МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВВГУ»)

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА ДАННЫХ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

По дисциплине

«Информатика и программирование»

Студент

гр. БИС-25-3 Д.С. Малых

Ассистент

преподавателя М.В. Водяницкий

Владивосток 2025

Задание

Выполнить задания на Python и оформить отчет по стандартам ВВГУ.

***Задание 1.*** Написать программу, которая определяет, как будет вести себя кондиционер. Если температура в помещении 20 градусов и выше, то кондиционер выключается, если меньше – включается. Температура должна вводиться пользователем с консоли.

Пример:

Введите температуру: 18

Кондиционер включен

***Задание 2.*** Год делится на четыре сезона: зима, весна, лето и осень. Написать программу, которая запрашивает у пользователя номер месяца и выводит, к какому сезону этот месяц относится.

Пример:

Введите номер месяца: 4

Это весна

***Задание 3.*** Считается, что один год, прожитый собакой, эквивалентен семи человеческим годам. При этом зачастую не учитывается, что собаки становятся абсолютно взрослыми уже к двум годам. Таким образом, многие предпочитают каждый из первых двух лет жизни собаки приравнивать к 10.5 годам человеческой жизни, а все последующие к 4.

Написать программу, которая будет переводить собачий возраст в человеческий. Программа должна корректно обрабатывать входные данные и выводить соответствующие сообщения об ошибках:

* Если вводится не число
* Если вводится число меньше 1
* Если вводится число большее 22

Пример:

Введите возраст собаки (в годах): 5

Возраст собаки в человеческих годах: 33.0

Пример:

Введите возраст собаки (в годах): 0

Ошибка: возраст должен быть не меньше 1

***Задание 4.*** Число делиться на 6 только в случае соблюдения двух условий:

* Последняя цифра четная
* Сумма всех цифр делиться на 3

Написать программу, которая выведет делиться ли введенное число на 6 или нет.

***Задание 5.*** Написать программу, которая будет проверять пароль на надежность. Пароль считается надежным, если его длина не менее 8 символов и если он содержит:

* Заглавные буквы латиницы
* Строчные буквы латиницы
* Числа
* Специальные знаки

В случае, если пароль не проходит по одному из условий, необходимо сообщить пользователю каким именно условиям он не удовлетворяет.

Пример:

Введите пароль: qwerty

Пароль ненадежный: отсутствуют заглавные буквы, числа и специальные символы

***Задание 6.*** Написать программу, которая определяет, является ли введенный пользователем год високосным. Год считается високосным, если он делится на 4, но не делится на 100, либо если он делится на 400.

Пример:

Введите год: 2024

2024 – високосный год

***Задание 7.*** Написать программу, которая запрашивает у пользователя три числа и выводит на экран наименьшее из них. При решении нельзя использовать встроенные функции min() и max().

Пример:

Введите три числа: 8 3 5

Наименьшее число: 3

***Задание 8.*** В магазине проводится акция. Акция работает по следующим правилам:

***Задание 9.*** Разбить предложение Съешь еще этих мягких французских булок, да выпей чаю на несколько переменных, содержащих по одному-два слова. Затем с помощью конкатенации строк необходимо собрать предложение вновь и вывести в консоль (не

забывайте про пробелы).

***Задание 10.*** Составить предложение, состоящее из строки «Нет! Да!», которая повторяется 4 раза (используйте умножение строк).

***Задание 11.*** Запросить на ввод с консоли три числа, разделенных запятой, и записать их в три отдельные переменные. Затем первое и третье число сложить и результат целочисленно разделить на второе число. Конечный результат вывести в консоль в виде

строки: Результат вычисления: <результат>.

***Задание 12.*** Запросить на ввод с консоли слово, содержащее не менее 10 символов.

С помощью срезов вывести:

1. первые 4 символа;
2. последние 2 символа;
3. символы от 4 до 8;
4. перевернутое слово.

Содержание

[1 Выполнение работы 2](#_Toc212189736)

[1.1 Задание 1 2](#_Toc212189737)

[1.2 Задание 2 3](#_Toc212189738)

[1.3 Задание 3 3](#_Toc212189739)

[1.4 Задание 4 4](#_Toc212189740)

[1.5 Задание 5 4](#_Toc212189741)

[1.6 Задание 6 5](#_Toc212189742)

[1.7 Задание 7 6](#_Toc212189743)

[1.8 Задание 8 6](#_Toc212189744)

[1.9 Задание 9 7](#_Toc212189745)

[1.10 Задание 10 8](#_Toc212189746)

[1.11 Задание 11 8](#_Toc212189747)

[1.12 Задание 12 9](#_Toc212189748)

# Выполнение работы

## Задание 1

В данном задании были созданы четыре переменные, каждая из которых относится к одному из основных типов данных: целые числа, числа с плавающей точкой, строки и логические значения. После объявления переменные были выведены на экран с помощью

функции print(). На рисунке 1 представлен код полученной программы.

|  |
| --- |
| # task1.py |
| # Объявление переменных разных типов   1. = 10 # int 2. = 3.14 # float 3. = "Привет" # str 4. = True # bool |

1

2

3

4

5

6

7

|  |
| --- |
| print(a, b, c, d) |

8

Рисунок 1 – Листинг программы для задания 1

Пояснение работы программы:

1. переменная a имеет тип int и хранит целое число;
2. переменная b имеет тип float и хранит дробное число;
3. переменная c содержит строковое значение, заключенное в кавычки;
4. переменная d является логической и может принимать значения True или False.

После выполнения программы в консоль последовательно выводятся значения всех

четырех переменных.

## Задание 2

В данном задании были созданы две переменные: одна хранит имя пользователя, другая – его возраст. После присвоения значений обе переменные выводятся в консоль

с помощью функции print(). На рисунке 2 представлен код программы.

|  |
| --- |
| # task2.py name = "Марк" age = 24 |
| print(name, age) |

1

2

3

4

Рисунок 2 – Листинг программы для задания 2

Пояснение работы программы:

1. Переменная name содержит строковое значение – имя пользователя.
2. Переменная age содержит целое значение – возраст пользователя.
3. Функция print() выводит оба значения через пробел в одной строке.

## Задание 3

В этом задании необходимо было объявить три переменные: две числовые и одну строковую, содержащую число. Затем требовалось найти сумму всех трёх чисел и вывести

результат в консоль. На рисунке 3 представлен код программы.

|  |
| --- |
| # task3.py a = 342 b = 56.2 c = '43' |
| result = a + b + int(c) |
| print(result) |

1

2

3

4

5

6

7

Рисунок 3 – Листинг программы для задания 3

Пояснение работы программы:

1. Переменная a хранит целое число 342.
2. Переменная b хранит дробное число 56.2.
3. Переменная c содержит строку «43», которая преобразуется в число

функцией int().

1. Сумма трёх чисел сохраняется в переменной result и выводится функцией print().

В результате выполнения программа выводит сумму трёх значений, приведённых

к числовому типу.

## Задание 4

В задании необходимо было вычислить значение выражения (*a* +4*b*)(*a* −3*b*)+ *a*2 для заданных переменных. На рисунке 4 представлен код решения.

|  |
| --- |
| # task4.py a = 3 b = 8 |
| result = (a + 4\*b) \* (a ‐ 3\*b) + a \*\* 2 |
| print(result) |

1

2

3

4

5

6

Рисунок 4 – Листинг программы для задания 4

Пояснение работы программы:

1. Переменным a и b присвоены значения 3 и 8.
2. Выражение вычисляется в соответствии с математическими правилами приори-

тета операций.

1. Оператор \*\* используется для возведения числа a в квадрат.
2. Результат сохраняется в переменной result и выводится в консоль.

После выполнения программы в консоли отображается вычисленное

значение выражения.

## Задание 5

Цель задания – реализовать программу для вычисления площади и периметра прямоугольника, стороны которого вводятся пользователем с консоли. На рисунке 5

представлен код программы.

|  |
| --- |
| # task5.py |
| width = float(input("Введите ширину прямоугольника: ")) |
| height = float(input("Введите высоту прямоугольника: ")) area = width \* height |
| perimeter = 2 \* (width + height) |
| print("Площадь:", area) |
| print("Периметр:", perimeter) |

1

2

3

4

5

6

7

8

9

Рисунок 5 – Листинг программы для задания 5

Пояснение работы программы:

1. С помощью функции input() запрашиваются длины сторон прямоугольника, ко-

торые преобразуются в тип float.

1. Переменная area вычисляется как произведение сторон прямоугольника.
2. Переменная perimeter вычисляется по формуле 2∗(*width*+*height*).
3. Оба результата выводятся на экран при помощи функции print().

После запуска программы пользователь вводит значения сторон и получает площадь

и периметр прямоугольника.

## Задание 6

В данном задании необходимо было вывести в консоль символическую (ASCII-art) букву w, составленную из символов \*, расположенных в три строки. Для реализации задачи использовались последовательные вызовы функции print(), каждый из которых

формирует одну строку рисунка. На рисунке 6 представлен код программы.

|  |
| --- |
| # task6.py |
| print("\* \* \*") |
| print(" \* \* \* \*") |
| print(" \* \*") |

1

2

3

4

Рисунок 6 – Листинг программы для задания 6

Пояснение работы программы:

1. Каждый вызов функции print() формирует отдельную строку символов.
2. Расположение пробелов и звёзд подобрано таким образом, чтобы визуально

получилась буква w.

1. Программа не требует ввода данных пользователем и выводит

фиксированный результат.

В результате выполнения в консоли отображается следующая фигура, показанная

на рисунке 7

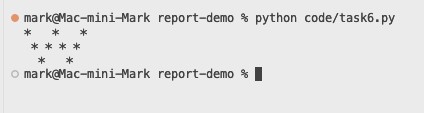


Рисунок 7 – Пример вывода программы из задания 6

## Задание 7

Задача заключалась в демонстрации работы основных арифметических и логических операторов языка Python. Для этого были созданы две переменные с произвольными числовыми значениями и проведены вычисления с использованием семи арифметических

и шести операторов сравнения. На рисунке 8 представлен код программы.

|  |
| --- |
| # task7.py x = 10 y = 3 |
| print("Сложение:", x + y) |
| print("Вычитание:", x ‐ y) |
| print("Умножение:", x \* y) |
| print("Деление:", x / y) |
| print("Целочисленное деление:", x // y) |
| print("Остаток от деления:", x % y) |
| print("Возведение в степень:", x \*\* y) |
| print("Равно:", x == y) |
| print("Не равно:", x != y) |
| print("Больше:", x > y) |
| print("Меньше:", x < y) |
| print("Больше или равно:", x >= y) |
| print("Меньше или равно:", x <= y) |

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

Рисунок 8 – Листинг программы для задания 7

Пояснение работы программы:

1. Переменным x и y присвоены значения 10 и 3.
2. В коде последовательно выполняются операции сложения, вычитания, умноже-

ния, деления, целочисленного деления, нахождения остатка и возведения в степень.

1. Затем производится сравнение двух чисел по различным условиям (равенство,

неравенство, больше, меньше и т.д.).

1. Каждая операция сопровождается пояснительным текстом в выводе.

После выполнения программы в консоли выводятся результаты всех арифметических и логических операций между переменными.

## Задание 8

В этом задании необходимо было вывести строку, содержащую имя и возраст

пользователя, с использованием f-строки. Такой способ форматирования позволяет удобно вставлять значения переменных непосредственно внутрь строки. На рисунке 9 представлен

код программы.

|  |
| --- |
| # task8.py name = "Марк" age = 24 |
| print(f"Меня зовут {name}, мне {age} лет") |

1

2

3

4

Рисунок 9 – Листинг программы для задания 8

Пояснение работы программы:

1. Созданы две переменные: name – содержит имя, и age – возраст.
2. Для объединения текста и значений переменных используется f-строка.
3. F-строка позволяет вставлять значения переменных в нужные места строки без

явной конкатенации.

После выполнения программы в консоли отображается сообщение вида: «Меня

зовут Марк, мне 24 лет».

## Задание 9

В этом задании нужно было собрать исходное предложение из нескольких переменных, каждая из которых хранит часть текста. Для объединения строк использовалась

операция конкатенации при помощи знака +. На рисунке 10 представлен код программы.

|  |
| --- |
| # task9.py part1 = "Съешь" part2 = "еще этих" part3 = "мягких французских" |
| part4 = "булок, да выпей чаю" sentence = part1 + " " + part2 + " " + part3 + " " + part4 |

1

2

3

4

5

6

7

|  |
| --- |
| print(sentence) |

8

Рисунок 10 – Листинг программы для задания 9

Пояснение работы программы:

1. Каждая из четырёх переменных (part1, part2, part3, part4) содержит

часть предложения.

1. При помощи операции + и добавления пробелов создаётся итоговая строка.
2. Полученная строка сохраняется в переменной sentence и выводится на экран.

Результатом выполнения программы является восстановленное предложение: «Съешь ещё этих мягких французских булок, да выпей чаю».

## Задание 10

В данном задании необходимо было создать строку «Нет! Да!», повторяющуюся четыре раза подряд. Для решения использовалась операция умножения строк, позволяющая повторить заданную последовательность символов нужное количество раз. На рисунке 11

представлен код программы.

|  |
| --- |
| # task10.py phrase = "Нет! Да!" result = phrase \* 4 |
| print(result) |

1

2

3

4

Рисунок 11 – Листинг программы для задания 10

Пояснение работы программы:

1. В переменной phrase хранится исходная строка «Нет! Да!».
2. С помощью операции умножения *phrase* ∗ 4 строка повторяется четыре раза

подряд.

1. Результат сохраняется в переменной result и выводится в консоль.

После выполнения программы в консоли отображается результат: «Нет! Да!Нет!

Да!Нет! Да!Нет! Да!».

## Задание 11

В этом задании требовалось запросить у пользователя три числа, разделённые запятыми, а затем вычислить выражение (a + c) // b с использованием целочисленного деления. Результат необходимо вывести в консоль в виде поясняющего сообщения. На

рисунке 12 представлен код программы.

|  |
| --- |
| # task11.py |
| numbers = input("Введите три числа через запятую: ") |
| a, b, c = map(float, numbers.split(",")) |
| result = int((a + c) // b) |
| print(f"Результат вычисления: {result}") |

1

2

3

4

5

6

Рисунок 12 – Листинг программы для задания 11

Пояснение работы программы:

1. Пользователь вводит три числа, разделённые запятыми, которые считываются

функцией input().

1. С помощью split( ”) введённая строка разбивается на три элемента.
2. Функция map(float, ...) преобразует каждое значение в тип с плавающей точкой.
3. После этого вычисляется результат целочисленного деления суммы первого и

третьего чисел на второе.

1. Итоговый результат выводится на экран с помощью форматированной строки.

После выполнения программа запрашивает ввод данных и выводит строку вида:

«Результат вычисления: 15».

## Задание 12

В последнем задании лабораторной работы необходимо было запросить у пользователя слово, содержащее не менее десяти символов, и при помощи срезов вывести отдельные части строки, а также её обратный вариант. На рисунке 13 представлен код

программы.

|  |
| --- |
| # task12.py |
| word = input("Введите слово не( менее 10 символов): ") |
| print("Первые 4 символа:", word[:4]) |
| print("Последние 2 символа:", word[‐2:]) |
| print("Символы с 4 по 8:", word[3:8]) |
| print("Перевернутое слово:", word[::‐1]) |

1

2

3

4

5

6

7

Рисунок 13 – Листинг программы для задания 12

Пояснение работы программы:

1. Программа запрашивает у пользователя ввод слова длиной не менее

десяти символов.

1. Срез word[:4] возвращает первые четыре символа строки.
2. Срез word[-2:] возвращает последние два символа.
3. Срез word[3:8] выводит символы с четвёртого по восьмой.
4. Срез word[::-1] переворачивает строку, выводя её в обратном порядке.

После выполнения программа последовательно отображает результаты всех операций со срезами для введённого слова.