

МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«САДОВСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА»
НИЖНЕГОРСКОГО РАЙОНА РЕСПУБЛИКИ КРЫМ

| | | |
|---|--|---|
| РАССМОТРЕНО На заседании методического объединения учителей _____ Протокол № __ от « ____ » _____ 2021 г. №- _____ Руководитель МО _____ (подпись) _____ (ФИО) | СОГЛАСОВАНО Заместитель директора школы И.Ф.Петренко _____ « ____ » _____ 2021 г. | УТВЕРЖДЕНО Приказом директора МБОУ «Садовская СОШ» О. В. Луцык _____ № __ от « ____ » _____ 2021г. |
|---|--|---|

**Дополнительная общеобразовательная программа
(дополнительная общеразвивающая программа)
«Программирование на Python»**

Направленность: **техническая**
Возраст обучающихся: **12 – 16 лет (7-11 класс)**
Срок реализации: **1 год**
Вид программы: **модифицированная**
Уровень: **базовый**

Составитель: **педагог дополнительного образования
Киселев А.А.**

с. Садовое
2021 г.

I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа дополнительной общеобразовательной программы «Программирование на языке Python» с использованием оборудования центра «Точка роста» определяет объем содержания образования, планируемые результаты освоения, распределение учебных часов по учебным темам.

Рабочая программа разработана в соответствии с федеральным государственным образовательным стандартом *основного и среднего* общего образования, основной образовательной программой *основного и среднего* общего образования, учебным планом.

Направленность программы: техническая.

Актуальность: Программирование на языке Python является одной из самых востребованных компетенций в современном мире. Эта программа нацелена на изучение основ программирования на языке Python, основных приёмов написания программ на современном языке программирования, развитие алгоритмического мышления учащихся, творческих способностей, аналитических и логических компетенций.

Цель программы: приобретение обучающимися устойчивых навыков программирования на языке программирования Python.

Отличительные особенности данной программы:

Модульность

В связи с тем, что объем информации, связанной с информационными технологиями, огромен, существует проблема выбора предмета изучения и последовательности изучения различных направлений ИТ. В данной программе эта проблема решается за счет того, что учащиеся на каждом году обучения получают законченный объем знаний определенного уровня по различным направлениям (модулям) программирования. В следующем году происходит углубление и расширение знаний по каждому из модулей. Каждый год обучения состоит из четырех модулей, которые можно условно назвать: "Языки программирования", "Синтаксис Python", "Принципы ООП Python", "Прикладное программирование на Python". Так, например, в "Языках программирования" учащиеся на первом году обучения осваивают основные алгоритмические конструкции языков программирования на примере Python, чтобы научиться писать простые программы, а на втором году — учатся работать в команде, составляя одну большую программу из кусочков, написанных разными детьми.

Адресат программы: программа рассчитана на учащихся с 12 до 17 лет.

Формы обучения и виды занятий: занятия проводятся в группах, продолжительность занятия 45 минут,

Интернет-ресурсы для учащихся

1. <https://pythonworld.ru/samouchitel-python>
Учебник по языку программирования Python
2. <https://habr.com/ru/post/61905/Python/>
Учебник Python 3.1
3. <https://pythonru.com/uroki/python-dlja-nachinajushhih>
Python для начинающих 2021

II. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Объём программы: общая продолжительность программы 36 часов

Срок освоения программы: 1 год

Режим занятий: 1 час в неделю.

Программа рассчитана на 1 учебный год, в течение которого 1 раз в неделю проходит занятие (45 мин.). Занятия могут быть теоретической или практической направленности. На теоретическом занятии учащихся знакомят с правилами и особенностями языка программирования Python3. На практических занятиях учащиеся применяют полученные теоретические знания для решения конкретных задач. Учащиеся проходят промежуточную аттестацию (декабрь) и итоговую аттестацию (май).

Основная форма занятий – групповая. Но также может использоваться индивидуальная форма работы с занимающимися, испытывающими трудности в освоении программы.

Формами занятий являются: учебное теоретическое занятие, урок-зачет, урок решения задач на компьютере.

План работы рассчитан на 36 недель.

Условия приема детей: в кружок принимаются все желающие.

III. ЦЕЛИ И ЗАДАЧИ ПРОГРАММЫ

Планируемые результаты освоения

Личностные:

- ✓ формирование умения самостоятельной деятельности;
- ✓ формирование умения работать в команде;
- ✓ формирование коммуникативных навыков;
- ✓ формирование навыков анализа и самоанализа;
- ✓ формирование целеустремлённости и усидчивости в процессе творческой, исследовательской работы и учебной деятельности.

Предметные:

- ✓ формирование понятий «алгоритм», «программа»;
- ✓ формирование понятий об основных конструкциях языка программирования Python, таких как оператор ветвления if, операторы цикла while, for,
- ✓ вспомогательные алгоритмы;
- ✓ формирование понятий о структурах данных языка программирования Python
- ✓ формирование основных приёмов составления программ на языке программирования Python;
- ✓ формирование алгоритмического и логического стилей мышления.

Метапредметные:

- ✓ формирование умения ориентироваться в системе знаний;

- ✓ формирование умения выбирать наиболее эффективные способы решения задач на компьютере в
- ✓ зависимости от конкретных условий;
- ✓ формирование приёмов 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, классифицировать, наблюдать, проводить
- ✓ эксперименты, делать выводы и заключения, доказывать, защищать свои идеи, оценивать
- ✓ результаты своей работы;
- ✓ формирование умения распределять время;
- ✓ формирование умений успешной самопрезентации.

Обучающие задачи программы:

- формирование представления о роли информационных технологий в современном обществе;
- знакомство с возможностями компьютерных технологий в отношении обработки и представления графической и текстовой информации посредством написания программ;
- формирование навыков работы с современным свободным программным обеспечением (СПО);
- изучение различных парадигм языка программирования Python 3.x, инструментов для создания графического интерфейса пользователя;
- формирование представления о мире как системе разнообразных взаимодействующих объектов;
- формирование представления об игровой стратегии, формирование навыков развития сюжетных линий, навыков прогнозирования игрового поведения.

Развивающие задачи программы:

- развитие абстрактного и логического мышления;
- развитие творческого подхода к решению различных задач.

Воспитательные задачи программы:

- воспитание самостоятельности;
- воспитание культуры взаимодействия с другими людьми в условиях открытого информационного общества.

IV. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ

В конце обучения учащийся должен иметь следующие личностные результаты:

- представление о современном языке программирования высокого уровня Python 3.x;
- синтаксис языка Python: основные инструкции языка программирования Python, списки, словари, строки, кортежи, генераторы функций; понимание блок-схем; представление о реализации анимации с помощью языка программирования;
- объекты, свойства и методы объектов; принцип построения программ «снизу-вверх» и «сверху-вниз»;
- решение простых прикладных задач;

метапредметные результаты:

- работа в любой среде разработчика, поддерживающей Python 3.x;
- подготовка программы к запуску;
- составление программ на языке программирования Python 3.x;
- создание анимированных изображений с помощью Python 3.x;
- работа в операционной системе на уровне пользователя;
- набор и редактирование текста на английском языке;
- создание простых приложений.

Регулятивные УУД:

- *определять и формулировать* цель деятельности на занятии с помощью учителя, а далее самостоятельно;
- *проговаривать* последовательность действий;
- уметь *высказывать* своё предположение (версию) на основе данного задания, уметь *работать* по предложенному учителем плану, а в дальнейшем уметь самостоятельно планировать свою деятельность;
- средством формирования этих действий служит технология проблемного диалога на этапе изучения нового материала;
- учиться совместно с учителем и другими воспитанниками *давать* эмоциональную *оценку* деятельности команды на занятии.

Средством формирования этих действий служит технология оценивания образовательных достижений (учебных успехов).

Познавательные УУД:

- добывать новые знания: *находить ответы* на вопросы, используя разные источники информации, свой жизненный опыт и информацию, полученную на занятии;
- перерабатывать полученную информацию: *делать выводы* в результате совместной работы всей команды;

Средством формирования этих действий служит учебный материал и задания.

Коммуникативные УУД:

- умение донести свою позицию до других: оформлять свою мысль. *Слушать* и *понимать* речь других;
- совместно договариваться о правилах общения и поведения в игре и следовать им;

- учиться выполнять различные роли в группе (лидера, исполнителя, критика).

Средством формирования этих действий служит организация работы в парах и малых группах.

Оздоровительные результаты программы внеурочной деятельности:

- осознание учащимися необходимости заботы о своём здоровье и выработки форм поведения, которые помогут избежать опасности для жизни и здоровья, уменьшить пропуски занятий по причине болезни, регулярно посещать спортивные секции и спортивно-оздоровительные мероприятия;
- социальная адаптация детей, расширение сферы общения, приобретение опыта взаимодействия с окружающим миром.

Примерные темы мини-проектов

1. Конвертор чисел (перевод числа в n-ичную систему счисления).
2. Шифровальщик текста (реализация шифра Цезаря).
3. Компьютерный тест.
4. Реализация игры «Камень, ножницы, бумага».
5. Калькулятор для ипотеки.

V. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

Основной учебной базой для проведения занятий является кабинет информатики 307

Инвентарь:

Ноутбуки – 3 штуки

Компьютеры – 12 штук

Мебель:

Парты – 6 штук на 12 посадочных мест

Стулья – 12 штук

Компьютерные столы – 12 штук на 12 посадочных мест

Кресла – 12 штук.

Методическое обеспечение программы.

| №\п | раздел | Формы занятий | Приемы и методы организации занятий | Дидактический материал | Формы подведения итогов |
|-----|---------------------------|---------------------------|--------------------------------------|--|-------------------------|
| 1. | Основы знаний | Учебное групповое занятие | Словесные, репродуктивные, наглядные | Проверочные задания в виде листинга фрагментов программ | Тест |
| 2. | Практика по решению задач | Учебное групповое занятие | Словесные, наглядные, практические. | Распечатки условий задач для решения на компьютере | Решение задачи |
| 3. | Промежуточная аттестация | Учебное групповое занятие | Словесные, наглядные, практические | Бумажный или электронный тест, распечатка условия задачи | Тест + задача |
| 4. | Итоговая аттестация | Учебное групповое занятие | Словесные, наглядные, практические | Бумажный или электронный тест, распечатка условия задачи | Тест + задача |

Контрольно-измерительные материалы КИМ для промежуточной аттестации Тест «Язык Python3»

Задание: Продолжите каждое предложение по смыслу. За каждый правильный ответ вы набираете 1 балл.

1. Программа на Python называется ...
2. Расширение файла Python – ...
3. Переменная в Python – это ...
4. Регистр букв в идентификаторах значение ...
5. Выражение в Python – это ...
6. Символ # в Python обозначает ...
7. ... в Python это тип данных для вещественных чисел, встроенный в Python по умолчанию.
8. Операция $3 ** 4$ – это
9. 345 – данные ... типа.
10. Операция $46 \% 10$ – это ...
11. Функция `round(d)` – это ...
12. Функция `input()` – предназначена для ...

13. Для вывода данных в Python есть функция - ...
14. ... в Python - это логический тип данных, встроенный в Python по умолчанию.
15. Строки – это ...
16. A = 'pri', s = 'vet'. A + s – это ...
17. E = 'no', print(E * 5) выведет на экран ...
18. К элементу в строке можно обратиться по ...
19. S = 'asdfgh'
print(s[-1]). Программа выведет ...
20. S = 'asdfgh'
print(s[2:4]). Программа выведет ...
21. Функция len(S) – возвращает ...
22. Списки – это ...
23. Пример списка - ...
24. Словари – это ...
25. Пример словаря - ...
26. Условный оператор в Python - ...
27. Цикл for называется циклом ...
28. Переведите конструкцию языка
S = [1, 2, 3]
for i in S:
 print(i * 4)
29. Функция range() переводится как ...
30. Переведите конструкцию языка
S = 0
while S < 10:
 print(S)
 S += 1

Ответы:

1. Скрипт
2. Py
3. имя/идентификатор, который может принимать некоторое значение.
4. Имеет
5. это фрагмент языка программирования, представляющий способ вычисления некоторого значения.
6. Комментарий
7. Float
8. Возведение в степень
9. Целочисленный, int
10. Остаток от деления
11. Округление числа
12. Ввода данных в строку
13. Print()
14. Bool
15. Упорядоченные неизменяемые последовательности символов, используемые для хранения и представления текстовой информации
16. Объединение, сложение строк. Конкатенация
17. Повторение строки 5 раз. Дублирование
18. Индексу
19. h

20. dfg
21. длину строки
22. изменяемая последовательность произвольных объектов.
23. C=[2,3,4.5, 'gh']
24. Изменяемые неупорядоченные коллекции произвольных объектов с доступом по ключу
25. K={1:'a',2:'b',3:'c'}
26. If
27. Обхода
28. для каждого элемента I в списке s делать следующее (то, что в теле цикла)
29. диапазон
30. пока условие истинно, то выполняется инструкция, после чего условие проверяется снова и снова выполняется инструкция. Так продолжается до тех пор, пока условие будет истинно, в противном случае мы выйдем из цикла.

Критерии оценивания:

| № п/п | Количество набранных баллов | Оценка |
|--------------|------------------------------------|-----------------|
| 1 | 27-30 | Высокий уровень |
| 2 | 20-26 | Базовый уровень |
| 3 | до 19 | Низкий уровень |

Задача на промежуточную аттестацию

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Ленты облаков, расшитые серебряными нитями, стягивались к Краедуге, где, увлекаемые вращением мира, свертывались в стомильные бигуди. Два помела, взрыхляя туман, оставляли за собой клубящиеся туннели пара, так что наблюдающие за этим полетом боги — которые наверняка наблюдали, поскольку им просто нечем больше заниматься — явились свидетелями грандиозного окучивания неба.

Поднявшись на тысячу футов и войдя в ледяную полосу атмосферы, ведьмы вновь принялись спорить. Когда спорят ведьмы, слова разлетаются фейерверком. Даже богам опасно вставать у них на пути. Слово ведьмы, да еще сказанное в горячке спора, может обладать броневой силой.

Напишите программу, выбирает слова по определенному правилу.

Формат ввода

Строка слов, разделенных символами `<&>`.

Строка слов, разделенных символами `-{-}`.

Формат вывода

Для каждого слова из первой строки нужно записать список слов из второй строки, в которых есть ровно 2 общие буквы с заданным. Одинаковые буквы в слове считаются за одну. Порядок вывода строк и слов в списке не важен. Слова в списке записываются через запятую и пробел. Если таких слов не оказалось, вывести **нет слов**.

Пример 1

| Ввод | Вывод |
|---|---|
| туман<&>метла<&>слово земля-{-}-камни-{-}-твердый-{-}-мягкий-{-}-удар-{-}-черт | туман: удар метла: камни, твердый, черт слово: нет слов |

Пример 2

| Ввод | Вывод |
|--|--|
| пейзаж<&>красота<&>ландшафт сон-{-}-ярость-{-}-суровый-{-}-пик-{-}-долина | пейзаж: нет слов красота: сон, долина ландшафт: нет слов |

КИМ для итоговой аттестации

Тест «Язык Python3»

Задание: Продолжите каждое предложение по смыслу. За каждый правильный ответ вы набираете 1 балл.

1. Программа на Python называется ...
2. Расширение файла Python – ...
3. Переменная в Python – это ...
4. Регистр букв в идентификаторах значение ...
5. Выражение в Python – это ...
6. Символ # в Python обозначает ...
7. ... в Python это тип данных для вещественных чисел, встроенный в Python по умолчанию.
8. Операция $3 ** 4$ – это
9. 345 – данные ... типа.
10. Операция $46 \% 10$ – это ...
11. Функция `round(d)` – это ...
12. Функция `input()` – предназначена для ...
13. Для вывода данных в Python есть функция - ...
14. ... в Python - это логический тип данных, встроенный в Python по умолчанию.
15. Строки – это ...
16. `A = 'pri', s = 'vet'. A + s` – это ...
17. `E = 'no', print(E * 5)` выведет на экран ...
18. К элементу в строке можно обратиться по ...
19. `S = 'asdfgh'`
`print(s[-1])`. Программа выведет ...
20. `S = 'asdfgh'`
`print(s[2:4])`. Программа выведет ...
21. Функция `len(S)` – возвращает ...
22. Область видимости функции – это ...
23. Глобальная переменная - это ...
24. Словари – это ...
25. Пример словаря - ...
26. Условный оператор в Python - ...
27. Цикл `for` называется циклом ...
28. Переведите конструкцию языка
`S = {1, 2, 3}`
`for i not in S:`
`S.add(i)`
29. Функция `round()` переводится как ...
30. Переведите конструкцию языка
`S = { }`
`for x in input().split():`
`s[x[0]] = x[1]`

Ответы:

1. Скрипт
2. Py
3. имя/идентификатор, который может принимать некоторое значение.
4. Имеет
5. это фрагмент языка программирования, представляющий способ вычисления некоторого значения.
6. Комментарий
7. Float
8. Возведение в степень
9. Целочисленный, int
10. Остаток от деления
11. Округление числа
12. Ввода данных в строку
13. Print()
14. Bool
15. Упорядоченные неизменяемые последовательности символов, используемые для хранения и представления текстовой информации
16. Объединение, сложение строк. Конкатенация
17. Повторение строки 5 раз. Дублирование
18. Индексу
19. h
20. dfg
21. длину строки
22. рамки подпрограммы или программы, где работает та или иная переменная
23. переменная, видимая из всех частей программы
24. Изменяемые неупорядоченные коллекции произвольных объектов с доступом по ключу
25. $K = \{1:'a', 2:'b', 3:'c'\}$
26. If
27. Обхода
28. для элемента, не входящего в множество S, добавить его в множество S
29. математическое округление
30. для элемента, введенного клавиатуры и разделенного на слова по пробелу, добавить в словарь по ключу – первому слову его значение – второе слово

Критерии оценивания:

| № п/п | Количество набранных баллов | Оценка |
|-------|-----------------------------|-----------------|
| 1 | 27-30 | Высокий уровень |
| 2 | 20-26 | Базовый уровень |
| 3 | до 19 | Низкий уровень |

Задача на итоговую аттестацию

Ограничение времени 1 секунда

Ограничение памяти 64Mb

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Напишите функцию `export_check(text)`, которая принимает на вход текст, где каждая строка состоит из:

1. **названия товара (строка)**
2. **цены за единицу товара (вещественное число)**
3. **количества единиц товара (целое число)**

(Значения внутри строки разделены символом табуляции).

Ваша программа должна обработать полученные данные и создать документ Excel `res.xlsx`, в котором будут следующие столбцы:

«Товар», «Цена за единицу товара», «Количество товара», «Общая стоимость товара» (названия столбцов выводить не надо)

В столбце «Общая стоимость товара» должна содержаться формула, которая рассчитывает общую стоимость товара как произведение цены и количества товара (с использованием простого перемножения ячеек)

Последней строкой должно идти «Итого», с пустыми значениями в столбцах «Цена за единицу товара» и «Количество товара».

В столбце «Общая стоимость товара» должна содержаться формула, которая считает общую сумму покупок. (с использованием функции Excel СУММ)

Для разделения товаров разных чеков используется строка «---». В документе `res.xlsx` информация о каждом чеке должна располагаться на отдельном листе. При дублировании товара (совпадении названия и цены за единицу) в чеке в `res.xlsx` должна создаваться только одна строка с общим количеством купленного товара.

Товары в чеке должны быть отсортированы по алфавиту.

Формат ввода

Многострочный текст в параметре `text`.

Формат вывода

Файл `res.xlsx`, содержащий обработанную информацию.

VI. СОДЕРЖАНИЕ ПРОГРАММЫ
УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

| Модуль | № темы | Название темы | Всего часов | Теория | Практика |
|--------|--------|---|-------------|--------|----------|
| I | 1. | Языки программирования. Обзор современных языков программирования. | 1 | 1 | 0 |
| | 2. | Язык программирования Python 3.x. Особенности ссылочных переменных. Блок-схемы. Команды ввода-вывода. Ветвления. Циклы. | 3 | 1 | 2 |
| II | 3. | Разные способы ввода-вывода информации: генераторы. Решение задач. | 2 | 1 | 1 |
| | 4. | Ветвления: отличия. Решение задач. | 2 | 1 | 1 |
| | 5. | Циклы: виды, особенности, генераторы условий. Решение задач. | 2 | 1 | 1 |
| | 6. | Типы данных: целый тип, вещественный тип. | 2 | 1 | 1 |
| | 7. | Строки. Списки. Решение задач. | 2 | 1 | 1 |
| | 8. | Кортежи. Срезы. Решение задач. | 2 | 1 | 1 |
| | 9. | Словари. Решение задач. Промежуточная аттестация. | 3 | 1 | 2 |
| | 10. | Функции. Работа с функциями в Python. Решение задач. | 2 | 1 | 1 |
| | 11. | Работа со встроенным графическим исполнителем Черепашка. | 2 | 1 | 1 |
| | 12. | Работа с подключенным из модуля исполнителем Робот. | 2 | 1 | 1 |
| III | 13. | Объекты. Теория объектов. Решение задач. | 3 | 1 | 2 |
| | 14. | Классы. Наследование свойств. Решение задач. | 3 | 1 | 2 |
| IV | 15. | Изучаем более совершенную графику с модулем Tkinter. Итоговая аттестация. | 6 | 1 | 5 |
| | 16. | Резерв | 1 | 0 | 1 |
| Итого: | | | 36 | 15 | 23 |

СОДЕРЖАНИЕ

1. Языки программирования. Основные понятия и конструкции языков программирования на примере Python (4).

1. Языки программирования. Обзор современных языков программирования.
2. Язык программирования Python 3.x. Особенности ссылочных переменных. Блок-схемы. Команды ввода-вывода. Ветвления. Циклы.

2. Синтаксис Python. Работа со стандартной библиотекой и с внешними библиотеками. Основы объектно-ориентированного программирования в Python (21).

1. Разные способы ввода-вывода информации: генераторы. Решение задач.
2. Ветвления: отличия. Решение задач.
3. Циклы: виды, особенности, генераторы условий. Решение задач.
4. Типы данных: целый тип, вещественный тип.
5. Строки. Списки. Решение задач.
6. Кортежи. Срезы. Решение задач.
7. Словари. Решение задач.
8. Промежуточная аттестация.
9. Функции. Работа с функциями в Python. Решение задач.
10. Работа со встроенным графическим исполнителем Черепашка. Подключение модуля. Работа с библиотекой модуля.
11. Работа с созданным исполнителем Робот.
12. Создание исполнителя Чертежник. Работа с Чертежником.
13. Создание собственного исполнителя. Проект.

3. Изучение принципов ООП Python. Технология разработки программ. Графические интерфейсы (6).

1. Объекты. Теория объектов. Решение задач.
2. Классы. Наследование свойств. Решение задач. Оформление отдельных абзацев и символов.

4. Решение прикладных задач и создание приложений с графическим интерфейсом и использованием различных библиотек Python. Профессиональная ориентация (7).

1. Изучаем более совершенную графику с модулем Tkinter.
2. Итоговая аттестация.
3. Резерв.

Форма проведения аттестации

1. Промежуточная аттестация - тест на знание теории и практическая задача.
2. Итоговая аттестация – тест на знание теории и практическая задача.

Критерии оценивания результатов

1. Тест на знание теории проводится на 10 ключевых вопросах, правильный ответ на которые оценивается 1 первичным баллом.
2. Практическая задача проверяется на компьютерных тестах. В зависимости от количества пройденных тестов к результату теста по теории добавляется от 0 (нет пройденных тестов) до 5 (все тесты пройдены) баллов.
3. Итоговый первичный балл, набранный по сумме баллов тестовой и практической части переводится в проценты делением на 15.
4. При наборе от 70% и более ученик показывает высокий уровень подготовки по дисциплине.
5. При наборе от 50% до 69% уровень полученных знаний и умений считается базовым.
6. При наборе ниже 50% уровень подготовки ученика считается низким.

Модули:

I. Языки программирования. Основные понятия и конструкции языков программирования на примере Python.

II. Синтаксис Python. Работа со стандартной библиотекой и с внешними библиотеками. Основы объектно-ориентированного программирования в Python.

III. Изучение принципов ООП Python. Технология разработки программ. Графические интерфейсы.

IV. Решение прикладных задач и создание приложений с графическим интерфейсом и использованием различных библиотек Python. Профессиональная ориентация.

Несмотря на то, что программа состоит из различных модулей, все они основаны на обучении программированию. Поэтому программа способствует ранней профессиональной ориентации обучающихся.

- *Использование свободного программного обеспечения (СПО)*

Использование СПО позволяет гарантировать равные возможности участникам образовательного процесса, несет в себе воспитательное значение как демонстрация положительного результата открытости и взаимодействия профессионального сообщества.

- *Возможность самостоятельного изучения курса программы, использование элементов дистанционного обучения (в разработке)*

С будущим целью методической поддержки программы будет создан сайт (<http://>), на котором в открытом доступе выложу дидактические материалы к занятиям. Обучающиеся смогут воспользоваться этим материалом для самообучения, задать вопросы по изучаемой теме, получить консультацию.

VII. СПИСОК ЛИТЕРАТУРЫ, ИНТЕРНЕТ-РЕСУРСЫ

1. <https://pythontutor.ru/>
2. <https://www.python.org/>
3. <https://pythonworld.ru/samouchitel-python>
4. <https://pythoner.name/>