

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

Л.С.Лавриненко

Ассистент
преподавателя

М. В. Водяницкий

Владивосток 2025

Задание

Выполнить задание 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 вводится пользователем с консоли.

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.

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.

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.

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.

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()`.

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

Сумма покупки	Скидка
до 1000	0%
1000–5000	5%
5000–10000	10%
более 10000	15%

Напишите программу, которая запрашивает сумму покупки и выводит размер скидки и итоговую сумму к оплате.

Задание 9. Написать программу, которая определяет время суток по введенному часу (целое число от 0 до 23).

Время	Период
0–5	Ночь
6–11	Утро
12–17	День
18–23	Вечер

Задание 10. Написать программу, которая определяет, является ли введенное число простым. Число называется простым, если оно больше 1 и делится только на 1 и само себя. Программа должна корректно обрабатывать некорректный ввод и выводить соответствующие сообщения об ошибках.

Содержание

1 Выполнение работы	3
1.1 Задание 1	3
1.2 Задание 2	3
1.3 Задание 3	4
1.4 Задание 4	5
1.5 Задание 5	6
1.6 Задание 6	7
1.7 Задание 7	8
1.8 Задание 8	9
1.9 Задание 9	10
1.10 Задние 10	11

1 Выполнение работы

1.1 Задание 1

В данном задании необходимо было написать программу, которая определяла бы как будет вести себя кондиционер. Если температура, которую ввел пользователь 20 градусов и выше, то выводится надпись, что кондиционер выключен, если меньше - включен. На рисунке 1 представлен код программы.

```
def task1():
    try:
        temperature = float(input("Введите температуру: "))
        if temperature >= 20:
            print("Кондиционер выключен")
        else:
            print("Кондиционер включен")
    except ValueError:
        print("Ошибка")
```

Рисунок 1 – Листинг программы для задания 1

Пояснение работы программы:

- 1) Пользователь вводит число, которое считывается функцией `float(input())`
- 2) Далее с помощью условного оператора проверяется условие: если температура больше или равна 20 градусам.
- 3) При выполнении условия выводится сообщение "Кондиционер выключен".
- 4) Если условие не выполняется (температура ниже 20 градусов), выводится сообщение "Кондиционер включен".

1.2 Задание 2

В данном задании необходимо было написать программу, которая запрашивала бы у пользователя номер месяца и выводила к какому сезону этот месяц относится. На рисунке 2 представлен код программы.

```
# задание 2
def task2():
    try:
        month = int(input("Введите номер месяца: "))
        if month in [12, 1, 2]:
            print("Это зима")
        elif month in [3, 4, 5]:
            print("Это весна")
        elif month in [6, 7, 8]:
            print("Это лето")
        elif month in [9, 10, 11]:
            print("Это осень")
        else:
            print("Ошибка")
    except ValueError:
        print("Ошибка")
```

Рисунок 2 – Листинг программы для задания 2

Пояснение работы программы:

1) Пользователь вводит число, которое считывается функцией `input()` и переводится из текста в цифровое значение с помощью функции `int()`.

2) Программа проверяет четыре условия для определения времени года:

- Зима: декабрь (12), январь (1), февраль (2)
- Весна: март (3), апрель (4), май (5)
- Лето: июнь (6), июль (7), август (8)
- Осень: сентябрь (9), октябрь (10), ноябрь (11)

3) В зависимости от введенного номера месяца выводится соответствующее время года.

1.3 Задание 3

В этом задании необходимо было запросить у пользователя возраст собаки (в годах), который бы программа переводила в человеческие годы. Также программа должна была корректно обрабатывать входные данные и выводить соответствующие сообщения об ошибках: если вводится не число, если вводится число меньше, если вводится число большее 22. На рисунке 3 представлен код программы.

```
# задание 3
def task3():
    try:
        dog_age = float(input("Введите возраст собаки (в годах): "))

        if dog_age < 1:
            print("Ошибка")
        elif dog_age > 22:
            print("Ошибка")
        else:
            if dog_age <= 2:
                human_age = dog_age * 10.5
            else:
                human_age = 21 + (dog_age - 2) * 4

            print(f"Возраст собаки в человеческих годах: {human_age}")
    except ValueError:
        print("Ошибка")
```

Рисунок 3 – Листинг программы для задания 3

Пояснение работы программы:

1) Пользователь вводит число, которое считывается функцией `float(input())` и переводится из текста в цифровое значение

- 2) Используется блок try-except для обработки ошибок ввода
- 3) Проверяются границы допустимого возраста (от 1 до 22 лет)
- 4) Для собак младше 3 лет применяется коэффициент 10.5 лет за каждый год и результат сохраняется в переменную `human_age`, которая выводится с помощью `print`
- 5) Для собак старше 2 лет используется формула: $21 + 4 \cdot (\text{возраст, который ввел пользователь} - 2)$. Полученный возраст сохраняется в переменную `human_age` и выводится с помощью `print`
- 6) При вводе нечислового значения выводится соответствующее сообщение об ошибке

1.4 Задание 4

В данном задании необходимо было запросить у пользователя число и с помощью свойств деления числа на цело на 6 (последняя цифра четная, сумма всех цифр делиться на 3) понять делится ли введенное число на 6. На рисунке 4 представлен код программы.

```
def task4():
    try:
        number = input("Введите число: ")

        last_digit = int(number[-1])
        even_last = last_digit % 2 == 0

        digit_sum = sum(int(digit) for digit in number if digit.isdigit())
        divisible_by_3 = digit_sum % 3 == 0

        if even_last and divisible_by_3:
            print(f"Число {number} делится на 6")
        else:
            print(f"Число {number} не делится на 6")

    except ValueError:
        print("Ошибка")
```

Рисунок 4 – Листинг программы для задания 4

Пояснение работы программы:

- 1) Пользователь вводит число, которое считывается как строка с помощью функции `input()`
- 2) Используется блок try-except для обработки ошибок ввода
- 3) Берется последняя цифра числа и проверяется на четность с помощью операции остатка от деления на 2

- 4) Вычисляется сумма всех цифр числа и проверяется делимость этой суммы на 3
- 5) Если число четное (последняя цифра четная) и сумма цифр делится на 3, то выводится сообщение что число делится на 6
- 6) Если хотя бы одно условие не выполняется, выводится сообщение что число не делится на 6
- 7) При вводе некорректных данных (например, строки без цифр) выводится сообщение об ошибке

1.5 Задание 5

В следующем задании необходимо было запросить у пользователя пароль, который бы программа проверяла на надежность по следующим критериям: есть заглавные буквы латиницы, строчные буквы латиницы, числа, специальные знаки и пароль содержит больше 8 символов. Если пароль не содержит всех условий, то выводится, что он ненадежный, если содержит, то выводится, что он надежный. На рисунке 5 представлен код программы.

```
def task5():
    password = input("Введите пароль: ")

    errors = []

    if len(password) < 8:
        errors.append("длина менее 8 символов")

    has_upper = any(char.isupper() for char in password)
    if not has_upper:
        errors.append("отсутствуют заглавные буквы")

    has_lower = any(char.islower() for char in password)
    if not has_lower:
        errors.append("отсутствуют строчные буквы")

    has_digit = any(char.isdigit() for char in password)
    if not has_digit:
        errors.append("отсутствуют цифры")

    has_special = any(not char.isalnum() for char in password)
    if not has_special:
        errors.append("отсутствуют специальные символы")

    if errors:
        print(f"Пароль ненадежный: {', '.join(errors)}")
    else:
        print("Пароль надежный")
```

Рисунок 5 – Листинг программы для задания 5

Пояснение работы программы:

- 1) Пользователь вводит пароль, который считывается как строка с помощью функции `input()`
- 2) Создается пустой список `errors` для хранения ошибок пароля
- 3) Проверяется длина пароля - если меньше 8 символов, добавляется соответствующая ошибка в список
- 4) Проверяется наличие заглавных букв в пароле - если нет ни одной, добавляется ошибка в список
- 5) Проверяется наличие строчных букв в пароле - если нет ни одной, добавляется ошибка в список
- 6) Проверяется наличие цифр в пароле - если нет ни одной, добавляется ошибка в список
- 7) Проверяется наличие специальных символов в пароле - если нет ни одного, добавляется ошибка в список
- 8) Если список `errors` не пустой, выводится сообщение о ненадежном пароле с перечислением всех ошибок
- 9) Если список `errors` пустой, выводится сообщение о надежном пароле

1.6 Задание 6

В следующем задании необходимо написать программу, которая определяет, является ли введенный пользователем год високосным. На рисунке 6 представлен код программы.

```
def task6():
    try:
        year = int(input("Введите год: "))

        if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
            print(f"{year} - високосный год")
        else:
            print(f"{year} - не високосный год")

    except ValueError:
        print("Ошибка")
```

Рисунок 6 – Листинг программы для задания 6

Пояснение работы программы:

- 1) Пользователь вводит год, который считывается функцией `int(input())` и переводится из текста в целое число
- 2) Используется блок `try-except` для обработки ошибок ввода

3) Проверяется условие високосного года: год должен делиться на 4, но не делиться на 100, либо делиться на 400

4) Если условие выполняется, выводится сообщение что год високосный

5) Если условие не выполняется, выводится сообщение что год не високосный

6) При вводе нечислового значения выводится соответствующее сообщение об ошибке

1.7 Задание 7

В данном задании необходимо было без специальных функций из введенных трех чисел вывести наименьшее. На рисунке 7 представлен код программы.

```
def task7():
    try:
        numbers = input("Введите три числа: ").split()
        if len(numbers) != 3:
            print("Ошибка")
            return

        a, b, c = map(float, numbers)

        min_number = a
        if b < min_number:
            min_number = b
        if c < min_number:
            min_number = c

        print(f"Наименьшее число: {min_number}")

    except ValueError:
        print("Ошибка")
```

Рисунок 7 – Листинг программы для задания 7

Пояснение работы программы:

1) Пользователь вводит три числа в одной строке, которые считаются как строка и разбиваются на список с помощью метода `split()`

2) Проверяется что в списке ровно три элемента, если нет - выводится ошибка и программа завершается

3) Три элемента списка преобразуются в числа с плавающей точкой с помощью функции `map(float, numbers)`

4) Переменная `min_number` инициализируется первым числом а

5) Сравнивается второе число `b` с текущим минимальным числом, и если `b` меньше, то `min_number` обновляется

6) Сравнивается третье число с текущим минимальным числом, и если с меньше, то `min_number` обновляется

7) Выводится найденное наименьшее число

8) При вводе некорректных данных (не числа или неправильное количество чисел) выводится сообщение об ошибке

1.8 Задание 8

В следующем задании необходимо было написать программу, которая запрашивала бы у пользователя сумму покупки, а позже выводила скидку, основанную на его сумме, и итоговую сумму с учетом скидки. На рисунке 8 представлен код программы.

```
def task8():
    try:
        purchase_amount = float(input("Введите сумму покупки: "))

        if purchase_amount < 1000:
            discount = 0
        elif purchase_amount <= 5000:
            discount = 5
        elif purchase_amount <= 10000:
            discount = 10
        else:
            discount = 15

        final_amount = purchase_amount * (1 - discount / 100)

        print(f"Ваша скидка: {discount}%")
        print(f"К оплате: {final_amount}")

    except ValueError:
        print("Ошибка")
```

Рисунок 8 – Листинг программы для задания 8

Пояснение работы программы:

1) Пользователь вводит сумму покупки, которая считывается функцией `float(input())` и переводится из текста в число с плавающей точкой

2) Используется блок `try-except` для обработки ошибок ввода

3) Проверяется диапазон суммы покупки для определения размера скидки по правилам:

- Меньше 1000: скидка 0%
- От 1000 до 5000: скидка 5%
- От 5001 до 10000: скидка 10%
- Больше 10000: скидка 15%

4) Вычисляется итоговая сумма к оплате с учетом скидки

- 5) Выводится информация о размере полученной скидки в процентах
- 6) Выводится итоговая сумма к оплате после применения скидки
- 7) При вводе нечислового значения выводится соответствующее сообщение об ошибке

1.9 Задание 9

В данном задании необходимо было написать программу, которая запрашивала бы у пользователя целое число от 0 до 23 и выводила к какой части дня это время относится. На рисунке 9 представлен код программы.

```
def task9():
    try:
        hour = int(input("Введите час (0-23):"))

        if 0 <= hour <= 5:
            print("Сейчас ночь")
        elif 6 <= hour <= 11:
            print("Сейчас утро")
        elif 12 <= hour <= 17:
            print("Сейчас день")
        elif 18 <= hour <= 23:
            print("Сейчас вечер")
        else:
            print("Ошибка")

    except ValueError:
        print("Ошибка")
```

Рисунок 9 – Листинг программы для задания 9

Пояснение работы программы:

- 1) Пользователь вводит час, который считывается функцией `int(input())` и переводится из текста в целое число
- 2) Используется блок `try-except` для обработки ошибок ввода
- 3) Проверяется значение часа и определяется время суток по правилам:
 - От 0 до 5 часов: ночь
 - От 6 до 11 часов: утро
 - От 12 до 17 часов: день
 - От 18 до 23 часов: вечер
- 4) Выводится соответствующее сообщение о текущем времени суток

5) Если введенное число не попадает в диапазон от 0 до 23, выводится сообщение об ошибке

6) При вводе нечислового значения выводится соответствующее сообщение об ошибке

1.10 Задние 10

В последнем задании необходимо было создать программу, которая определяла является ли введенное пользователем число простым или составным. На рисунке 10 представлен код программы.

```
def task10():
    try:
        number = int(input("Введите число: "))

        if number <= 1:
            print(f"{number} - не простое число")
            return

        is_prime = True
        for i in range(2, int(number**0.5) + 1):
            if number % i == 0:
                is_prime = False
                break

        if is_prime:
            print(f"{number} - простое число")
        else:
            print(f"{number} - составное число")

    except ValueError:
        print("Ошибка")
```

Рисунок 10 – Листинг программы для задания 10

Пояснение работы программы:

- 1) Пользователь вводит число, которое считывается функцией `int(input())` и переводится из текста в целое число
- 2) Используется блок `try-except` для обработки ошибок ввода
- 3) Проверяется базовое условие: если число меньше или равно 1, оно не является простым, выводится сообщение и программа завершается
- 4) Устанавливается флаг `is_prime` в значение `True`, предполагая что число простое
- 5) Проверяются все возможные делители числа от 2 до квадратного корня из числа

- 6) Если находится хотя бы один делитель, флаг `is_prime` устанавливается в `False` и цикл прерывается
- 7) После проверки всех возможных делителей выводится сообщение о том, является ли число простым или составным
- 8) При вводе нечислового значения выводится соответствующее сообщение об ошибке