

**Министерство образования и науки Смоленской области  
Муниципальное бюджетное общеобразовательное учреждение  
«Средняя общеобразовательная школа № 2» г. Сафоново  
Смоленской области**

Принята на заседании  
педагогического совета  
Протокол № 1 от 29.08.2024 г.



**Дополнительная общеобразовательная общеразвивающая  
программа технической направленности  
«Язык программирования Python с нуля до первых проектов»**

**Срок реализации – 1 год  
Возраст детей – 13 - 16 лет**

Программа составлена педагогом  
дополнительного образования  
Егоровой Мариной Евгеньевной

г. Сафоново, 2024 г.

## **ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

### **Программа разработана на основе:**

1. Федерального закона «Об образовании в «РФ» от 29 декабря 2012 г. № 273-ФЗ;
2. Концепции развития дополнительного образования детей до 2030 года;
3. Приказа Минпросвещения России от 27.07.2022. № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
4. Методических рекомендаций по проектированию дополнительных общеразвивающих программ (Письмо Минобрнауки РФ «О направлении информации» от 18 ноября 2015 г. № 09- 3242);
5. Письма Минпросвещения России от 31.01.2022 № ДГ-245/06 "О направлении методических рекомендаций" (вместе с "Методическими рекомендациями по реализации дополнительных общеобразовательных программ с применением электронного обучения и дистанционных образовательных технологий");
6. Постановления главного государственного санитарного врача от 28.09.2020 № 28 СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи»;
7. Устава муниципального бюджетного общеобразовательного учреждения «МБОУ «СОШ № 2» г. Смоленской области.

### **Направленность программы:** техническая.

**Актуальность программы.** Навыки программирования широко востребованы в современном обществе. Владение базовыми понятиями из этой сферы необходимо для взаимодействия со специалистами смежных областей и использования различного программного обеспечения. Язык Python имеет очень простую «точку входа», именно поэтому большинство современных образовательных программ начинают изучение программирования именно с Python. Программа курса погружает учеников в мир программирования в интересной и увлекательной форме, вызывая интерес к сфере информационных технологий

### **Педагогическая целесообразность программы.**

Кружок «Язык программирования Python с нуля до первых проектов» для учащихся 9 классов, как технического направления, так и гуманитарного. Программа кружка неразрывно сочетает теоретическую подготовку и освоение практических приемов работы.

**Адресат программы:** Программа рассчитана для обучающихся 13-16 лет, проявляющим интерес к современным компьютерным технологиям, обладающих навыками работы в среде Windows, умение работать со средами программирования Python.

**Наполняемость групп** до 15 человек. Срок освоения программы: 1 год. Объем программы: 68 академических часов.

**Режим занятий:** 2 академических часа в неделю.

**Форма организации образовательного процесса:** очная форма.

**Условия реализации программы**

*Техническое и программное обеспечение образовательного процесса:*

Персональный компьютер (ноутбук) с процессором класса Intel Core i3-i5 с тактовой частотой не ниже 3 ГГц, оперативной памятью не ниже 4Гб, объем жёсткого диска не менее 500 ГБ и доступом к сети Интернет. Программное обеспечение: среда программирования Python.

**Формы проведения занятий:** беседа, лекция, компьютерный практикум.

Программа включает в себя две части: теоретическую и практическую. Теоретическая часть организована в форме лекций. Лекции проводятся с обязательным использованием иллюстративных материалов. Практическая часть - в форме самостоятельных практических заданий и проектов, что является важной составляющей всего курса.

**Цель программы:** Цель образовательной программы — создание условий для получения знаний, умений и навыков в области программирования на языке Python; знакомство с основными направлениями информационных технологий; закрепление изученного материала с помощью творческих и проектных заданий.

**Задачи:**

*Задачи обучения*

- формирование и развитие навыков алгоритмического и логического мышления, грамотной разработки программ;
- формирование навыков процедурного программирования;
- изучение конструкций языка программирования Python;
- знакомство с основными структурами данных и типовыми методами обработки этих структур;
- приобретение навыков поиска информации в интернете, анализ выбранной информации на соответствие запросу, использование информации при решении задач;
- знакомство с основными направлениями информационных технологий.

*Задачи развития*

- развитие у обучающихся интереса к информационным технологиям;
- формирование самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;
- расширение кругозора обучающихся и профессиональное ориентирование в области информационных технологий.

*Задачи воспитания*

- воспитание упорства в достижении результата;

- подготовка к самостоятельной профессиональной деятельности;
- создание условий для социализации и саморазвития личности обучающихся.

### **Планируемые результаты обучения**

По окончании курса ученик приобретает следующие компетенции:

- знает основы языка программирования Python;
- умеет объяснять и использовать на практике как простые, так и сложные структуры данных и конструкции для работы с ними;
- умеет искать и обрабатывать ошибки в коде;
- умеет разбивать решение задачи на подзадачи;
- способен писать грамотный, красивый код;
- способен анализировать как свой, так и чужой код;
- понимает основы представления, кодирования, хранения и передачи информации, логических законов построения компьютеров;
- способен работать с информацией: находить, оценивать и использовать информацию из различных источников, необходимую для решения профессиональных задач (в том числе на основе системного подхода);
- способен грамотно строить коммуникацию, исходя из целей и ситуации

### **Универсальные коммуникативные действия**

#### *Общение:*

- сопоставлять свои суждения с суждениями других участников диалога, обнаруживать различие и сходство позиций;

- публично представлять результаты выполненного опыта (исследования, проекта);

- выбирать формат выступления с учётом задач презентации и особенностей аудитории и в соответствии с ним составлять устные и письменные тексты с использованием иллюстративных материалов.

#### *Совместная деятельность (сотрудничество):*

- понимать и использовать преимущества командной и индивидуальной работы при решении конкретной проблемы, в том числе при создании информационного продукта;

- принимать цель совместной информационной деятельности по сбору, обработке, передаче, формализации информации; коллективно строить действия по её достижению: распределять роли, договариваться, обсуждать процесс и результат совместной работы;

- выполнять свою часть работы с информацией или информационным продуктом, достигая качественного результата по своему направлению и координируя свои действия с другими членами команды;

- оценивать качество своего вклада в общий информационный продукт по критериям, самостоятельно сформулированным участниками взаимодействия;

- сравнивать результаты с исходной задачей и вклад каждого члена команды в достижение результатов, разделять сферу ответственности и проявлять готовность к предоставлению отчёта перед группой.

### **Универсальные регулятивные действия.**

#### *Самоорганизация:*

- выявлять в жизненных и учебных ситуациях проблемы, требующие решения;

- составлять алгоритм решения задачи (или его часть), выбирать способ решения учебной задачи с учётом имеющихся ресурсов и собственных возможностей, аргументировать выбор варианта решения задачи;

- составлять план действий (план реализации намеченного алгоритма решения), корректировать предложенный алгоритм с учётом получения новых знаний об изучаемом объекте.

#### *Самоконтроль (рефлексия):*

- владеть способами самоконтроля, самомотивации и рефлексии;

- учитывать контекст и предвидеть трудности, которые могут возникнуть при решении учебной задачи, адаптировать решение к меняющимся обстоятельствам;

- вносить коррективы в деятельность на основе новых обстоятельств, изменившихся ситуаций, установленных ошибок, возникших трудностей;

- оценивать соответствие результата цели и условиям.

#### *Эмоциональный интеллект:*

- ставить себя на место другого человека, понимать мотивы и намерения другого.

#### *Принятие себя и других:*

- осознавать невозможность контролировать всё вокруг даже в условиях открытого доступа к любым объёмам информации;

- осознанно относиться к другому человеку, его мнению.

**Уровни программы:** начальный.

### **Уровни оценки результатов освоения программы:**

– *максимальный* (учащийся знает от 80-100% теоретического материала программы, успешно без помощи педагога выполняет все практические и творческие задания, участвует в конкурсах и выставках);

– *средний* (учащийся знает от 50-80% теоретического материала программы, выполняет все практические и творческие задания самостоятельно, в редких случаях прибегает к помощи педагога, участвует в конкурсах и выставках в рамках творческого объединения);

- *минимальный* (учащийся знает менее 50% теоретического материала программы, выполняет все практические и творческие задания самостоятельно, но часто прибегает к помощи педагога, редко участвует в конкурсах и выставках в рамках творческого объединения)

**Условия реализации программы.** Необходима математическая подготовка на уровне 7 класса общеобразовательной школы, умение программировать не требуется.

**Формы аттестации и контроля:** тестирование и выполнение проектной работы

*Диагностические инструменты*

Проектная работа оценивается по известным учителю и ученику критериям. В середине проектной работы происходит предварительная оценка, далее ученику предлагается доработать проект. Тестирование может содержать закрытые тестовые вопросы и задачи на написание кода с автопроверкой.

*Показатели и критерии оценивания*

В каждом модуле ведется подсчет рейтинга: задачи тестирования приносят 5 баллов, проекты – 6 баллов. Итого 11 баллов рейтинга в каждом модуле.

Проходным баллом считается 7 баллов за каждый модуль.

*Шкала оценивания, нижнее значение*

0 баллов, если ни одна из задач тестирования не решена, ни один из базовых критериев не выполнен.

*Шкала оценивания, верхнее значение*

11 баллов за полное решение всех задач в тестировании и выполнение всех базовых критериев.

*Шкала оценивания, минимальный проходной балл*

7 баллов – выполнены часть базовых критериев, выполнена часть задач тестирования.

Проектная работа оценивается по известным учителю и ученику критериям. В середине проектной работы происходит предварительная оценка, далее ученику предлагается доработать проект.

## Учебный план

№ п/п	Модуль	Тема	Количество часов		Форма организации	Форма аттестации и контроля
			Теория	Практика		
1	Модуль 1. Линейные и разветвленные алгоритмы	Тема 1. Знакомство с Python. Команды input() и print()	2	2	Лекция с элементами беседы. Практикум	тестирование и выполнение проектной работы
		Тема 2. Параметры sep, end. Переменные. Комментарии	2	2		
		Тема 3. Целочисленная арифметика в Python. Решение задач	2	2		
		Тема 4. Условный оператор. Логические операции and, or, not	2	1		
		Тема 5. Проектная работа «Калькулятор»	0	2		
2	Модуль 2. Циклические алгоритмы.	Тема 1. Повторение пройденного. Вложенные условия	2	2	Лекция с элементами беседы. Практикум	тестирование и выполнение проектной работы
		Тема 2. Типы данных int, float, str. Функции min(), max(), abs()	2	2		
		Тема 3. Циклические алгоритмы. Цикл for. Функция range()	2	2		
		Тема 4. Задачи с циклами. Цикл while. Операторы break, continue	2	1		
		Тема 5. Проектные работы «Продвинутый калькулятор», «Случайный фильм», «Плейлист»	0	2		
3	Модуль 3. Основные структуры данных	Тема 1. Повторение пройденного. Задачи с условиями, циклами	2	2	Лекция с элементами беседы. Практикум	тестирование и выполнение проектной работы
		Тема 2. Работа со строками. Индексация, срезы, методы строк	2	2		

		Тема 3.Основы работы со списками. Вывод элементов списка	2	2		
		Тема 4.Методы списков. Списочные выражения. Решение задач	2	1		
		Тема 5.Проектные работы «Персональный помощник», «Быки и коровы», «Текстовый квест»	0	2		
4	Модуль 4. Функции.	Тема 1. Повторение пройденного. Задачи со спискам	2	2	Лекция с элементами беседы. Практикум	тестирование и выполнение проектной работы
		Тема 2.Функции. Локальные и глобальные переменные	2	2		
		Тема 3.Как функции упрощают код? Решение задач	2	2		
		Тема 4.Генерация случайных чисел. Модуль random	1	2		
		Тема 5.Проектная работа «Генератор сложных паролей»	0	2		
Итого:			28ч	40ч		
Всего:			68 часов			

## Содержание учебного плана

### Модуль 1 *Линейные и разветвленные алгоритмы (17ч)*

#### **Тема 1.** Знакомство с Python. Команды input() и print() (2+2)

Алгоритмы и блок-схемы, знакомство с Python, ввод и вывод данных. Решение задач.

*Формы контроля:* тестирование.

#### **Тема 2.** Параметры sep, end. Переменные. Комментарии (2+2)

Параметры sep и end, переменные и ключевые слова, комментарии. Решение задач.

*Формы контроля:* тестирование.

#### **Тема 3.** Целочисленная арифметика в Python. Решение задач (2+2)

Целые числа и строки, операции с ними, разбор типичных ошибок в коде. Решение задач.

*Формы контроля:* тестирование.

#### **Тема 4.** Условный оператор. Логические операции and, or, not(1+2)

Условный оператор, инструкция if-else, логические операции and, or, not. Решение задач.

*Формы контроля:* тестирование.

#### **Тема 5.** Проектная работа «Калькулятор» (0+2)

Решение задач. Проект и тест.

*Формы контроля:* тестирование и выполнение проектной работы.

### Модуль 2 *Циклические алгоритмы (17ч)*

#### **Тема 1.** Повторение пройденного. Вложенные условия(2+2)

Повторение, вложенные условия. Решение задач.

*Формы контроля:* тестирование.

#### **Тема 2.** Типы данных int, float, str. Функции min(), max(), abs()(2+2)

Операции с типами данных int, float, str; функции min(), max(), abs(). Решение задач.

*Формы контроля:* тестирование.

#### **Тема 3.** Циклические алгоритмы. Цикл for. Функция range()(2+2)

Циклические алгоритмы, цикл for, функция range() и её параметры. Решение задач.

*Формы контроля:* тестирование.

**Тема 4.** Задачи с циклами. Цикл while. Операторы break, continue(1+2)  
Цикл while; операторы break, continue. Решение задач.

*Формы контроля:* тестирование.

**Тема 5.** Проектные работы «Продвинутый калькулятор», «Случайный фильм», «Плейлист». (0+2)  
Решение задач. Проект и тест.

*Формы контроля:* тестирование и выполнение проектной работы.

### **Модуль 3 Основные структуры данных (17ч)**

**Тема 1.** Повторение пройденного. Задачи с условиями, циклами(2+2)  
Повторение, задачи с условиями, циклами. Решение задач.

*Формы контроля:* тестирование.

**Тема 2.** Работа со строками. Индексация, срезы, методы строк(2+2)  
Индексация, срезы, методы строк. Решение задач.

*Формы контроля:* тестирование.

**Тема 3.** Основы работы со списками. Вывод элементов списка(2+2)  
Списки, вывод элементов списка. Решение задач.

*Формы контроля:* тестирование.

**Тема 4.** Методы списков. Списочные выражения. Решение задач(1+2)  
Методы списков, списочные выражения. Решение задач.

*Формы контроля:* тестирование.

**Тема 5.** Проектные работы «Персональный помощник», «Быки и коровы», «Текстовый квест» (0+2)  
Решение задач. Проект и тест.

*Формы контроля:* тестирование и выполнение проектной работы.

### **Модуль 4 Функции (17ч)**

**Тема 1.** Повторение пройденного. Задачи со списками. (2+2)  
Повторение, задачи со списками. Решение задач.

*Формы контроля:* тестирование.

**Тема 2.** Функции. Локальные и глобальные переменные(2+2)

Функции, аргументы и параметры, локальные и глобальные переменные. Решение задач.

*Формы контроля:* тестирование.

**Тема 3.** Как функции упрощают код? Решение задач(2+2)

Применение разных функций. Решение задач.

*Формы контроля:* тестирование.

**Тема 4.** Генерация случайных чисел. Модуль random(1+2)

Генерация случайных чисел, модуль random. Решение задач.

*Формы контроля:* тестирование.

**Тема 5.** Проектная работа «Генератор сложных паролей» (0+2)

Решение задач. Проект и тест.

*Формы контроля:* тестирование и выполнение проектной работы.

## Календарный учебный график

№ п/п	Месяц, число, время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
<b>Модуль 1</b>						
1.		теория	1	Знакомство с курсом, алгоритмы и блок-схемы	Кабинет «Точка роста»	тестирование
2.		теория	1	Философия Python, вывод данных, команда print()		тестирование
3.		теория	1	Ввод данных, команда input(), параметры команды print()		тестирование
4.		теория	1	Переменные и ключевые слова, комментарии		тестирование
5.		практика	1	Целые числа и строки, операции +, -, *, /		тестирование
6.		практика	1	Ошибки в коде, решение задач		тестирование
7.		практика	1	Дополнительные операции **, //, %, решение задач		тестирование
8.		практика	1	Алгоритм получения цифр числа, решение задач		тестирование
9.		теория	1	Условный оператор, инструкция if-else		тестирование
10.		практика	1	Условный оператор, решение задач		тестирование
11.		теория	1	Логические операторы and, or, not		тестирование
12.		теория	1	Логические операторы and, or, not		тестирование
13.		практика	1	Составные условия, решение задач		тестирование
14.		практика	1	Составные условия, решение задач		тестирование
15.		практика	1	Составные условия, решение задач		тестирование
16.		практика	1	Работа над проектом		Оценка проекта по критериям
17.		практика	1	Урок-тест		тестирование

Модуль 2						
18.		теория	1	Повторение	Кабинет «Точка роста»	тестирование
19.		теория	1	Вложенные условия		тестирование
20.		теория	1	Типы данных int, float, str		тестирование
21.		теория	1	Функции min(), max(), abs()		тестирование
22.		практика	1	Функции min(), max(), abs()		тестирование
23.		теория	1	Циклические алгоритмы		тестирование
24.		теория	1	Цикл for. Функция range()		тестирование
25.		теория	1	Функция range()		тестирование
26.		практика	1	Функция range()		тестирование
27.		практика	1	Решение задач		тестирование
28.		практика	1	Решение задач		тестирование
29.		теория	1	Цикл while		тестирование
30.		практика	1	Решение задач		тестирование
31.		практика	1	Операторы break, continue		тестирование
32.		практика	1	Решение задач	тестирование	
33.		практика	1	Работа над проектом	Оценка проекта по критериям	
34.		практика	1	Урок-тест	тестирование	
Модуль 3						
35.		теория	1	Повторение	Кабинет «Точка роста»	тестирование
36.		теория	1	Задачи с условиями, циклами		тестирование
37.		теория	1	Работа со строками		тестирование
38.		теория	1	Индексация, срезы, методы строк		тестирование
39.		практика	1	Решение задач		тестирование

40.		практика	1	Решение задач		тестирование
41.		теория	1	Основы работы со списками		тестирование
42.		практика	1	Вывод элементов списка		тестирование
43.		практика	1	Решение задач		тестирование
44.		практика	1	Решение задач		тестирование
45.		теория	1	Методы списков		тестирование
46.		теория	1	Списочные выражения		тестирование
47.		теория	1	Списочные выражения		тестирование
48.		практика	1	Решение задач		тестирование
49.		практика	1	Решение задач		тестирование
50.		практика	1	Работа над проектом		Оценка проекта по критериям
51.		практика	1	Урок-тест		тестирование
<b>Модуль 4</b>						
52.		теория	1	Повторение	Кабинет «Точка роста»	тестирование
53.		практика	1	Решение задач		тестирование
54.		теория	1	Функции		тестирование
55.		теория	1	Функции		тестирование
56.		теория	1	Локальные и глобальные переменные		тестирование
57.		теория	1	Локальные и глобальные переменные		тестирование
58.		практика	1	Решение задач		тестирование
59.		практика	1	Решение задач		тестирование
60.		теория	1	Как функции упрощают код?		тестирование
61.		теория	1	Как функции упрощают код?		тестирование
62.		практика	1	Решение задач		тестирование
63.		практика	1	Решение задач		тестирование

64.			1	Генерация случайных чисел. Модуль random		тестирование
65.		практика практика	1	Решение задач		тестирование
66.		практика	1	Решение задач		тестирование
67.		практика	1	Работа над проектом		Оценка проекта по критериям
68.		практика	1	Урок-тест		тестирование

## Учебно-методические материалы

### *Методы, формы и технологии*

При реализации программы применяются следующие педагогические технологии

- информационно-коммуникативные;
- деятельностные;
- проектные с элементами исследовательской деятельности;
- дифференцированные (индивидуальная траектория обучения);
- модульное обучение

### *Используются следующие методы и формы преподавания*

- наглядные;– словесные;– с применением технических средств;– практические;– проблемные

### *Методические разработки*

Онлайн-учебник; учебные задачи базового уровня сложности для отработки элементов содержания изучаемой темы; дополнительные задачи повышенного уровня сложности для придания вариативности обучению учащихся с разным уровнем подготовки и разной скоростью усвоения материала.

### *Материалы модуля*

В каждом модуле есть проектная работа. Для выполнения проектной работы ученику предлагается обратиться к пройденным темам и на основе полученных знаний при поддержке преподавателя написать работающую программу, соответствующую определенному техническому заданию. Преподаватель объясняет ученику теоретический материал, необходимый для выполнения проекта.

Завершает каждый модуль тестирование, на котором учащиеся решают 5 задач с закрытыми тестами и написанием кода с автопроверкой. Это контрольное мероприятие, позволяющее оценить уровень усвоения материала модуля. Примеры учебных задач и проектов представлены в разделе «Примеры контрольных заданий»

### *Учебная литература*

Онлайн учебник с изложением теоретического материала урока, примерами решения типичных задач по изучаемой теме и элементов, на которые нужно обратить особое внимание.

## Календарный план воспитательной работы

№ п/п	Мероприятие	Цель	Дата проведения
1	Открытка для мамы	Воспитание чувства любви и уважения к своим мамам.	По расписанию
2	Участие в конкурсах по компьютерной графике (федеральных, региональных, районных)	Развитие интереса к избранному виду деятельности; упорство в достижении цели (участие в конкурсах и олимпиадах)	В течение года

## **Формы аттестации**

Аттестация происходит по итогам выполнения проектной работы и решения заданий

тестирования. В каждом модуле ведется подсчет рейтинга: задачи тестирования приносят 5

баллов, проекты – 6 баллов. Итого – 11 баллов рейтинга в каждом модуле.

Проходным

баллом считается 7 баллов за каждый модуль.

## **Оценочные материалы**

### **Показатели и критерии оценивания**

Учитель оценивает проектную работу по известным ему и ученику критериям. В середине

проектной работы происходит предварительная оценка, далее ученику предлагается

доработать проект.

Задачи контрольного урока могут содержать закрытые тестовые вопросы и задачи на

написание кода с автопроверкой.

### **Контрольные задания**

#### **Проектная работа по итогам первого модуля**

Проект «Калькулятор».

Критерии:

базовые:

– при запуске программа описывает свой функционал;

– функционал программы включает сложение, вычитание, умножение, деление (целочисленное и с остатком);

– код сопровождается комментариями, описывающими назначение отдельных частей

программы;

продвинутые:

– функционал программы включает конвертер мер и весов.

#### **Проектная работа по итогам второго модуля**

Проект «Продвинутый калькулятор»

Критерии:

базовые:

– при запуске программа описывает свой функционал;

– функционал программы включает сложение, вычитание, умножение, деление (целочисленное и с остатком), возведение в степень, сохранение ответа для

дальнейшего

использования;

– код сопровождается комментариями, описывающими назначение отдельных частей

программы;

продвинутые:

– функционал программы включает в себя расчет доходности вклада;

– функционал программы включает в себя перевод из различных систем счисления.

### **Проектная работа по итогам третьего модуля**

Проект «Персональный помощник»

Критерии:

базовые:

– при запуске программа описывает свой функционал;

– функционал программы включает приветствие пользователя, выполнение не менее 3

различных сценариев;

– код сопровождается комментариями, описывающими назначение отдельных частей

программы;

продвинутые:

– функционал программы включает в себя не менее 10 различных сценариев;

– функционал программы включает в себя проект «Продвинутый калькулятор».

### **Проектная работа по итогам четвертого модуля**

Проект «Генератор сложных паролей»

Критерии:

базовые:

– при запуске программа описывает свой функционал;

– функционал программы включает приветствие пользователя, создание пароля из 10

случайных символов, среди которых присутствуют большие и маленькие латинские

буквы, цифры и специальные символы;

– код сопровождается комментариями, описывающими назначение отдельных частей

программы;

продвинутые:

– функционал программы включает в себя проверку защищённости пароля.

### **Примеры тестовых вопросов**

В каком фрагменте кода отступы поставлены корректно?

1.

```
if i % 3 == 0:
```

```
print(i, 'кратно 3')
```

```
else:
```

```
print(i, 'не кратно 3')
```

2.

```
if i > 0:
print(i, 'положительное')
else:
print(i, 'не положительное')
3.
if i % 5 == 0:
print(i, 'кратно 5')
else:
print(i, 'не кратно 5')
```

Что напечатает эта программа?

```
s = '0'
s = s + '1'
s = s + '0'
print(s)
```

Примеры задач на написание кода с автопроверкой

Напиши программу для нахождения цифр четырехзначного числа. Программа должна

вывести текст в соответствии с условием задачи.

**Пример 1.** Пользователь ввёл 1234.

Программа должна вывести:

Тысяч: 1

Сотен: 2

Десятков: 3

Единиц: 4

**Пример 2.** Пользователь ввёл 5678.

Программа должна вывести:

Тысяч: 5

Сотен: 6

Десятков: 7

Единиц: 8

Напиши программу для определения количества цифр в строке.

**Пример 1.** Пользователь ввёл Абракадабра67

Программа должна вывести:

2

**Пример 2.** Пользователь ввёл УРА

Программа должна вывести:

0