


Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего
образования
«Национальный исследовательский Томский государственный университет»



СЕРТИФИЦИРУЮ
Проректор по образовательной
деятельности


Е.В. Луков
(подпись)

«23» июля 2023 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Разработчик на Python: с нуля до junior»

технической направленности

Уровень сложности: «Начальный»

144 часа

СОГЛАСОВАНО:

Директор ЦРСК

Заместитель директора ИДО




М.В. Назарова

В.С. Дубровская

г. Томск - 2023




Пояснительная записка

Актуальность программы	Актуальность программы обусловлена необходимостью профориентации школьников в области информационных технологий, а также повышения их мотивации к обучению современным языкам программирования. Содержание программы способствует формированию необходимых компетенций для применения информационных технологий в России. А получение практических навыков работы с библиотеками и синтаксисом языка Python позволит школьникам легко адаптироваться в современном информационном обществе и использовать полученные знания в дальнейшей учебной и профессиональной деятельности.
Цель программы	Изучение языка программирования Python с целью дальнейшего профессионального самоопределения, развитие инженерного мышления, воспитание конкурентоспособной личности. Достижение цели происходит за счет решения следующих задач: получить практические навыки работы синтаксисом языка Python, библиотеками, а также методами программирования в среде Python для решения типовых задач.
Планируемые результаты обучения (знания, умения, навыки)	В результате обучения выпускник программы будет уметь использовать базовые типы данных, арифметические операции, условные логические конструкции, итерационные и условные циклы, контейнеры, методы и функции, классы и декораторы на языке программирования Python, а также способен работать с файлами и базовыми модулями и применять основы объектно-ориентированного программирования.
Категория обучающихся по программе	Учащиеся 8 класса, Учащиеся 9 класса, Учащиеся 10 класса, Учащиеся 11 класса Обучающиеся по программам среднего профессионального образования
Срок освоения программы	2 года
Формы и режим занятий	Очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения
Форма обучения	Очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения
Трудоемкость программы	144 ак.ч.
Примечание	Начальный уровень

**Общие данные о Дополнительной общеобразовательной программе
«Разработчик на Python: с нуля до junior»**

Об организации

Наименование поля	Допустимые значения поля	Значение поля
ИНН организации, осуществляющей образовательную деятельность	10 арабских цифр	7018012970
Наименование организации	строка	Федеральное государственное автономное учреждение высшего образования «Национальный исследовательский Томский государственный университет»
Логотип организации	изображение в формате jpeg разрешением не менее 100x100 пиксель	 <p>Национальный исследовательский Томский государственный университет</p>
Ссылка на логотип организации	URL на изображение, находящееся в сети интернет	https://drive.google.com/file/d/170J9BfWu1szAoJPBAKlaWFJgwtJiqYT/view?usp=share_link
Контакты ответственного за программу (с указанием фамилии, имени, отчества).	строка от 5 до 255 символов	Шарыпина Полина Андреевна
Контакты ответственного за программу. Должность	строка от 5 до 255 символов	Заместитель директора Центра совместных образовательных программ
Контакты ответственного за программу. Телефон	Формат +7(XXX)XXXXXXXX	+7 (952) 895 7687
Контакты ответственного за программу. E-mail	строка	sharypinarina@gmail.com

Информация о программе

Наименование поля	Допустимые значения поля	Значение поля (примеры)
Название программы (курса)	строка	Разработчик на Python: с нуля до junior
Описание программы	строка не менее 1000 не более 5000 символов	<p>Данная программа предусмотрена для учащихся школ 8-11 классов и обучающихся по программам среднего профессионального образования без навыков по программированию на языке Python. Обучаться по программе могут школьники и обучающиеся по программам среднего профессионального образования, успешно сдавшие вступительное испытание. Изучение языка начинается со знакомства с возможностями, синтаксисом языка, технологией и методами программирования в среде Python. В процессе освоения образовательной программы обучающийся получит опыт применения шаблонов проектирования на языке Python, работы с Python библиотеками, использования объектно-ориентированного и функционального программирования, а также обучающийся научится выбирать тот или иной алгоритм для поставленной задачи, искать и обрабатывать ошибки в коде, разбивать решение задачи на подзадачи, использовать математические и логические способности, грамотно и эффективно презентовать проект, выстраивать взаимодействия в команде при работе над общей задачей, принимать решения, реагировать на изменение условий работы, распределять ресурсы и управлять своим временем. Содержание программы способствует повышению интереса школьников к программированию на языке Python, формированию самостоятельности и творческого подхода к решению задач с помощью средств современной вычислительной техники.</p> <p>Задачи программы: Обучающие:</p>

<p>Аннотация (для размещения на маркетплейсе, понятное и привлекательное для Потенциальных покупателей поддержки, включающее полное и содержательное описание Дополнительной общеобразовательной программы:</p> <ol style="list-style-type: none"> 1) краткое описание Дополнительной общеобразовательной программы; 2) описание требований и рекомендаций для обучения по Дополнительной общеобразовательной программе; 3) краткое описание результатов обучения в свободной форме, включая описание практикоориентированного характера Дополнительной общеобразовательной программы) 	<p>строка до 1000 символов</p>	<ul style="list-style-type: none"> • обучить языку программирования Python и созданию программ на его основе; • научить создавать прикладное программное обеспечение; • расширять кругозор обучающихся в области программирования. <p>Развивающие:</p> <ul style="list-style-type: none"> • развивать память и внимание, познавательную и творческую активность; • развивать логическое мышление. <p>Воспитательные:</p> <ul style="list-style-type: none"> • прививать интерес к активному творческому самовыражению, культуре труда; • воспитывать упорство в достижении желаемого результата; • воспитывать чувство взаимопомощи, доверия, коллективизма. <p>Программа «Разработчик на Python: с нуля до junior» направлена на изучение языка программирования Python с целью дальнейшего профессионального самоопределения, развитие инженерного мышления, воспитание конкурентоспособной личности. Программа включает в себя 4 модуля, направленных на поэтапное усвоение языка программирования в формате «от простого к сложному». Школьники не только изучат язык программирования Python, но и различные библиотеки, чтобы обрабатывать свои парсеры и написать простейшую графическую игру.</p> <p>Обучаться по программе могут школьники, обучающиеся с 8 по 11 классы и обучающиеся по программам среднего профессионального образования, являющиеся гражданами Российской Федерации и успешно сдавшие вступительное испытание.</p> <p>В результате обучения выпускник программы будет уметь использовать базовые типы данных, арифметические операции, условные логические конструкции, итерационные и условные циклы, контейнеры, методы и функции, классы и декораторы на языке программирования Python, а также способен работать с</p>
---	--------------------------------	--

<p>файлами и базовыми модулями и применять основы объектно-ориентированного программирования.</p>		
<p>Цель программы - изучение языка программирования Python с целью дальнейшего профессионального самоопределения, развитие инженерного мышления, воспитание конкурентоспособной личности. Достижение цели происходит за счет решения следующих задач: получить практические навыки работы синтаксисом языка Python, библиотеками, а также методами программирования в среде Python для решения типовых задач.</p>	<p>строка не менее 100 символов</p>	<p>Цель программы</p>
<p>Актуальность программы обусловлена необходимостью профориентации школьников в области информационных технологий, а также повышения их мотивации к обучению современным языкам программирования. Содержание программы способствует формированию необходимых компетенций для применения информационных технологий в России. А получение практических навыков работы с библиотеками и синтаксисом языка Python позволит школьникам легко адаптироваться в современном информационном обществе и использовать полученные знания в дальнейшей учебной и профессиональной деятельности.</p>	<p>строка не менее 500 символов</p>	<p>Актуальность</p>

Дополнительная информация	строка	<p>Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Учащиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, углубляя их, учася составлять простые и сложные программы.</p>
Формат обучения	<p>значение из: очная форма без применения дистанционных образовательных технологий; очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения</p>	<p>очная форма с применением дистанционных образовательных технологий, в том числе, с применением средств электронного обучения</p>
Уровень сложности	<p>значение из: «Начальный» «Базовый» «Продвинутый»</p>	<p>«Начальный»</p>
Срок освоения образовательной программы	<p>строка, значение в ак.ч.</p>	<p>144 ак.ч.</p>
Объем каждого модуля в ак.ч.	<p>целое число</p>	<p>36 ак.ч.</p>

Объем часов в неделю в ак. ч.	целое число	4 ак. ч.
Количество занятий	целое число	100
Направленность программы	строка	Современные языки программирования
Язык программирования	строка	Python
Дополнительная общеобразовательная программа не представлена для участия в иных федеральных проектах, направленных на дополнительное образование граждан, кроме федерального проекта «Развитие кадрового потенциала ИТ-отрасли»	строка, значения: «Не представлена»	«Не представлена»
Дополнительная общеобразовательная программа не была реализована до начала отбора и/или не реализовывается в период отбора на безвозмездной основе	строка, значения «Не реализована»	«Не реализована»
Категория обучающихся по программе	строка не менее 10 символов	Учащиеся 8 класса, Учащиеся 9 класса, Учащиеся 10 класса, Учащиеся 11 класса Обучающиеся по программам среднего профессионального образования
Описание планируемых результатов обучения	строка не менее 10 символов	В результате обучения выпускник программы будет уметь использовать базовые типы данных, арифметические операции, условные логические конструкции, итерационные и условные циклы, контейнеры, методы и функции, классы и декораторы на языке программирования Python, а также способен работать с файлами и базовыми модулями и применять основы объектно-ориентированного программирования.
Ссылка на лендинг Образовательной программы	строка не менее 10 символов	https://it-school.tgu-dpo.ru/python

Ссылка на LMS	строка не менее 10 СИМВОЛОВ	https://odin.study/ru/
Страница обучения на курсе	строка не менее 10 СИМВОЛОВ	https://www.odin.study/ru/EducationalProgram/Info/7528

Аттестация

Промежуточная аттестация	
Модуль 1. Введение в Python	
Количество академических часов	1 академический час
Формы контроля	Тестирование
Диагностические инструменты	Тестирование с автоматической оценкой
Показатели и критерии оценивания	Оценка "5" выставляется, если обучающийся набрал 5 баллов за тестирование. Оценка "4" ставится, если обучающийся набрал 4 балла за тестирование. Оценка "3" ставится, если обучающийся набрал 3 балла за тестирование. Оценка "2" ставится, если обучающийся набрал менее 3 баллов за тестирование.
Примеры заданий	<p>1. Какие из имен допустимы для названия переменных в Python? Выберите все подходящие ответы из списка</p> <ul style="list-style-type: none"> ● point_2 ● 2point ● point ● point2 <p>2. Укажите значение переменной x после выполнения следующего кода: Введите ответ числом</p> <pre>x = 3 y = 4 z = x + y z = z + 1 x = y y = 5</pre>

$x = z + y + 7$

3. Дан код:

$x = 23$

$num = 0$ if $x > 10$ else 11

print(num)

Что выведет данный код?

Выбери верный вариант ответа.

1) 11

2) 0

3) 10

4) 23

4. Дан код:

$a = 30$

$b = 12$

$k = 0$

while $a \neq 0$ and $b \neq 0$:

if $a > b$:

$a = a \% b$

else:

$b = b \% a$

print(a+b)

Что выведет данный код?

Выбери верный вариант ответа.

1) 30

2) 12

3) 6

4) 0

5. Дан код:

name = "John"

print(f'Hi, {name}')

		<p>Что выведет данный код? Выбери верный вариант ответа.</p> <p>1) Hi, name 2) Hi, 3) Ошибка 4) Hi, John</p>
Шкала оценивания, нижнее значение	строка не менее 10 символов	"2"
Шкала оценивания, верхнее значение	строка не менее 10 символов	"5"
Шкала оценивания, минимальный проходной балл	строка не менее 10 символов	"3"
Модуль 2. Погружение в Python		
Количество академических часов	строка не менее 10 символов	1 академический час
Формы контроля	строка не менее 10 символов	Тестирование
Диагностические инструменты	строка не менее 10 символов	Тестирование с автоматической оценкой
Показатели и критерии оценивания	строка не менее 10 символов	<p>Оценка "5" выставляется, если обучающийся набрал 5 баллов за тестирование.</p> <p>Оценка "4" ставится, если обучающийся набрал 4 балла за тестирование.</p> <p>Оценка "3" ставится, если обучающийся набрал 3 балла за тестирование.</p> <p>Оценка "2" ставится, если обучающийся набрал менее 3 баллов за тестирование.</p>
Примеры заданий	строка не менее 10 символов	<p>1. Что выведет следующий программный код?</p> <pre>myset = {'Yellow', 'Orange', 'Black'} print(myset[1])</pre> <p>Выберите один вариант из списка</p>

ошибка во время выполнения кода

1. Black
2. Orange
3. Yellow

2. Программист написал программу, состоящую из одной строки `print(len(set(input())))`

Что выведет эта программа при вводе следующих строк:
карабас-барabas
Введите ответ

3. Что увидим в результате выполнения приведенного ниже фрагмента кода?

```
stuff = {1: 'aaa', 2: 'bbb', 3: 'vvv'}  
print(stuff[3])
```

Выберите один вариант из списка

1. 1
2. aaa
3. 3
4. vv
5. 2
6. bbb

4. Что покажет приведенный ниже фрагмент кода?

```
dict = {'понедельник': 1, 'вторник': 2, 'среда': 3}  
print(dict.get('пятница', 'Не найдено'))
```

Выберите один вариант из списка

1. произойдет ошибка
2. 2
3. пятница
4. Не найдено
5. понедельник
6. 1
7. 3
8. среда
9. вторник

		<p>5. Когда программа закончила использовать файл, она должна выбрать один вариант из списка</p> <ol style="list-style-type: none"> 1. открыть файл 2. стереть файл 3. закрыть файл 4. зашифровать файл
Шкала оценивания, нижнее значение	строка не менее 10 символов	"2"
Шкала оценивания, верхнее значение	строка не менее 10 символов	"5"
Шкала оценивания, минимальный проходной балл	строка не менее 10 символов	"3"
Модуль 3. Погружение в Python. Продолжение		
Количество академических часов	строка не менее 10 символов	1 академический час
Формы контроля	строка не менее 10 символов	Тестирование
Диагностические инструменты	строка не менее 10 символов	Тестирование с автоматической оценкой
Показатели и критерии оценивания	строка не менее 10 символов	<p>Оценка "5" выставляется, если обучающийся набрал 5 баллов за тестирование.</p> <p>Оценка "4" ставится, если обучающийся набрал 4 балла за тестирование.</p> <p>Оценка "3" ставится, если обучающийся набрал 3 балла за тестирование.</p> <p>Оценка "2" ставится, если обучающийся набрал менее 3 баллов за тестирование.</p>
Примеры заданий	строка не менее 10 символов	<p>1. Как записывается списочное выражение?</p> <p>Выберите один вариант из списка</p>

1. [переменная for выражение in последовательность]
2. [выражение for последовательность in переменная]
3. [переменная for переменная in выражение]
4. [выражение for переменная in последовательность]

2. В переменной d хранится следующее выражение-генератор
from random import randint
d = (randint(1, 20) for i in range(7))

Какие преобразования допустимо делать с переменной d?
Выберите все подходящие ответы из списка

1. list(d)
2. set(d)
3. tuple(d)
4. dict(d)
5. int(d)

3. Какой магический метод вызывается после создания экземпляра класса?

Выбери верный вариант ответа.

1. __init__
2. __del__
3. __str__
4. __set__

4. Что будет выведено на экран при выполнении данной строки кода
print(type([1, 2, 3]))?

Выбери верный вариант ответа.

1. <class 'list'>
2. Ошибка
3. object
4. 1 2 3

5. Что будет выведено на экран при выполнении данного кода?

```
class Test:  
    pass  
test = Test()
```

		<pre>setattr(test, 'value', 5) print(test.value)</pre> <p>Выбери верный вариант ответа.</p> <ol style="list-style-type: none"> 1. value 2. 5 3. Ошибка 4. test
Шкала оценивания, нижнее значение	строка не менее 10 символов	"2"
Шкала оценивания, верхнее значение	строка не менее 10 символов	"5"
Шкала оценивания, минимальный проходной балл	строка не менее 10 символов	"3"
Модуль 4. Прикладная разработка на Python		
Количество академических часов	строка не менее 10 символов	2 академических часа
Формы контроля	строка не менее 10 символов	Итоговое тестирование
Диагностические инструменты	строка не менее 10 символов	Тестирование с автоматической оценкой
Показатели и критерии оценивания	строка не менее 10 символов	<p>Оценка "5" выставляется, если обучающийся набрал 5 баллов за итоговое тестирование.</p> <p>Оценка "4" ставится, если обучающийся набрал 4 балла за итоговое тестирование.</p> <p>Оценка "3" ставится, если обучающийся набрал 3 балла за итоговое тестирование.</p> <p>Оценка "2" ставится, если обучающийся набрал менее 3 баллов за итоговое тестирование.</p>

Примеры заданий

строка не менее 10
символов

1. Дан код:
array = [1, 10, 2, 3, 4, 6, 5, 8, 7, 9]
n = len(array)

for i in range(n - 1, 0, -1):
 k = 0
 for j in range(0, i + 1):
 if array[j] > array[k]:
 k = j
 array[i], array[k] = array[k], array[i]
 print(array)

Какая сортировка реализована в программе?
Выбери верный вариант ответа.

- 1) Сортировка слиянием
- 2) Сортировка выбором
- 3) Сортировка вставками
- 4) Быстрая сортировка

2. Дан код:

```
import turtle  
import time  
  
turtle.reset()  
turtle.down()  
i = 1  
while i <= 4:  
    turtle.right(90)  
    turtle.forward(100)  
    i += 1  
turtle.up()  
time.sleep(5)
```

Что выведет данный код?
Выбери верный вариант ответа.

- 1) Незамкнутая ломаная линия

- 2) Правильный треугольник
- 3) Квадрат
- 4) Правильный пятиугольник

3. Дан стек [1, 2, 3, 4, 5]. Как будет выглядеть стек после выполнения метода pop(), если сначала выполнить метод push(1)?
Выбери верный вариант ответа.

- 1) [2, 3, 4, 5, 1]
- 2) [2, 3, 4, 5]
- 3) [1, 2, 3, 4, 5, 1]
- 4) [1, 2, 3, 4, 5]

4. Дан код:

```
from bs4 import BeautifulSoup
a = "<div class='col-lg-4'>
<div class='card'>
<a href='/list/9/'><img alt='' class='card' src='/img/9.jpg' /></a>
<div class='card-body'>
<h4 class='card-title'>
<a href='/list/9/'>1 кг. яблочек</a>
</h4> <h5>240 р.</h5>
</div> </div> </div>"
bs = BeautifulSoup(a, 'html.parser')
itemName =
itemPrice =
print(f{itemName} за {itemPrice})
```

Дополните данный код программы, чтобы результатом вывода на экран было следующее: 240 р. за 1 кг. яблочек.

5. Дан код:

```
import tkinter
window = tkinter.Tk()
window.title("My program")
window.geometry("400x300")
```

Дополните данный код программы, чтобы запустилось графическое окно.

Шкала оценивания, нижнее значение	строка не менее 10 символов	"2"
Шкала оценивания, верхнее значение	строка не менее 10 символов	"5"
Шкала оценивания, минимальный проходной балл	строка не менее 10 символов	"3"

Преподаватели

ФИО	Наименование основного места работы	Должность	Высшее образование или среднее профессиональное образование направлен ию	Высшее образование или среднее профессиональное образование направлен ию	Ссылка на веб-страницы с портфолио	Информация о курсах повышения квалификации по профилю преподаваемой дисциплины (за последние 3 года)	Пройдена промежуточная аттестация не менее чем за два года обучения по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направлению ДОП	Отметка о полученном согласии на обработку персональных данных
строка от 2 до 100 символов	строка от 2 до 255 символов.	строка от 2 до 255 символов	да/нет	да/нет	строка		да/нет	да/нет
Бубарева Олеся Александровна	Бийский технологический институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Алтайский государственный технический	Доцент	нет	да		"Программа повышения квалификации наставников по проведению рефлексии профессиональных проб и модели осознанности и целеустремленности у обучающихся 6-11-х классов", 2020 г., Повышение квалификации "Реализация современных информационно- коммуникационных образовательных технологий в	да	да

	университет им. И.И. Ползунова»								образовательных учреждений", 2021 г., Повышение квалификации "Психолого-педагогические и организационные аспекты сопровождения инклюзивного образования студентов с ОВЗ и инвалидностью в вузе", 2020 г.	да	
Величко Екатерина Геннадьевна	Центр детского научного и инженерно-технического творчества «Наследники Ползунова»	преподаватель дисциплины «Информатика и программирование для робототехников»	нет	да						да	
Поротов Виктор Витальевич	Центр детского научного и инженерно-технического творчества «Наследники Ползунова»	преподаватель дисциплины «Информатика и программирование для робототехников», преподаватель программирования	нет	нет						да	
Фартышев Дмитрий Александрович	МБУ ДО «Центр внешней работы «Малая Академия», г. Рубцовск	педагог дополнительного образования	нет	да					Повышение квалификации «Организация работы с обучающимися с ограниченными возможностями здоровья в соответствии с ФГОС», 2021 г. Повышение квалификации Цифровизация образовательного процесса в школах 2022 г.,	да	

								“Современные языки программирования интегрированной оболочки Microsoft Visual Studio C++ Net, C++ Net с использованием структурного и объектно-ориентированного метода разработки корпоративных систем”, 2021 г. Повышение квалификации “Развитие ИКТ-компетенций педагога для повышения образовательных результатов ученика”, 2021 г. Повышение квалификации “Защита персональных данных в образовательных организациях”, 2022 г.	да	да	
Щигорев Дмитрий Дмитриевич	АлГТУ им. И.И. Ползунова, отдел разработки информационных систем	младший программист	нет	да					да		
Рогозянский Андрей Денисович	ООО Консалт Инфо	программист	нет	да					да		
Вайцель Никита Сергеевич	КГБУЗ "Консультативно-диагностический центр", г.Бийск,	программист	нет	да					да		
Лунев Валерий Константинович	Центр детского научного и инженерно-	Преподаватель робототехники, информатики и	нет	да					да		

		технического творчества «Наследники Ползунова»	программирования для робототехников, олимпиадной информатики и программирования	нет	да		робототехники и программирования” 2022, Повышение квалификации “Актуальные вопросы и современные практики естественнонаучного и инженерно-технического образования одаренных школьников” 2022	да	
Шишин Константин Сергеевич	ООО «Бинго-софт»		системный аналитик	нет	да			да	
Сидоренко Антон Игоревич	Бийский технологический институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Алтайский государственный технический университет им. И.И. Ползунова»		Доцент	нет	да		Повышение квалификации: «Актуальные вопросы и современные практики естественнонаучного и инженерно-технического образования школьников», 2022 г.	да	
Степанок Екатерина Геннадьевна	ЦДНИТТ при КузГТУ «УникУм»,		преподаватель	нет	да		Компьютерные курсы «Основы автоматизации тестирования» Asadem IT School Компьютерные курсы «Тестирование ПО»	да	
Астраханцев Евгений	Центр детского и научного и		преподаватель робототехники и	нет	нет		Сертификат о прохождении обучения «Актуальные	да	

Сергеевич	инженерно-технического творчества «Наследники Ползунова»	программирования.			вопросы и современные практики естественнонаучного и инженерно-технического образования одарённых школьников»		
-----------	--	-------------------	--	--	---	--	--

Рабочая программа с описанием каждого модуля

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
<p>Модуль 1. Введение в Python Введение в структурное программирование на языке Python: типы данных, условия, циклы</p>	<p>Тема 1. Введение в программирование</p>	<p>Задачи профессии программиста. Сферы применения программирования. Программа как вид информационной модели. Программа как информационный процесс. Формальные и естественные языки. Знакомство с понятиями: программа, алгоритм, язык программирования, среда выполнения алгоритма, исполнитель. Решение практических заданий по анализу, исполнению алгоритмов, перевод словесного описания алгоритма в другую форму записи. Входные, выходные, промежуточные данные.</p>	<p>теоретические занятия</p>	<p>1</p>
		<p>Работа по закреплению материала. Упражнения по переводу словесного описания алгоритма в другую форму записи; по определению среды исполнения.</p>	<p>практические занятия</p>	<p>2</p>
	<p>Тема 2. Введение в язык программирования Python</p>	<p>Работа по закреплению материала. Упражнения по переводу словесного описания алгоритма в другую форму записи; по определению среды исполнения. История языка программирования Python. Особенности языка программирования Python. Среда программирования на языке Python. Установка среды разработки Python. Комментарии.</p>	<p>самостоятельная работа</p>	<p>1</p>
		<p>Установка Python для разработки на Python. Знакомство со средой программирования. Запуск программы. Настройка интерфейса. Режимы работы. Решение практических задач по сохранению файлов; по работе с операторами вывода информации на экран и ввода информации с клавиатуры. Работа по закреплению материала. Упражнения по использованию операторов вывода информации на экран и ввода информации с клавиатуры.</p>	<p>теоретические занятия</p>	<p>1</p>
	<p>Тема 3. Типы данных</p>	<p>Понятие переменной. Имена переменных. Операторы присваивания. Арифметические операции. Выражения и инструкции. Значения и типы. Динамическая типизация. Стандартный ввод/вывод.</p>	<p>практические занятия</p>	<p>2</p>
			<p>самостоятельная работа</p>	<p>1</p>
			<p>теоретические занятия</p>	<p>1</p>

	<p>Решение практических задач по работе с типами данных; преобразованием данных; операторами присваивания; арифметическим операторами; множественным присваиванием; библиотекой math; синтаксисом вызова функций. Целочисленное деление, деление по модулю. Операции с целыми числами. Операции с вещественными числами.</p> <p>Работа по закреплению материала. Упражнения по преобразованию данных. Упражнения с операторами присваивания, арифметических операторов.</p>	<p>практические занятия</p>	<p>2</p>
<p>Тема 4. Логические операторы и выражения</p>	<p>Алгебра логики: высказывания, логические операции. Операции сравнения. Логический тип данных.</p> <p>Решение практических задач по работе с логическими операторами <code>or</code>, <code>and</code>, <code>not</code>. Работа с логическими выражениями; операторами сравнения.</p> <p>Работа по закреплению материала. Упражнения по использованию логических операторов.</p>	<p>теоретические занятия</p> <p>практические занятия</p> <p>самостоятельная работа</p>	<p>1</p> <p>2</p> <p>1</p>
<p>Тема 5. Условия</p>	<p>Условные операторы <code>if</code>, <code>else</code>, <code>elif</code>. Условная конструкция множественного выбора. Условия на языке блок-схем. Условное выполнение. Пробельное форматирование, отступы.</p> <p>Решение практических задач по работе с условными выражениями. Работа с несколькими условиями последовательно; вложенностью операторов. Общие правила структуризации программы на Python.</p> <p>Работа по закреплению материала. Упражнения по составлению условий.</p>	<p>теоретические занятия</p> <p>практические занятия</p> <p>самостоятельная работа</p>	<p>1</p> <p>2</p> <p>2</p>
<p>Тема 6. Циклы</p>	<p>Виды циклов и их особенности. Примеры ситуаций с циклами. Инструкции <code>while</code>. Инструкции <code>break</code>, <code>continue</code>. Блок-схема и псевдокод алгоритма с циклом <code>while</code>. Цикл</p>	<p>теоретические занятия</p>	<p>1</p>

	for. Блок-схема и псевдокод алгоритма с циклом for. Вложенные циклы.			
	Решение практических задач на исполнение алгоритмов с циклами. Оператор цикла с параметрами. Работа с функцией range(), операторами цикла с условием. Классические алгоритмы перебора данных с использованием цикла. Работа со счетчиками и сумматорами. Решение практических задач: алгоритм разбиения числа на цифры, перевод между системами счисления, вычисление факториала, НОД, НОК.	практические занятия	5	
	Работа по закреплению материала. Упражнения по использованию циклических конструкций при решении задач на языке Python.	самостоятельная работа	2	
Тема 7. Строки и символы	Понятие строки в языке программирования Python, функция len(). Обход элементов с помощью цикла for. Срезы строк. Оператор in. Поиск. Сравнение строк. Строковые методы.	теоретические занятия	1	
	Решение практических задач по работе с методами над строковыми данными. Срезы строк. Методы FIND, REPLACE и COUNT. Алгоритмы поиска подстроки в строке, сравнение строк. Отработка навыков решения задач уровня junior на языке Python по материалам модуля 1. Промежуточная аттестация - прохождение тестирования по модулю 1.	практические занятия	4	
	Работа по закреплению материала. Упражнения по применению методов работы со строками при решении задач на языке Python.	самостоятельная работа	1	
		Объем в ак.ч.	Объем в %	
			7	
		теоретические занятия		
		практические занятия	18	50%
		самостоятельная работа	10	28%
		аттестация	1	
ИТОГО:				

	Всего:	36
--	---------------	-----------

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 2. Погружение в Python Элементы функционального программирования на языке Python. Структуры данных. Работа с функциями.	Тема 1. Списки	Понятие списка. Изменяемый тип данных. Обход списка. Добавление, удаление элемента. Понятия индекса элемента списка. Срезы списков. Методы списков. Решение практических задач по созданию списка, добавлению, удалению элементов. Работа с индексами списка, доступ к элементам списка, перебор элементов списка. Алгоритмы поиска и сортировки списка. Работа по переводу списка в строку.	теоретические занятия	1
		Работа по закреплению материала. Упражнения по использованию методов работы со списками при решении задач на языке Python.	самостоятельная работа	1
	Тема 2. Словари, кортежи, множества	Понятия множества, словаря, кортежа. Добавление, удаление элемента. Работа со словарем. Получение значения элемента по ключу. Циклы и словари. Решение практических задач по работе со словарем, множеством. Добавление, удаление, поиск элемента. Работа с кортежем. Распаковка кортежа. Присваивание значения кортежа. Работа с многомерными списками данных. Работа с функцией enumerate().	теоретические занятия	1
		Работа по закреплению материала. Упражнения по использованию методов работы с коллекциями данных при решении задач на языке Python.	практические занятия	5
		Работа по закреплению материала. Упражнения по использованию методов работы с коллекциями данных при решении задач на языке Python.	самостоятельная работа	2
	Тема 3. Стандартные функции	Понятие функции. Инструкция def, return. Возвращаемые значения. Функции и стек вызовов. Оператор Пространства имён и области видимости. Способы передачи параметров в функции.	теоретические занятия	1

	Python	<p>Подключение библиотек. Точечная нотация. Виртуальное окружение.</p> <p>Решение практических задач по применению функций. Объявление и вызов функций. Задания на эксперименты с областью видимости переменных.</p> <p>Работа с подключением библиотек.</p> <p>Работа по закреплению материала. Упражнения по применению функций, подключению библиотек при решении задач на языке Python.</p>	<p>практические занятия</p>	2
<p>Тема 4. Рекурсия. Работа с функциями: functools и lambda функции.</p>	<p>Рекурсивный вызов функций. Разработка рекурсивной функции на базе процедурной ее формы. Последовательная, параллельная и псевдопараллельная рекурсии. Библиотека <code>functools</code>. Анонимные функции – <code>lambda</code>-выражения. Списковый вызов функций. Функция высшего порядка <code>map</code>. Передача функции в качестве аргумента. Функция <code>map</code> – подробнее. Функции <code>filter</code>, <code>zip</code>, <code>reduce</code>.</p> <p>Решение практических задач по разработке рекурсивной функции на базе процедурной ее формы. Выполнение обработки списков рекурсивными функциями.</p> <p>Оценка времени выполнения программ с рекурсией. Способы ускорения рекурсии. Решение практических задач по работе с функцией <code>map</code>. Передача функции в качестве аргумента.</p> <p>Работа по закреплению материала. Упражнения на создание собственных функций с рекурсией и применение <code>lambda</code> функции.</p>	<p>теоретические занятия</p>	1	
<p>Тема 5. Работа с файлами</p>	<p>Файл в памяти компьютера, имена, режимы открытия файла. Обработка исключений. Устойчивость. Оператор форматирования. Чтение данных из файла, запись данных в файл. Двоичные файлы.</p> <p>Решение практических задач по определению режима открытия файла; корректному считыванию с файла</p>	<p>самостоятельная работа</p> <p>теоретические занятия</p> <p>практические занятия</p>	2	

		информации посимвольно и построчно. Запись в файл информации в строчку, а также с переходом на другую строку. Библиотека os. Перебор данных из текстового файла.			
		Работа по закреплению материала. Упражнения по использованию методов работы с файлами при решении задач на языке Python.	самостоятельная работа	2	
Тема 6. Библиотека Numru и массивы данных		Библиотека Numru. Одномерные и многомерные массивы данных. Операции над массивами данных. Методы массивов. Индексация.	теоретические занятия	1	
		Решение практических задач на сложение, вычитание, умножение и сортировку массивов. Задачи на основные операции с массивами: объявление, заполнение, вывод на экран; обработку элементов массивов. Осуществлять поиск в упорядоченном массиве. Отработка навыков решения задач уровня junior на языке Python по материалам модуля 2. Промежуточная аттестация - прохождение тестирования по модулю 2.	практические занятия	4	
		Работа по закреплению материала. Упражнения по работе с массивами данных с помощью библиотеки Numru.	самостоятельная работа	2	
			Объем в ак.ч.	Объем в %	
			теоретические занятия	6	
			практические занятия	19	53%
			самостоятельная работа	10	28%
			аттестация	1	
			Всего:	36	
ИТОГО:					
Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.	

Модуль 3. Погружение в Python. Продолжение Элементы объектно-ориентированного программирования. Решения задач программирования уровня middle	Тема 1. Полезные библиотеки Python	Модули <code>sorted</code> , <code>keyword</code> . Библиотеки <code>time</code> , <code>random</code> , <code>sys</code> и другие. Синтаксический сахар.	теоретические занятия	1
		Решение практических задач с библиотекой <code>time</code> . Работа с методами преобразования дат. Разбор примеров с генерацией случайных значений <code>random</code> . Работа с методами модулей <code>sorted</code> , <code>keyword</code> . «Синтаксический сахар» в Python.	практические занятия	2
		Работа по закреплению материала. Упражнения по применению стандартных модулей при решении задач на языке Python.	самостоятельная работа	1
	Тема 2. Декораторы, ошибки и исключения	Как понимать сообщения об ошибках в коде. Виды ошибок. Способы отладки программы. Понятие исключения, обработка исключений. Собственные исключения.	теоретические занятия	1
		Решение практических задач с декораторами функций. Работа по обработке исключений. Блоки <code>try / except</code> . Обработка исключений. Блоки <code>finally</code> и <code>else</code> .	практические занятия	2
		Работа по закреплению материала. Упражнения на отладку программ с различными типами ошибок на языке Python.	самостоятельная работа	2
	Тема 3. Основы объектно-ориентированного программирования. Классы и методы.	Парадигмы программирования объектно-ориентированного программирования.	теоретические занятия	1
		Конструктор класса. Инициализация объектов класса. Пользовательские типы. Классы и объекты. Поля. Свойства. Ключевое слово <code>this</code> . Ключевое слово <code>static</code> . Метод <code>__init__</code> . Метод <code>__str__</code> . Объектная модель в Python.		
		Решение практических задач по работе с конструкторами класса. Инициализация объектов класса. Работа с динамическими атрибутами класса. Статические методы и методы объекта. Методы принимающие аргументы и возвращающие значения.	практические занятия	5
		Отработка навыков решения задач уровня middle –		

		наследование классов: создание классов игральных карт, карточных колод и карточных комбинаций. Работа по закреплению материала. Упражнения по решению задач в функциональном стиле и в стиле объектно-ориентированного программирования на языке Python.	самостоятельная работа	2	
Тема 4. Алгоритмы сортировки и поиска.		Сортировка подсчетом. Сравнение списков и кортежей. Сортировка вставками. Шейкерная сортировка. Сортировка пузырьком. Анализ алгоритмов поиска.	теоретические занятия	1	
		Решение практических задач с использованием алгоритмов сортировки. Отработка навыков решения задач уровня middle - выбор структуры данных: частотный анализ слов, гистограмма слов, самые распространенные слова, облако слов.	практические занятия	5	
		Работа по закреплению материала. Упражнения по применению алгоритмов сортировки и поиска при решении задач на языке Python.	самостоятельная работа	2	
Тема 5. Основы теории графов		Определение графа, примеры графов. Виды маршрутов в графах. Связные и несвязные графы.	теоретические занятия	1	
		Решение практических задач по построению графов с помощью библиотеки NetworkX. Отработка навыков решения задач уровня middle – библиотека для графов: построение графа социальной сети на языке Python. Отработка навыков решения задач уровня middle на языке Python по материалам модуля 3. Промежуточная аттестация - прохождение тестирования по модулю 3. Работа по закреплению материала. Упражнения по решению задач построения графов на языке Python.	практические занятия	7	
			самостоятельная работа	3	
			Объем в ак.ч.	Объем в %	5
			теоретические занятия		

		ИТОГО:		
		практические занятия	20	56%
		самостоятельная работа	10	28%
		аттестация	1	
		Всего:	36	
Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 4. Прикладная разработка на Python Разработка простейших графических игр. Разработка веб-парсера.	Тема 1. Создание игры. Знакомство с графикой	Рисование геометрических фигур и картинок из геометрических фигур. Работа с цветом. Решение практических задач по работе с графической областью для рисования, рисование разного рода линий. Рисование геометрических фигур и картинок из геометрических фигур. Цвет линии, цвет заливки, раскрашивание нарисованных картинок. Работа по закреплению материала. Упражнения по рисованию геометрических фигур на языке Python	теоретические занятия практические занятия	1 2
	Тема 2. Графика с модулем tkinter	Добавление кнопки. Отображение текста. Создание простой анимации. Решение практических задач по созданию игрового холста. Проработка движения. Добавление пауз, изменение толщины линий, добавление реакций на движение мыши и на нажатие клавиш. Обработка навыков решения задач уровня middle – работа с графикой: написание первой игры на языке Python. Работа по закреплению материала. Упражнения по работе с модулем tkinter на языке Python	самостоятельная работа теоретические занятия практические занятия	1 1 5
	Тема 3. Разработка парсеров. Введение в requests и BeautifulSoup	Введение в Requests. Углубленный анализ HTML кода. HTML. Атрибуты http-запросы, html-страницы. Метод requests.get(). Статус коды. Получаем содержимое response объекта. Парсинг AJAX. Веб-краулеры. Модели веб-краулинга. Сбор данных из форм. Скрапинг данных JS. Обработка изображений и текста.	самостоятельная работа теоретические занятия	2 1

	Решение практических задач по работе с библиотеками requests и BeautifulSoup. Работа с протоколом HTTP. Углубленный анализ HTML кода. Поиск элементов на странице. Поиск узлов и элементов. Обработка изображений и текста. Пагинация. Скрапинг табличных данных. Сохранение результата в Excel. Отработка навыков решения задач уровня middle – разработка парсеров: написание парсера сайта. Работа по закреплению материала. Упражнения по работе по анализу HTML кода на языке Python	практические занятия	5
		самостоятельная работа	2
		теоретические занятия	1
Тема 4. API	Сбор данных через API. Распространенные форматы текстовых файлов: CSV, JSON. Решение практических задач по сбору данных через API. Разбор формата JSON. Решение задач на API VK. Отработка навыков решения задач уровня middle – веб-краулинг с помощью API.	практические занятия	5
		самостоятельная работа	2
Тема 5. Практикум по решению задач	Алгоритмы решения задач обработки данных, требующих использования эффективных алгоритмов и рационального планирования структуры данных. Формализация прикладной задачи. Документирование программ. Отработка навыков решения задач уровня middle: задачи обработки данных. Итоговая аттестация - прохождение итогового тестирования. Работа по закреплению материала. Упражнения по решению задач обработки данных на языке Python	теоретические занятия	1
		практические занятия	5
		самостоятельная работа	2
		Объем в ак.ч.	Объем в %
			5
		теоретические занятия	

ИТОГО:	практические занятия	20	56%
	самостоятельная работа	9	25%
	аттестация	2	
	Всего:	36	

Календарно-тематическое планирование

№	Тема и № модуля	Тема занятия	Кол-во занятий*	Кол-во часов	Дата
1	Модуль 1. Введение в Python	Тема 1. Введение в программирование	3	4	01.10.23
2		Тема 2. Введение в язык программирования Python	3	4	09.10.23
3		Тема 3. Типы данных	3	5	16.10.23
4		Тема 4. Логические операторы и выражения	3	4	23.10.23
5		Тема 5. Условия	3	5	30.10.23
6		Тема 6. Циклы	6	8	06.11.23
7		Тема 7. Строки и символы	4	5	13.11.23
	Аттестация по модулю 1			1	16.11.23
1	Модуль 2. Погружение в Python	Тема 1. Списки	3	4	21.11.23
2		Тема 2. Словари, кортежи, множества	6	8	29.11.23
3		Тема 3. Стандартные функции Python	3	4	11.12.23
4		Тема 4. Рекурсия. Работа с функциями: functools и lambda функции.	6	8	19.12.23
5		Тема 5. Работа с файлами	3	5	08.01.24
6		Тема 6. Библиотека NumPy и массивы данных	4	6	13.01.24
	Аттестация по модулю 2			1	16.01.24

1	Модуль 3. Погружение в Python. Продолжение	Тема 1. Полезные библиотеки Python	3	4	21.01.24
2		Тема 2. Декораторы, ошибки и исключения	3	5	29.01.24
3		Тема 3. Основы объектно-ориентированного программирования. Классы и методы.	6	8	08.02.24
4		Тема 4. Алгоритмы сортировки и поиска.	6	8	19.02.24
5		Тема 5. Основы теории графов	7	10	01.03.24
	Аттестация по модулю 3			1	10.03.24
1	Модуль 4. Прикладная разработка на Python	Тема 1. Создание игры. Знакомство с графикой	3	4	16.03.24
2		Тема 2. Графика с модулем tkinter	6	8	25.03.24
3		Тема 3. Разработка парсеров. Введение в requests и BeautifulSoup	6	8	08.04.24
4		Тема 4. API	6	8	19.04.24
5		Тема 5. Практикум по решению задач	4	6	13.05.24
	Итоговая аттестация			2	25.05.24

*количество занятий не включают часы, отведенные на самостоятельное изучение, и часы, отведенные на прохождение аттестации

Учебно-методические материалы

Наименование поля	Допустимые значения полей	Значение полей	Значение полей	Значение полей
Порядковый номер модуля	строка не менее 10 символов	1	2	3
Методы, формы и технологии	строка не менее 10 символов	Лекционные занятия реализуются на базе инструментов видеоконференцсвязи и включают в себя установочные интерактивные лекции, с обязательным использованием инструмента обратной связи. Для проведения онлайн занятий применяется платформа Zoom. В качестве площадок для совместной синхронной работы будут использованы виртуальные доски и Google-сервисы.		

		<p>Практические занятия реализуются в режиме реального времени в формате индивидуальной или групповой работы обучающихся. Преподаватель разбивает задание на несколько частей и дает школьникам по 10–15 минут на выполнение каждой части (либо это могут быть несколько отдельных заданий). Обучающиеся в течение практики самостоятельно выполняют задания. После завершения времени на выполнение каждого задания происходит его разбор и совместное обсуждение в эфире полученных результатов. Преподаватель может выборочно попросить слушателей продемонстрировать свой экран с программным кодом либо прокомментировать результат в микрофон. Для организации групповой работы необходимо предварительно разбить школьников на подгруппы и распределить их по сессионным залам в Zoom. Изучение теоретического материала (СРС) предполагается до и после онлайн-занятия.</p>
<p>Методические разработки</p>	<p>строка не менее 10 символов</p>	<p>Методические рекомендации и пособия по изучению программы. Программа реализуется в формате онлайн обучения, с применением активных технологий обучения и общения с аудиторией в электронной среде. Обучение в очной части реализуется в виде лекционных и практических занятий, в которых применяются интерактивные формы обучения, семинары, мастер-классы, активные и ситуативные методы обучения, решения кейсов и обсуждения.</p>
<p>Материалы модуля</p>	<p>строка не менее 10 символов</p>	<p>По данной программе имеется электронный учебно-методический комплекс в LMS Odin. УМК содержит: систему навигации по программе (учебно-тематический план, интерактивный график работы по программе, сведения о результатах обучения, о преподавателях программы, чат для объявлений и вопросов преподавателям), набор видеолекций, презентации к лекциям, набор ссылок на внешние образовательные ресурсы и инструменты, систему заданий с подробных установок и инструкциями, списки основной и дополнительной литературы. В электронном курсе реализована система обратной связи, а также онлайн-площадки для взаимного обучения.</p>
<p>Учебная литература</p>	<p>строка не менее 10 символов</p>	<ol style="list-style-type: none"> 1. Доусен М. Программируем на Python / М. Доусен - СПб.: Питер, 2016. - 416 с. 2. Банкрашков, А.В. Программирование для детей на языке Python / А.В. Банкрашков. - М.: АСТ, 2018. - 288 с. 3. Васильев, А. Н. Python на примерах. Практический курс по программированию [Текст] / А.Н. Васильев. – СПб.: Наука и техника, 2016. – 432 с.

	<p>4. Злагопольский Д.М. Основы программирования на языке Python. – М.: ДМК Пресс, 2017. – 284 с.</p> <p>5. Лутц М. Изучаем Python, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 1280 с.</p> <p>6. Лобанович Б. Простой Python. Современный стиль программирования / Б. Лобанович. - СПб.: Питер, 2016. - 480с.</p> <p>7. МакГрат, Майк. Программирование на Python для начинающих [Текст] / Майк МакГрат; [пер. с англ. М.А. Райтмана]. – М.: ЭКСМО, 2015. – 194 с.</p> <p>8. Мэгиз, Э. Изучаем Python. Программирование игр, визуализация данных, веб-приложения / Э. Мэгиз. - СПб.: Питер, 2017. - 320 с.</p> <p>9. Саммерфилд, М. Программирование на Python 3. Подробное руководство / М. Саммерфилд. - М.: Символ-Плюс, 2011. - 608 с.</p> <p>10. Саммерфилд М. Python на практике / М. Саммерфилд, пер. А.А. Слинкин – М.: ДМК-Пресс, 2014. - 338с.</p> <p>11. Свейгарт, Эл. Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих. Пер. с англ. — М.: Вильямс, 2016. – 592 с.</p>
--	--

Материально-технические условия реализации программы

Наименование поля	Допустимые значения полей	Значение полей	Значение полей	Значение полей
Порядковый номер модуля	строка не менее 10 символов	1	2	3
Наименование требуемого оборудования	строка не менее 2 символов	Manufacturer - Acer Model - Veriton M6660G Size - 17x42x37 cm Extra details - CPU i7 8700 / RAM 32 GB DDR4 2400 GHz / HDD 1Tb / SSD 256 / nVidia GeForce GTX1050Ti GPU 4 GB / Win10 и выше мышь, клавиатура, монитор		
Наименование требуемого программного обеспечения	строка не менее 2 символов	ОС - MS Windows 10 и выше Браузер PyCharm Community		

Электронные информационные ресурсы	строка не менее 10 символов	<ol style="list-style-type: none"> 1. https://www.python.org/ 2. https://pythonworld.ru/samouchitel-python 3. https://pythoner.name/
Электронные образовательные ресурсы	строка не менее 10 символов	<ol style="list-style-type: none"> 1. Тренажер по Python https://www.w3schools.com/python/default.asp 2. Тренажер по Python https://pythontutor.ru/