

Муниципальное бюджетное общеобразовательное учреждение  
«Каменская средняя общеобразовательная школа»

СОГЛАСОВАН  
с зам. директора по ВР  
Данилова Е.С.  
от 29.08.2024 г.

УТВЕРЖДЕН  
приказом директора  
МБОУ «Каменская СОШ»  
№ 140 от 29.08.2024 г.

**Дополнительная общеразвивающая программа  
"Python. Основы программирования"**

Адресат программы: 7 - 11 класс  
Срок реализации: 1 год

Разработчик (и) программы:  
Ребзон Сергей Леонидович  
педагог дополнительного  
образования

с. Каменка, 2024 г.

## СОДЕРЖАНИЕ

Пояснительная записка.....	3
Комплекс основных характеристик программы.....	4
Содержание программы.....	4
Планируемые результаты.....	4
Комплекс организационно-педагогических условий.....	6
Учебный план.....	6
Календарный учебный график.....	6
Календарно тематическое планирование.....	7
Дополнительные материалы.....	7
Условия реализации.....	12
Список литературы.....	12
Оценочные материалы.....	12

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

(аннотация)

Python – язык программирования общего назначения, оптимизированный для создания качественного программного обеспечения. Язык Python один из самых используемых языков программирования в мире в таких областях, как создание веб-сценариев, системное программирование, создание пользовательских интерфейсов, настройка программных продуктов под пользователя, численное программирование и др. Python относительно прост в освоении и хорошо подходит для решения олимпиадных задач по программированию. Python дает более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

### Цель Программы:

- ознакомить обучающихся с возможностями, синтаксисом, технологией языка Python и обучить методам программирования для решения прикладных математических и информационных задач, подготовить учащихся к олимпиаде по программированию.

### Задачи Программы:

Обучающие:

- сформировать представление об основах программирования в среде Python;
- сформировать навыки грамотной работы в системе программирования Python;
- ознакомить с базовыми понятиями теории алгоритмов при решении математических задач;
- обучить методам решения задач, реализуемым на языке Python;
- сформировать практические навыки решения прикладных задач;
- сформировать навыки поиска информации, работы с технической литературой.

Развивающие:

- развить навыки самостоятельного и творческого подхода к решению задач с помощью средств современной вычислительной техники;
- развить алгоритмическое и логическое мышление учащихся;
- развить творческие способности обучающихся, их потребность в самореализации;
- развить интеллектуальные и практические умения, самостоятельно приобретать и применять на практике полученные знания;

Воспитательные:

- содействовать воспитанию устойчивого интереса к изучению программирования;
- содействовать воспитанию информационной культуры;
- формировать потребность в творческой деятельности, стремление к самовыражению через техническое творчество;
- содействовать воспитанию интереса профессиям, связанным с программированием.

### Место учебного курса

Учебный курс реализуется за счет вариативного компонента, формируемого участниками образовательного процесса. Используется время, отведенное на внеурочную деятельность. Общий объем курса - 136 часов, 4 часа в неделю.

Данная Программа разработана на основе онлайн-курса алгоритмического программирования на языке Python 3 "Введение в программирование на языке Python" на сайте <https://online.sochisirius.ru>. Авторы курса — преподаватели ведущих школ и вузов,

действующие педагоги очных программ Центра «Сириус» Кириенко Д. П., Андреева Е. В., Крохина Д. А., Мамай И. Б.

Реализация Программы, основана на деятельностном подходе, более 70% времени отводится практической деятельности, способствующей развитию активной познавательной деятельности, творчества и достижению высоких результатов в области информационно-коммуникационных технологий.

Программа может быть реализована через дистанционное обучение – онлайн-занятия 2 раза в неделю по 2 часа. Программа включает в себя теоретические и практические занятия.

## КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

### Содержание программы

Название темы	Основное содержание по темам
1. Введение в язык программирования Python. (4 часа)	Знакомство с Python. Установка интерпретатора Python 3 и среды разработки Wing IDE 101. Первая программа. Философия программирования («Дзен Питона»). Основные арифметические операции. Регистрация на сайте «Школа программиста»
2. Переменные. Типы данных. (4 часа)	Переменные. Типы данных. Ввод и вывод данных. Обмен значений двух переменных.
3. Целочисленная арифметика. Линейные алгоритмы. (10 часов)	Операции целочисленного деления и взятия остатка. Решение задач.
4. Разветвляющиеся алгоритмы. (14 часов)	Условная конструкция (if). Полная и неполная форма ветвления. Операции сравнения. Тип bool. Логические операции. Вложенные условные конструкции. Решение задач.
5. Циклические алгоритмы. (26 часов)	Цикл с параметром (for). Функция range. Цикл while. Решение задач.
6. Строки. (10 часов)	Строки. Обращение к элементу строки по индексу. Срезы. Методы find, rfind, count, replace. Решение задач.
7. Списки. (20 часов)	Списки (массивы). Операции со списками. Создание списков. Методы split, join, map. Генераторы списков. Модуль random. Срезы. Решение задач.
8. Двумерные массивы. (20 часов)	Двумерные массивы (матрицы). Хранение, обработка и вывод двумерных списков. Копирование и создание двумерных списков. Считывание двумерных массивов. Вложенные генераторы. Решение задач.
9. Функции (8 часов)	Стандартные и пользовательские функции. Локальные и глобальные переменные. Решение задач
10. Символы (10 часов)	Обработка символов в Python. Кодовая таблица символов. Основные методы для работы с символами. Решение задач.
11. Рекурсия. (10 часов)	Рекурсия. Решение задач.

### Планируемые результаты

Основными предметными результатами, формируемыми при изучении учебного курса, являются:

- развитие алгоритмического мышления;
- развитие умений составить и записать алгоритм решения задачи;

- формирование знаний об алгоритмических конструкциях, логических значениях и операциях;
- знакомство с языком программирования Python и основными алгоритмическими структурами - линейной, условной и циклической;

Основными личностными результатами, формируемыми при изучении учебного курса, являются:

- наличие представлений об информации как важнейшем стратегическом ресурсе развития личности, государства, общества;
- понимание роли информационных процессов в современном мире;
- владение первичными навыками анализа и критичной оценки получаемой информации;
- ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;
- развитие чувства личной ответственности за качество окружающей информационной среды;
- способность увязать учебное содержание с собственным жизненным опытом, понять значимость подготовки в области информатики и ИКТ в условиях развития информационного общества;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики и ИКТ;
- способность и готовность к общению и сотрудничеству со сверстниками и взрослыми в процессе образовательной, общественно-полезной, учебно-исследовательской, творческой деятельности;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Основными метапредметными результатами, формируемыми при изучении учебного курса, являются:

- владение общепредметными понятиями «объект», «система», «модель», «алгоритм», «исполнитель» и др.;
- владение информационно-логическими умениями: определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;
- владение умениями самостоятельно планировать пути достижения целей; соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией; оценивать правильность выполнения учебной задачи;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы; поиск и выделение необходимой информации, применение методов информационного поиска; структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий; самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
- владение информационным моделированием как основным методом приобретения знаний: умение преобразовывать объект из чувственной формы в пространственно-

графическую или знаково-символическую модель; умение строить разнообразные информационные структуры для описания объектов; умение «читать» таблицы, графики, диаграммы, схемы и т. д., самостоятельно перекодировать информацию из одной знаковой системы в другую; умение выбирать форму представления информации в зависимости от стоящей задачи, проверять адекватность модели объекту и цели моделирования;

- ИКТ-компетентность — широкий спектр умений и навыков использования средств информационных и коммуникационных технологий для сбора, хранения, преобразования и передачи различных видов информации, навыки создания личного информационного пространства (обращение с устройствами ИКТ; фиксация изображений и звуков; создание письменных сообщений; создание графических объектов; создание музыкальных и звуковых сообщений; создание, восприятие и использование гипермедиа-сообщений; коммуникация и социальное взаимодействие; поиск и организация хранения информации; анализ информации).

## КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

### Учебный план

№	Название разделов	Количество часов			Форма промежуточной (итоговой) аттестации
		всего	теория	практика	
1	Введение в язык программирования Python.	4	1	3	Опрос, решение задач
2	Переменные. Типы данных. (4 часа)	4	1	3	Опрос, решение задач
3	Целочисленная арифметика. Линейные алгоритмы.	10	2	8	Опрос, решение задач
4	Разветвляющиеся алгоритмы. (14 часов)	14	2	12	Опрос, решение задач
5	Циклические алгоритмы. (26 часов)	26	4	22	Опрос, решение задач
6	Строки. (10 часов)	10	2	8	Опрос, решение задач
7	Списки. (20 часов)	20	4	16	Опрос, решение задач
8	Двумерные массивы. (20 часов)	20	4	16	Опрос, решение задач
9	Функции (8 часов)	8	2	6	Опрос, решение задач
10	Символы (10 часов)	10	2	8	Опрос, решение задач
11	Рекурсия. (10 часов)	10	2	8	Опрос, решение задач
	<i>Итого:</i>	136	26	110	

### Календарный учебный график

Раздел/месяц	сентябрь	октябрь	ноябрь	декабрь	январь	февраль	март	апрель	май
Раздел 1	<b>4 ч.</b>								

Раздел 2	4 ч.								
Раздел 3	8 ч.	2 ч.							
Раздел 4		14 ч.							
Раздел 5			16 ч.	10 ч.					
Раздел 6				6 ч.	4 ч.				
Раздел 7					10 ч.	10 ч.			
Раздел 8						6 ч.	12 ч.	2 ч.	
Раздел 9								8 ч.	
Раздел 10								6 ч.	4 ч.
Раздел 11									10 ч.
Всего:	16 ч.	16 ч.	16 ч.	16 ч.	14 ч.	16 ч.	12 ч.	16 ч.	14 ч.
Итого:	70 ч.								

### Календарно тематическое планирование

№	№ темы	№ урока по теме	Тема урока	Дата проведения		ЦОР
				план	факт	
<b>1. Введение в язык программирования Python. (4 часа)</b>						
1	1	1	Знакомство с Python. Установка интерпретатора Python 3 и среды разработки Wing IDE 101.	3.09		
2		2	Первая программа. Философия программирования («Дзен Питона»). Основные арифметические операции.	3.09		
3		3	Регистрация на сайте «Школа программиста». Другие ресурсы онлайн проверки решений задач.	5.09		
4		4	Решение задач	5.09		
<b>2. Переменные. Типы данных. (4 часа)</b>						
5	2	1	Переменные. Типы данных.	10.09		
6		2	Решение задач	10.09		
7		3	Ввод и вывод данных. Обмен значений двух переменных.	12.09		
8		4	Решение задач	12.09		
<b>3. Целочисленная арифметика. Линейные алгоритмы. (10 часов)</b>						
9	3	1	Операции целочисленного деления и взятия остатка.	17.09		
10		2	Задачи: «Батарейки», «Сумма цифр числа»	17.09		
11		3	Задачи: «Дележ яблок», «Настольный теннис»	19.09		
12		4	Задачи: «Предпоследняя цифра», «Электронные часы»	19.09		
13		5	Задачи: «Стоимость покупки», «Дележ яблок-2»	24.09		
14		6	Задачи: «Автопробег», «Парты»	24.09		
15		7	Задачи: «Страницы книги»,	26.09		

			«Шахматная доска»			
16		8	Задачи: «Следующее четное», «Симметричное число»	26.09		
17		9	Решение задач на сайте асmp.ru	1.10		
18		10	Решение задач на сайте асmp.ru	1.10		
<b>4. Разветвляющиеся алгоритмы. (14 часов)</b>						
19	4	1	Условная конструкция (if). Задача «Разрезание торта»	3.10		
20		2	Операции сравнения. Задача «Автопробег»	3.10		
21		3	Тип данных bool. Логические операции. Задача «Високосный год»	8.10		
22		4	Задачи «Какое число больше», «Шахматная доска-2»	8.10		
23		5	Задачи «Ход ладьи», «Ход слона»	10.10		
24		6	Задачи «Ход коня», «Яша плавает в бассейне»	10.10		
25		7	Задача «Узник замка Иф»	15.10		
26		8	Вложенные условные конструкции. Задача «Високосный год»	15.10		
27		9	Задачи «Максимум трех чисел», «Сколько совпадает чисел»	17.10		
28		10	Задачи «Упорядочить три числа», «Апельсины бочками»	17.10		
29		11	Задача «Линейное уравнение»	22.10		
30		12	Решение задач на сайте асmp.ru	22.10		
31		13	Решение задач на сайте асmp.ru	24.10		
32		14	Решение задач на сайте асmp.ru	24.10		
<b>5. Циклические алгоритмы. (26 часов)</b>						
33	5	1	Цикл с параметром. Функция range.	29.10		
34		2	Задачи «Ряд», «Ряд-2»	29.10		
35		3	Задачи «Цифры числа», «Диофантово уравнение»	5.11		
36		4	Задачи «Факториал», «Сумма квадратов»	5.11		
37		5	Задача «Остатки»	7.11		
38		6	Задача «Четные числа»	7.11		
39		7	Задачи «Числа, не делящиеся на 3», «Сумма n чисел», «Минимум из n чисел»	12.11		
40		8	Задачи «Количество нулей», «Делители»	12.11		
41		9	Задачи «Лесенка», «Сумма произведений соседних чисел»	14.11		
42		10	Задача «Сумма факториалов»	14.11		
43		11	Решение задач на сайте асmp.ru	19.11		
44		12	Решение задач на сайте асmp.ru	19.11		



45		13	Цикл while	21.11		
46		14	Задача «Трискайдекафобия (боязнь числа 13)»	21.11		
47		15	Задача «Номер числа Фибоначчи»	26.11		
48		16	Задача «Точная степень двойки»	26.11		
49		17	Задача «Банковские проценты»	28.11		
50		18	Задачи «Минимальный простой делитель», «Список квадратов»	28.11		
51		19	Обработка последовательностей неизвестной длины	3.12		
52		20	Задача «Второй минимум»	3.12		
53		21	Задачи «Количество элементов, которые больше предыдущего», «Количество четных элементов последовательности»	5.12		
54		22	Задачи «Количество локальных максимумов», «Среднее значение последовательности»	5.12		
55		23	Задача «Самое частое число в последовательности»	10.12		
56		24	Задача «Выдача сдачи»	10.12		
57		25	Решение задач на сайте acmp.ru	12.12		
58		26	Решение задач на сайте acmp.ru	12.12		
<b>6. Строки. (10 часов)</b>						
59	6	1	Строки - объекты типа str. Обращение по индексу. Задача «Количество букв "a" в строке»	17.12		
60		2	Задача «Значение выражения»	17.12		
61		3	Срезы. Задача «Делаем срезы»	19.12		
62		4	Задачи «Палиндром», «Сначала четные, потом нечетные»	19.12		
63		5	Методы find, rfind, count, replace	24.12		
64		6	Задача «Совпадающие начало и конец»	24.12		
65		7	Задачи «Замена подстроки», «Количество слов»	26.12		
66		8	Задачи «Замена внутри фрагмента», «Вставка символов»	26.12		
67		9	Решение задач на сайте acmp.ru	14.01		
68		10	Решение задач на сайте acmp.ru	14.01		
<b>7. Списки. (20 часов)</b>						
69	7	1	Списки (массивы). Создание списков	16.01		
70		2	Задачи «Список одинаковых чисел», «Список квадратов»	16.01		
71		3	Методы split и join	21.01		
72		4	Задачи «Четные элементы», «Четные индексы»	21.01		
73		5	Задача «Индекс первого вхождения заданного элемента в	23.01		

			массиве»			
74		6	Задачи «Больше своих соседей», «Соседи одного знака»	23.01		
75		7	Задачи «Наименьший положительный», «Количество различных элементов»	28.01		
76		8	Задачи «Ближайшее число», «Шеренга»	28.01		
77		9	Задачи «Циклический сдвиг вправо», «Два ближайших числа»	30.01		
78		10	Задача «Медиана»	30.01		
79		11	Генераторы списков. Модуль random.	4.02		
80		12	Считывание данных. Задача «Список степеней»	4.02		
81		13	Срезы. Задача «Переставить в обратном порядке»	6.02		
82		14	Задача «Развернуть элементы с нечетными индексами»	6.02		
83		15	Основные операции со списками. Задачи «Удалить элемент», «Вставить элемент»	11.02		
84		16	Задача «Количество различных элементов-2»	11.02		
85		17	Задачи «Уникальные элементы», «Самое частое число»	13.02		
86		18	Задачи «Количество совпадающих пар», «Переставить min и max»	13.02		
87		19	Решение задач на сайте acmp.ru	18.02		
88		20	Решение задач на сайте acmp.ru	18.02		
<b>8. Двумерные массивы. (20 часов)</b>						
89	8	1	Двумерные массивы (матрицы, список списков). Обработка и вывод списка.	20.02		
90		2	Копирование списков. Создание двумерных списков.	20.02		
91		3	Задача «Снежинка»	25.02		
92		4	Задача «Ход коня»	25.02		
93		5	Считывание двумерных массивов	27.02		
94		6	Задача «Поменять столбцы»	27.02		
95		7	Задача «Симметричен ли массив?»	4.03		
96		8	Задача «Максимум»	4.03		
97		9	Задача «Поменять местами две диагонали»	6.03		
98		10	Задача «Транспонировать квадратную матрицу»	6.03		
99		11	Задача «Треугольник Паскаля»	11.03		
100		12	Задача «Кинотеатр»	11.03		

101		13	Задача «Золото»	13.03		
102		14	Вложенные генераторы	13.03		
103		15	Задача «Диагонали, параллельные главной»	18.03		
104		16	Задача «Заполнение змейкой»	18.03		
105		17	Задачи «Шахматная доска - 3», «Слева направо, сверху вниз»	20.03		
106		18	Задача «Квадранты»	20.03		
107		19	Решение задач на сайте асmp.ru	25.03		
108		20	Решение задач на сайте асmp.ru	25.03		
<b>9. Функции (8 часов)</b>						
109	9	1	Стандартные функции в Python и функции пользователя. Задача «Треугольник»	27.03		
110		2	Задача «Елочка»	27.03		
111		3	Задача «На завод!»	1.04		
112		4	Глобальные и локальные переменные	1.04		
113		5	Тестовые задачи по теме «Глобальные и локальные переменные». Решение задач на сайте асmp.ru	3.04		
114		6	Решение задач на сайте асmp.ru	3.04		
115		7	Решение задач на сайте асmp.ru	8.04		
116		8	Решение задач на сайте асmp.ru	8.04		
<b>10. Символы (10 часов)</b>						
117	10	1	Сравнение строк. Кодовая таблица символов. Функция ord и chr	10.04		
118		2	Задачи «ASCII-код символа», «Символ с данным ASCII- кодом», «Символы в заданном интервале»	10.04		
119		3	Методы upper и lower. Задача «Сменить регистр символа»	15.04		
120		4	Задача «Слова с прописной буквы»	15.04		
121		5	Задача «Самая частая буква»	17.04		
122		6	Задача «Палиндром»	17.04		
123		7	Задача «Шифр Цезаря»	22.04		
124		8	Задача «Ход короля и шахматная нотация»	22.04		
125		9	Решение задач на сайте асmp.ru	24.04		
126		10	Решение задач на сайте асmp.ru	24.04		
<b>11. Рекурсия. (10 часов)</b>						
127	11	1	Рекурсивная функция	6.05		
128		2	Задача «Разворот последовательности»	6.05		
129		3	Задача «Быстрое возведение в степень»	13.05		
130		4	Задача «Рекурсивный перевод»	13.05		

131		5	Задача «Количество разбиений на слагаемые»	15.05		
132		6	Задача «Ханойские башни»	15.05		
133		7	Задача «Построение кривой дракона»	20.05		
134		8	Решение задач на сайте acmp.ru	20.05		
135		9	Решение задач на сайте acmp.ru	22.05		
136		10	Решение задач на сайте acmp.ru	22.05		

## ДОПОЛНИТЕЛЬНЫЕ МАТЕРИАЛЫ

### Условия реализации

Успешная реализация программы учебного курса "Python. Основы программирования" ориентирована на существующую информационно-образовательную среду школы. Информационно-образовательная среда образовательного учреждения включает: комплекс информационных образовательных ресурсов, в том числе цифровые образовательные ресурсы, совокупность технологических средств информационных и коммуникационных технологий: компьютеры, иное ИКТ-оборудование, коммуникационные каналы.

### Список литературы

Поляков К.Ю. - Программирование. Python. C++. 2019. Издательство БИНОМ

Интернет-ресурсы:

<https://online.sochisirius.ru> онлайн-курс алгоритмического программирования

<https://acmp.ru/> - Школа программиста

<https://informatics.msk.ru/> - дистанционная подготовка по информатике

<http://olymp.isu.ru/> - сервер спортивного программирования ИГУ

<https://stepik.org/> - онлайн-курс "Программирование на Python"

## ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

### Тема 2: Переменные. Типы данных

Если в вашей программе присутствуют большие вычисления, то неудобно их записывать в виде одного длинного выражения, лучше вычислять его значения по частям, используя переменные для хранения промежуточных результатов.

**Переменная** — это некоторая ячейка в памяти, в которой хранится какое-то значение любого из возможных типов, например, число или строка. В переменную можно записать какое-то значение (это называется **присвоить переменной значение**). Для присваивания значения переменной используется операция `=`, где слева от знака присваивания записывается имя переменной, справа — присваиваемое значение, которое может быть числом, строкой, другой переменной, каким-то выражением. Значение переменной можно использовать при вычислениях или вывести его на экран при помощи функции `print()`.

### Пример

Пусть длина школьного урока хранится в переменной `lesson_time`, продолжительность перемены между уроками — `rest_time`, а количество уроков — `n`. Хотим вычислить сколько времени занимает заданное количество уроков, которое будет храниться в переменной

total\_time.

```
>>> lesson_time = 45
>>> rest_time = 15
>>> n = 6
>>> total_time = lesson_time * n + (n - 1) * rest_time
>>> print(total_time)
345
```

Имя переменной (также называется **идентификатором**) — последовательность букв, цифр и символа подчёркивания, первым символом должна быть буква или символ подчёркивания. В языке Python допускаются буквы не латинского алфавита в качестве имён переменных, но следует ограничиваться только латинскими символами.

Имена переменных должны быть осмысленными и понятными, например, lesson\_time или rest\_time.

### Пример

```
>>> a = 14
>>> v = a
>>> a = 7
>>> a = a + 3
>>> v = a + 3
>>> print(a, v)
10 13
```

Посмотрим, как работает данная программа. В первой строке переменной `a` присваивается значение `14`, затем переменной `v` присваивается значение переменной `a`, то есть `14`. При этом переменные существуют независимо друг от друга, поэтому при выполнении присваивания `a = 7` значение переменной `v` не изменится.

При выполнении присваивания `a = a + 3` вычисляется сначала правая часть выражения, равная в данном случае `10`, после чего это значение присваивается переменной. При выполнении присваивания `v = a + 3`, аналогично вычисляется сначала правая часть выражения. В данном случае в переменной `a` хранится значение, равное `10`, значит, правая часть выражения будет равна `13`. Именно это значение и будет храниться после присваивания в переменной `v`.

После этого значения переменных `a` и `v` можно вывести на экран при помощи функции `print(a, v)`, которую мы позже рассмотрим подробнее.

Решите задачи по теме:

10. Не запуская код, ответьте на вопрос: что выведет на экран данная программа?

```
a = 5
b = a * 10
a = a + b + 4
print(a + b)
```

11. Не запуская код, ответьте на вопрос: что выведет на экран данная программа?

```
a = 3
a = a * 5 + a ** 2
print(a)
```

## Продолжение темы 2: Переменные. Типы данных

### Типы данных

Если мы присваиваем переменной `a` значение, равное `5`, то переменная будет иметь значение типа `int`, который используется для целых чисел. При умножении, вычитании и сложении чисел этого типа мы получаем числа того же типа.

Мы уже сталкивались с тем, что при выполнении операции деления результатом будет дробное число. Если переменной `b` присвоить значение `10 / 2`, при выводе на экран результата увидим число `5.0`, которое равно числу `5`, но имеет другой тип. Переменная `b` имеет значение типа `float`. Это специальный тип данных для представления действительных чисел, и при умножении, вычитании и сложении чисел этого типа мы получаем числа того же типа.

Еще один тип данных используется для работы со строками. Если мы присваиваем переменной `s` значение, равное `'Hello'`, то эта переменная будет иметь значение типа `str`. Строки тоже можно складывать. Однако операция сложения для целых чисел и для строк работает по-разному: для чисел это сложение, а для строк — **конкатенация**.

### Основные операции над строками:

`A + B` — конкатенация, то есть соединение строк путём размещения строки `B` сразу после строки `A`;

`A * n` — повторение строки `n` раз, значение `n` должно быть целого типа.

### Преобразование типов

Иногда полезно целое число записать как строку. И, наоборот, если строка состоит из цифр, то полезно эту строку представить в виде числа, чтобы дальше можно было выполнять арифметические операции с ней. Для этого используются функции, имя которых совпадает с именем типа, то есть `int`, `float`, `str`. Например, `int('123')` вернёт целое число `123`, а `str(123)` вернёт строку `'123'`.

**Пример:**

```
>>>n = 5
>>>price = 15
>>>s = str(n) + " ручек стоят " + str(n * price) + " рублей"
>>>print(s)
5 ручек стоят 75 рублей
```

**Пример:**

```
>>>print(str(1 + 1) * int('1' + '1'))
2222222222
```

Результатом будет строка '2', повторенная 11 раз.

Решите задачи по теме:

12. Напишите программу, которая выводит слово 'Python' 1234 раза подряд (только буквы, без кавычек).

13. Квадраты чисел из одних единиц выглядят красиво:  
 $11^2 = 121$ ,  $111^2 = 12321$ ,  $1111^2 = 1234321$ .

Напишите программу, которая будет выводить на экран число, равное квадрату числа, записанного при помощи 123 единиц.

14. Число  $15^{10}$  записали 10 раз подряд. Полученное число возвели в квадрат.

Напишите программу, которая повторит указанные действия и выведет полученный результат.

Продолжение темы 2: Переменные. Типы данных**Ввод данных: функция `input()`**

Все ранее рассмотренные программы выдавали один и тот же результат при каждом своем запуске. Для изменения входных данных нужно было вручную изменять значения переменных. Этого можно избежать, считывая данные с клавиатуры.

Для считывания строки с клавиатуры используется функция `input()`, которая возвращает значение считанной строки. Например, можно считать значения двух строк следующим образом:

```
a = input()
b = input()
```

Правда, функция `input()` возвращает текстовую строку, а нам бывает нужно сделать так, чтобы переменные имели целочисленные значения. Поэтому сразу же после считывания выполним преобразование типов при помощи функции `int` и запишем новые значения в переменные `a` и `b`.

```
a = int(a)
b = int(b)
```

Можно объединить считывание строк и преобразование типов, если вызывать функцию `int` для того значения, которое вернёт функция `input()`:

```
a = int(input())
b = int(input())
```

Далее в программе можно использовать значения переменных `a` и `b` для вычисления новых переменных или для вывода результата на экран. Теперь мы можем, не меняя исходного кода программы, многократно использовать её для решения различных задач. Для этого нужно запустить программу и после запуска ввести с клавиатуры два числа, нажимая после каждого клавишу *Enter*. Затем программа сама выведет результат.

### Вывод данных: функция `print()`

Функция `print()` может выводить не только значения констант или переменных, но и значения выражений. Например, допустима такая запись:

```
print(a + b)
```

#### Решите задачи по теме:

15. Сумма.
16. Обмен значений – 1.
17. Обмен значений – 2.
18. Следующее и предыдущее.

Задачи с сайта [acmp.ru](http://acmp.ru): 108, 903, 1, 195, 773, 33, 4, 819, 756, 529, 312