 2) оптимальным уровнем влажности в классе считается 55-62 % при температуре не более +21 °С, но не менее +19 °С., скорость движения воздуха не более 0,1 м/с.
- 3) помещения для занятий ежедневно проветриваются во время перерывов между занятиями, между сменами и в конце дня.
- 4) световой режим - для искусственного освещения предусматривается использование ламп по спектру цветоизлучения: белый, тепло-белый, естественно-белый.
- 5) После 30 - 45 минут теоретических занятий рекомендуется организовывать перерыв длительностью не менее 10 мин.
- 6) регламентирует минимальную диагональ электронных средств обучения: она должна составлять для монитора персонального компьютера и ноутбука – не менее 39,6 см, планшета – 26,6 см.
- 7) Показатель сопротивления изоляции токопроводимых элементов в норме составляет не менее 0,5 Ом, а для заземляющих устройств – не более 4 Ом.

Для занятий необходимо:

- компьютер (или планшет) - 15 шт,
- проектор - 1 шт,
- принтер- 1 шт
- интерактивная доска - 1 шт (продолжительность непрерывного использования на занятиях интерактивной доски для детей 8 - 13 лет составляет не более 20 минут)
- ПО ЯП Python (среда IDLE, PyCharm, Wing, VSC, Geany)
- ОС Альт Образование 10

2.2.2. Информационное обеспечение

1. Питонтьютор. Бесплатный курс по программированию с нуля. // [Электронный ресурс] URL: <https://pythontutor.ru/> (дата обращения: 14.04.2021);
2. Code Basics: обучение базовым аспектам языков программирования от образовательной платформы Hexlet. // [Электронный ресурс] URL: <https://ru.code-basics.com/> (дата обращения: 20.04.2021);

2.2.3. Кадровое обеспечение

Педагог дополнительного образования. Имеет курсы педагога дополнительного образования У педагога высшее профессиональное образование с учетом профиля образования.

2.3. Формы аттестации

Согласно учебному плану не предусмотрена входящая диагностика.

Текущий контроль:

Каждое занятие проводится опрос по теме занятия и тому, что ребята делают и на какой стадии они находятся. Это позволяет отслеживать понимание материала на данном этапе.

Промежуточный контроль: Самостоятельные работы и собственные проекты. Итоговый контроль проводится в конце изучения программы для оценки результатов освоения программы. Диагностика проводится в форме защиты проектов и мозгового штурма.

2.4. Оценочные материалы.

Текущий контроль складывается из следующих компонентов:

- в конце каждого занятия проводить опрос по пройденному материалу;
- в конце блока занятий проводится контроль проделанной работы методом презентации своих проектов;

2.5. Методические материалы

Методы обучения: словесный, наглядный, практический; объяснительно-иллюстративный; репродуктивный; частично-поисковый, исследовательский; проблемный, игровой, дискуссионный, проектный и др.; активные и интерактивные методы обучения; социогровые методы.

Методы воспитания: убеждение, поощрение, упражнение, беседа, пример, соревнования, практического задания.

При реализации программы используются следующие педагогические технологии:

Технология обучения в сотрудничестве (обучение в малых группах)

Информационные технологии: использование программных средств и компьютеров для работы с информацией.

Технология проблемного обучения.

Обучение развитию критического мышления.

Дифференцированное обучение

Игровые технологии.

2.6. Список литературы

2.6.1 Список литературы для обучающихся

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.

2.6.2 Список литературы для педагога

1. Архитектура компьютера, Таненбаум Эндрю, Остин Тодд - СПб.: Прогресс книга, 2022 - 816 с.;
2. Гид по Computer Science для каждого программиста, Вильям Спрингер - СПб.: Питер, 2020 - 193 с.;
3. Информатика, Тимофеева Е.В. М.: Эксмо, 2021 - 176 с.;
4. Python, например, Никола Лейси, - СПб.: Питер, 2021 - 192 с.;
5. Ли Воган. «Непрактичный» Python занимательные проекты для тех, кто хочет поумнеть.
- СПб.: БХВ-Петербург, 2021. - 457 с..

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) null2)

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')

4. 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

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

6. 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 i 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)