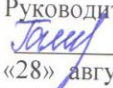
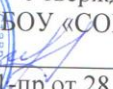


Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа № 18»
Изобильненского муниципального округа Ставропольского края

Согласовано:
Руководитель ЦО
 Е.В.Галигузова
«28» августа 2024 г.



Утверждаю
Директор МБОУ «СОШ №18» ИМОСК
 С.И.Чепетова
Приказ № 341-пр от 28.08.2024



РАБОЧАЯ ПРОГРАММА

курса внеурочной деятельности
«Программирование на Python, как инструмент решения задач единого
государственного экзамена»
для 10,11 классов с использованием
оборудования центра «Точка роста»
на 2023/24 учебный год

Педагог дополнительного образования:
Суглобова Н. Н.

Название курса	Основы программирования на языке Python
Уровень образования	среднее общее образование
класс	10-11 класс
Количество часов	102 ч
Количество часов в неделю	3
Аннотация	<p>Программа рассчитана на изучение материала в течение одного года обучения. Программа курса разработана в соответствии с требованиями ФГОС среднего общего образования, Концепции духовно-нравственного развития и воспитания личности гражданина России, в соответствии с содержанием Кодификатора требований к уровню подготовки выпускников образовательных организаций для проведения единого государственного экзамена.</p> <p>Учебные занятия включают в себя лекции и практические занятия. Основной тип - комбинированный урок. Каждая тема курса начинается с постановки задачи. Теоретический материал излагается в форме мини-лекций. После изучения теоретического материала выполняются задания для активного обучения, практические задания для закрепления.</p> <p>Формы методы контроля: решение практических задач программирования (практическое занятие), занятие – соревнование, самостоятельная работа. Основным дидактическим средством для предлагаемого курса являются тексты рассматриваемых типов задач, которые могут быть выбраны из различных источников, но в большей степени в числе для подготовку к единому государственному экзамену.</p> <p>Практические занятия составляют важную часть теоретической и профессиональной подготовки. Они направлены на формирование практических навыков и умений. Практические занятия создают оптимальные дидактические условия для деятельностного освоения обучающимися содержания и методологии изучаемой дисциплины, использование специального оборудования, технических средств. Ведущей дидактической целью практических занятий является формирование практических умений – профессиональных (выполнять определённые действия, операции, необходимые в последующем в профессиональной деятельности).</p>
Актуальность	<p>Python - один из самых универсальных языков программирования. Python входит в топ-10 самых востребованных языков программирования. Самые прогрессивные IT-компании, такие как, Google, Pixar, Youtube, Instagram, NASA, Intel, Pinterest используют именно данный язык программирования.</p> <p>Python — высокоуровневый язык общего назначения. Это значит, что такой язык быстр и удобен в работе, а программы, написанные на нём, просты для понимания программистами. Обычно его используют для веб-разработки, анализа данных, написания скриптов и игр. Сильные стороны языка, такие как, простота в освоении, низкий порог входа, читабельность,</p>

	<p>универсальность, большое и активное сообщество, позволяют начать изучать его с достаточно раннего возраста. Также Python часто советуют в качестве первого языка начинающим программистам.</p> <p>Такая популярность вызвана универсальностью и простотой изучения языка — он становится главным ориентиром новичков. Чтобы работать с ним, необязательно быть продвинутым программистом.</p> <p>На рынке всё ещё чувствуется недостаток специалистов по разработке программного обеспечения. Поэтому количество открытых вакансий на должность разработчика — показатель востребованности специалистов. В 2022 году Java, Python, JavaScript и C++ входили в число популярных языков, требуемых работодателями.</p> <p>Особая роль курса заключается в формировании алгоритмического мышления ученика. Задача школы популяризировать и актуализировать среди обучающихся необходимость знания программирования в жизни, а также формирование IT-культуры обучающихся, повышение их компьютерной грамотности.</p> <p>Решение задач программирования способствует раскрытию и расширению прикладного аспекта изучаемого материала.</p> <p>Обучающиеся, сдающие экзамен по информатике и ИКТ, планируют связать дальнейшую деятельность с IT-сферой, с программированием, разработкой различных продуктов. Поэтому знание такого универсального языка, как Python, необходимо для будущего выпускника школы и абитуриента университета.</p> <p>Практически все задачи единого государственного экзамена решаются с использованием языка Python, поэтому изучение данного языка, значительно облегчит подготовку обучающихся к экзамену, и будет способствовать успешному результату сдачи экзамена.</p>
Цель курса	<p>Способствовать формированию и развитию у обучающихся навыков решения задач программирования, анализа и систематизации полученных ранее знаний, развитие алгоритмического мышления и культуры обучающихся, повышение интереса к изучению программирования, информатики и информационно-коммуникационных технологий, способности к личному самоопределению и самореализации.</p>
Задачи курса	<ol style="list-style-type: none"> 1. расширение и углубление представления обучающихся о приемах и методах решения алгоритмических задач; 2. формирование и развитие аналитического, логического мышления при проектировании решения задач у обучающихся; 3. развитие самостоятельно анализировать и решать задачи по образцу и в незнакомой ситуации; 4. развитие самостоятельно анализировать и решать алгоритмические задачи; 5. развитие математических способностей, повышение уровня математической грамотности; 6. развитие математической интуиции, нахождение оптимального способа решения задач; 7. формирование навыка работы с дополнительной литературой, использования различных Интернет-ресурсов;

	8. формирование навыков соблюдения единых требований к правилам решения задач, включенных в основной государственный экзамен и единый государственный экзамен. 9. формирование навыков решения олимпиадных задач программирования.
Личностные, метапредметные, предметные результаты освоения курса	
<ul style="list-style-type: none"> • личностные результаты 	1. начало профессионального самоопределения, ознакомление с миром профессий, связанных с информатикой. 2. умение грамотно, ясно и точно излагать свои мысли в устной и письменной форме, понимать смысл поставленной задачи, выстраивать аргументацию, приводить; 3. умение грамотно структурировать данные, оформлять решение алгоритмической задачи; 4. развитие критического мышления, умение распознавать логически некорректные высказывания; 5. креативное мышление, инициатива, находчивость, активность при решении математических задач; 6. умение контролировать процесс и результат учебной математической и алгоритмической деятельности; 7. способность к восприятию математических и алгоритмических объектов, задач, решений, рассуждений;
<ul style="list-style-type: none"> • метапредметные результаты 	1. осмысленно читать и воспринимать на слух текст задания; 2. уметь извлекать и анализировать информацию, полученную из текста; 3. уметь критически оценивать данную информацию; 4. уметь читать таблицы, блок-схемы, условные обозначения, программу, написанную на языке программирования, а также на алгоритмическом языке. 5. иметь представления об идеях и о методах математики как об универсальном языке науки и техники, о средстве моделирования явлений и процессов; 6. иметь представления об идеях и о методах программирования, о средствах моделирования на нем; 7. уметь планировать свою деятельность, ставить задачи, находить пути их решения; 8. уметь видеть математическую задачу в контексте проблемной ситуации в задачах программирования; 9. уметь находить в различных источниках информацию, необходимую для решения проблем, и представлять ее в понятной форме; принимать решение в условиях неполной и избыточной, точной и вероятностной информации; 10. уметь выдвигать гипотезы при решении учебных задач и понимать необходимость их проверки; умение применять индуктивные и дедуктивные способы рассуждений, видеть различные стратегии решения задач; 11. уметь действовать в соответствии с предложенным алгоритмом; 12. уметь самостоятельно ставить цели, выбирать и создавать алгоритмы для решения учебных проблем; 13. уметь планировать и осуществлять деятельность, направленную на решение задач исследовательского характера;
<ul style="list-style-type: none"> • предметные результаты 	знать/понимать: <ul style="list-style-type: none"> • этапы решения задачи на компьютере; • типы данных; • базовые конструкции изучаемых языков

	<p>программирования;</p> <ul style="list-style-type: none">• принципы структурного и модульного программирования;• принципы объектно-ориентированного программирования;• основы создания приложений в Python;• инструментальные средства Python. <p>уметь:</p> <ul style="list-style-type: none">• работать в среде программирования;• реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;• настраивать рабочую среду Python.
--	---

2. Содержание рабочей программы

№	Тема	Виды учебной деятельности	Количество часов
1.	Язык программирования Python. Введение.	Знать основные конструкции языка программирования, понятия переменной, оператора присваивания, типы переменных, переводить число в строку, строковый тип в числовой	5
2.	Реализация вычислений и ветвлений.	Знать математические операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, знать принципиальные отличия между формальными, локальными и глобальными переменными. Знать логический тип данных, логические выражения и операторы, сложные условные выражения (логические операции). Уметь реализовывать ветвления в языке Python.	12
3.	Реализация циклических, вспомогательных алгоритмов.	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы (линейные, разветвляющиеся и с циклами),	17
4.	Функции. Рекурсия	иметь представление о значении полноценных процедур и функций для структурно-ориентированного языка высокого уровня, знать правила описания процедур и функций в Python и построение вызова процедуры, знать область действия описаний в процедурах, иметь представление о рекурсии, знать ее реализацию на Python, владеть основными приемами формирования процедуры и функции	8
5.	Встроенные модули	знать некоторые встроенные модули, в том числе кэширования данных, использование которых позволяет решать сложные олимпиадные задачи. Знать отличия между функциями некоторых модулей, таких как product и permutation	12
6.	Позиционные системы счисления	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы. Уметь переводить числа из одной системы счисления в другую в ручную и посредством языка программирования, находить количество требуемых заданием цифр в числе любой системы счисления, проводить различные арифметические операции с числами в различных системах счисления	11
7.	Множества. Словари. Кортежи. Списки.	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы (линейные, разветвляющиеся и с циклами). Уметь использовать листовое включение (list comprehension). Знать отличия множеств от словарей, списков и кортежей. Уметь применять ранее полученные знания при решении задач с использованием списков, кортежей, множеств и словарей.	9

8.	Файлы . Работа с файлами.	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы . Знать операторы открытия файлов и записывать информацию в файл. Уметь решать задачи, которые требуют обработку файлов.	3
9.	Решение задач повышенной сложности	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы . уметь решать задачи единого государственного экзамена, используя знания, умения, навыки, полученные ранее	25

Календарно-тематическое планирование

№ п/п	Название раздела, темы	Всего часов	Дата проведения	
			План.	Факт.
1	<i>Язык программирования Python. Введение.</i>	5		
1.	Ввод и вывод данных	1		
2.	Числа. Арифметические операции. Основные операторы.	1		
3.	Числа. Арифметические операции. Основные операторы.	1		
4.	Строки. Операции со строками. Основные операторы.	1		
5.	Строки. Операции со строками. Основные операторы.	1		
2	<i>Реализация вычислений и ветвлений.</i>	12		
6.	Условный алгоритм. Полное и неполное ветвление.	1		
7.	Условный алгоритм. Полное и неполное ветвление.	1		
8.	Логические операнды	1		
9.	Решение задач с использованием логических операндов	1		
10.	Решение задач с использованием логических операндов	1		
11.	Решение задач с использованием all	1		
12.	Решение логических задач с использованием Python (задание 2)	1		
13.	Решение логических задач с использованием Python (задание 2)	1		
14.	Решение логических задач с использованием Python (задание 15)	1		
15.	<i>Игра «Брэйн-ринг за минуту» по теме: «Решение логических задач с использованием Python» (задание 15)</i>	1		
16.	Решение логических задач с использованием Python (задание 15)	1		
17.	Решение логических задач с использованием Python (задание 15)	1		
3	<i>Реализация циклических, вспомогательных алгоритмов.</i>	17		
18.	Цикл for	1		
19.	Цикл for	1		
20.	Цикл while	1		
21.	Цикл while	1		
22.	Инструкции break, continue, ветвь else в циклах	1		
23.	Создание простой программы исполнителя с помощью языка программирования Python (задание 5)	1		
24.	Создание простой программы исполнителя с помощью языка программирования Python (задание 5)	1		
25.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 6)	1		
26.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 6)	1		
27.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 22)	1		
28.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 22)	1		
29.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 22)	1		
30.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 22)	1		
31.	Решение задач на анализ алгоритмов, содержащих цикл и ветвление (задание 22)	1		
32.	Решение задач на исполнение алгоритма для конкретного	1		

	исполнителя с определенным набором команд посредством языка программирования Python (задание 12)			
33.	Решение задач на исполнение алгоритма для конкретного исполнителя с определенным набором команд посредством языка программирования Python (задание 12)	1		
34.	Решение задач на исполнение алгоритма для конкретного исполнителя с определенным набором команд посредством языка программирования Python (задание 12)	1		
4	<i>Функции. Рекурсия</i>	8		
35.	Определение функций, инструкция return	1		
36.	Описание переменных в функции, локальные и глобальные переменные	1		
37.	Аргументы функции	1		
38.	Решение задач с использованием рекуррентных функций (задание 16)	1		
39.	Решение задач с использованием рекуррентных функций (задание 16)	1		
40.	Решение задач с использованием рекуррентных функций (задание 16)	1		
41.	Решение задач с использованием рекуррентных функций (задание 16)	1		
42.	Решение задач с использованием рекуррентных функций (задание 16)	1		
5	<i>Встроенные модули</i>	12		
43.	None – эквивалент null в python	1		
44.	Модуль functools	1		
45.	Решение задач	1		
46.	Модуль Lru_cache	1		
47.	Модуль itertools в Python	1		
48.	Функция product() модуля itertools в Python	1		
49.	Функция permutations() модуля itertools в Python	1		
50.	Знакомство с методом join	1		
51.	Решение задач комбинаторики посредством языка программирования Python (задание 8)	1		
52.	Решение задач комбинаторики посредством языка программирования Python (задание 8)	1		
53.	Решение задач комбинаторики посредством языка программирования Python (задание 8)	1		
54.	Решение задач комбинаторики посредством языка программирования Python (задание 8)	1		
6	<i>Позиционные системы счисления</i>	11		
55.	Перевод чисел из десятичной системы счисления в двоичную, восьмеричную, шестнадцатеричную и n-ю системы счисления	1		
56.	Перевод чисел из двоичной, восьмеричной, шестнадцатеричной и n-й системы счисления в десятичную систему счисления	1		
57.	Перевод из двоичной в четверичную, восьмеричной, шестнадцатеричную	1		
58.	Перевод чисел из десятичной системы счисления в двоичную, восьмеричную, шестнадцатеричную и n-ю системы счисления средствами языка программирования Python	1		
59.	Перевод чисел из двоичной, восьмеричной, шестнадцатеричной и n-й системы счисления в десятичную систему счисления средствами языка программирования Python	1		

60.	Перевод из n-й системы счисления в k-ю средствами языка программирования Python	1		
61.	Решение задач (задание 14)	1		
62.	Решение задач (задание 14)	1		
63.	Решение задач (задание 14)	1		
64.	Решение задач (задание 14)	1		
65.	Решение задач (задание 14)	1		
7	<i>Множества. Словари. Кортежи. Списки</i>	9		
66.	Понятие «Множества» в python. Операции с множествами	1		
67.	Решение задач по теме «Множества»	1		
68.	Понятие «Словари» в python. Методы словарей	1		
69.	Решение задач по теме «Словари»	1		
70.	Понятие «Кортеж» в python. Операции с кортежами	1		
71.	Решение задач по теме «Кортеж». Понятие «Список» в python	1		
72.	Функции и методы списков.	1		
73.	Индексы и срезы	1		
74.	Решение задач по теме «Список»	1		
8	<i>Файлы. Работа с файлами</i>	3		
75.	Чтение из файла. Запись в файл.	1		
76.	Решение задач	1		
77.	Решение задач	1		
9	<i>Решение задач повышенной сложности</i>	25		
78.	Решение задач на составление алгоритмов обработки числовой последовательности с помощью языка программирования Python (задние 17)	1		
79.	Решение задач на составление алгоритмов обработки числовой последовательности с помощью языка программирования Python (задние 17)	1		
80.	Решение задач на составление алгоритмов обработки числовой последовательности с помощью языка программирования Python (задние 17)	1		
81.	Решение задач с использованием циклов и ветвлений с помощью языка программирования Python (задние 23)	1		
82.	Решение задач с использованием циклов и ветвлений с помощью языка программирования Python (задние 23)	1		
83.	Решение задач с использованием циклов и ветвлений с помощью языка программирования Python (задние 23)	1		
84.	Создание программ для обработки символьной информации на языке программирования Python (задние 24)	1		
85.	Создание программ для обработки символьной информации на языке программирования Python (задние 24)	1		
86.	Создание программ для обработки символьной информации на языке программирования Python (задние 24)	1		
87.	Создание программ для обработки символьной информации на языке программирования Python (задние 24)	1		
88.	Создание программ для обработки целочисленной информации на языке программирования Python (задние 25)	1		
89.	Создание программ для обработки целочисленной информации на языке программирования Python (задние 25)	1		
90.	Создание программ для обработки целочисленной информации на языке программирования Python (задние 25)	1		
91.	Создание программ для обработки целочисленной информации на языке программирования Python (задние 25)	1		
92.	Создание программ для обработки целочисленной информации на языке программирования Python (задние 25)	1		

93.	Создание программ для обработки целочисленной информации на языке программирования Python (задние 25)	1		
94.	Создание программ для обработки целочисленной информации на языке программирования Python с использованием сортировки (задние 26)	1		
95.	Создание программ для обработки целочисленной информации на языке программирования Python с использованием сортировки (задние 26)	1		
96.	Создание программ для обработки целочисленной информации на языке программирования Python с использованием сортировки (задние 26)	1		
97.	Создание программ для обработки целочисленной информации на языке программирования Python с использованием сортировки (задние 26)	1		
98.	Создание программ для анализа числовых последовательностей на языке программирования Python (задние 27)	1		
99.	Создание программ для анализа числовых последовательностей на языке программирования Python (задние 27)	1		
100.	Создание программ для анализа числовых последовательностей на языке программирования Python (задние 27)	1		
101.	Создание программ для анализа числовых последовательностей на языке программирования Python (задние 27)	1		
102.	Создание программ для анализа числовых последовательностей на языке программирования Python (задние 27)	1		

Список учебно-методической литературы

1. А.М. Нигмедзянова, Е.А. Ефремова. Решение задач программирования для подготовки к ЕГЭ по информатике на языке PYTHON. Учебно-методическое пособие. 2019
2. Г.Россум, Ф.Л.Дж.Дрейк, Д.С. Откидач и др. Язык программирования Python. 2001
3. Д.Мусин. Самоучитель Python. Выпуск 0.2. 2017
4. С.К. Буйначев, Н.Ю. Боклаг. Основы программирования на языке Python. Учебное пособие. 2014
5. М.Саммерфилд. Программирование на Python 3. Подробное руководство. 2009
6. Кодификатор проверяемых требований к результатам освоения основной образовательной программы среднего общего образования и элементов содержания для проведения единого государственного экзамена по информатике. ФИПИ. 2022
7. Спецификация контрольных измерительных материалов для проведения в 2022 году единого государственного экзамена по информатике. ФИПИ. 2022