

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

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6
по дисциплине «Информатика и
программирование»

Студент _____ Сакс Р.А.
Преподаватель _____ Водяницкий М.В.

Задание

Задание 1 Написать функцию, которая конвертирует время из одной величины в другую.

На вход подается:

число (величина времени) исходная единица измерения единица измерения, в которую нужно перевести Функция должна вернуть конвертированное значение

Примеры (формат ввода/вывода можно выбрать свой, если нет строгих требований):

Вход Выход 4h m 240m 30m h 0.5h 12s h 0.03h Задание 2 Пользователь делает вклад в банке в размере a рублей сроком на n лет

Процент по вкладу зависит от суммы и срока

Зависимость от суммы:

каждые 10 000 рублей увеличивают ставку на 0.3но суммарное увеличение не может превышать 5минимальный вклад - 30 000 рублей Зависимость от срока:

первые 3 года - 3от 4 до 6 лет - 5более 6 лет - 2Необходимо написать функцию, которая рассчитывает прибыль пользователя без учета первоначально вложенной суммы

Используется сложный процент: каждый год процент начисляется на текущую сумму вклада

На вход подаются: сумма вклада и количество лет. Результат: сумма прибыли (не весь вклад, а только заработанные проценты)

Примеры:

Вход Выход 30000 3 3648.67 100000 5 38920.10 200000 8 183925.42 Задание 3 Написать функцию для вывода всех простых чисел в заданном диапазоне. Нужно учитывать некорректные данные (например, начало больше конца или диапазон без простых чисел)

На вход подаются два числа: начало и конец диапазона (включительно). На выходе - список всех простых чисел или сообщение об ошибке

Примеры:

Вход Выход 1 10 2 3 5 7 15 120 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103 107 109 113 0 1 Error! (Формат вывода списка простых чисел может быть любым удобным: в строку через пробел, в несколько строк и т.п.)

Задание 4 Реализовать функцию сложения двух матриц

При сложении двух матриц получается новая матрица того же размера, где каждый элемент - это сумма элементов с тем же индексом из двух исходных матриц

Ограничения:

складывать можно только матрицы одинакового размера размер матрицы должен быть строго больше 2 (например, 3×3 , 4×4 и т.д.) при нарушении условий нужно вывести сообщение об ошибке На вход подаются:

размер матрицы n (для квадратной матрицы $n \times n$) элементы первой матрицы (по строкам, через пробел) элементы второй матрицы в таком же формате Результат - новая матрица (в том же формате), либо сообщение об ошибке

Пример (один из возможных вариантов формата):

Вход:

2 2 5 5 3 5 2 4 1 Выход:

7 7 9 4 Пример с ошибкой (слишком маленький размер, неправильный ввод и т.п.):

1 4 5 Выход:

Error! Задание 5 Написать функцию, которая определяет, является ли строка палиндромом

Палиндром - это строка, которая читается одинаково слева направо и справа налево (обычно без учета пробелов, регистра и знаков препинания - эти правила нужно явно задать в своей реализации)

На вход подается строка. На выходе:

Да, если это палиндром Нет, если это не палиндром Примеры:

Вход Выход А роза упала на лапу Азора Да Borrow or rob Да Алфавитный порядок
Нет

Содержание

1 Выполнение работы	3
1.1 Задание 1	3
1.2 Задание 2	4
1.3 Задание 3	5
1.4 Задание 4	6
1.5 Задание 5	7

1 Выполнение работы

1.1 Задание 1

Функция Lab6Zad1 реализует конвертацию времени между часами и минутами. Программа принимает на вход три параметра: числовое значение времени, исходную единицу измерения и целевую единицу измерения.

На рисунке 1 предоставлен код данной программы.

```
def Lab6Zad1():
    def Zad(time0,ed0,ed2):
        time0 = int(time0)
        ed0,ed2 = str(ed0).lower(),str(ed2).lower()
        if ed0 == "h" and ed2 == "m":
            return F"{time0}H == {time0*60}M"
        elif ed2 == "h" and ed0 == "m":
            return F"{time0}M == {time0/60}H"
        else:
            return time0,ed0
    print(Zad( time0: 12, ed0: "H", ed2: "m"))

# Lab6Zad1()
```

Рисунок 1 – Листинг программы для задания 1

Пояснение работы программы:

- 1) Функция Zad принимает три параметра: время и две единицы измерения.
- 2) Преобразует единицы измерения к нижнему регистру для унификации.
- 3) Если исходная единица - часы (h), а целевая - минуты (m), то выполняется умножение на 60.
- 4) Если исходная единица - минуты (m), а целевая - часы (h), то выполняется деление на 60.
- 5) В случае других комбинаций возвращается исходное значение и единица измерения.
- 6) Функция возвращает строку с результатом конвертации.

1.2 Задание 2

Функция Lab6Zad2 рассчитывает прибыль по банковскому вкладу с учётом сложного процента. Процентная ставка зависит от срока вклада и суммы, с дополнительными условиями по минимальному вкладу и максимальному увеличению ставки.

На рисунке 2 предоставлен код данной программы.

```

def Lab6Zad2():
    def Zad(SumNak, years):
        # Проверка минимальной суммы
        if SumNak < 30000:
            print("Ошибка: минимальный вклад 30 000 рублей")
            return

        # 1. Процент за срок
        if years <= 3:
            procient = 3
        elif years >= 4 and years <= 6:
            procient = 5
        else: # больше 6 лет
            procient = 2

        # 2. Процент за сумму
        # Сколько полных 10 тысяч в сумме
        kol_10k = SumNak // 10000
        dobavka = kol_10k * 0.3

        # Проверяем максимум 5%
        if dobavka > 5:
            dobavka = 5

        # Итоговая ставка
        procient += dobavka

        # 3. Считаем прибыль со сложным процентом
        # Формула: сумма * (1 + процент/100)^годы
        itog_summa = SumNak

        # Начисляем проценты каждый год
        for god in range(years):
            itog_summa = itog_summa * (1 + procient / 100)

        # Прибыль = итоговая сумма - начальная сумма
        pribyl = itog_summa - SumNak

        # Округляем до 2 знаков (копеек)
        pribyl = round(pribyl, 2)

    print(f"Вклад: {SumNak} руб, срок: {years} лет")
    print(f"Процентная ставка: {procient}% годовых")

```

Рисунок 2 – Листинг программы для задания 2

Пояснение работы программы:

- 1) Проверяется минимальная сумма вклада (30 000 рублей).
- 2) Определяется базовая процентная ставка в зависимости от срока вклада.
- 3) Рассчитывается дополнительная ставка в зависимости от суммы (0.3% за каждые 10 000 рублей, но не более 5%).
- 4) Вычисляется итоговая процентная ставка.
- 5) Рассчитывается конечная сумма вклада с учётом сложного процента.
- 6) Вычисляется прибыль как разница между конечной и начальной суммой.
- 7) Результат округляется до 2 знаков после запятой (копеек).

1.3 Задание 3

Функция Lab6Zad3 находит все простые числа в заданном диапазоне. Программа учитывает некорректные данные (например, когда начало диапазона больше конца) и сообщает об ошибке при отсутствии простых чисел.

На рисунке 3 предоставлен код данной программы.

```

def Lab6Zad3():
    def SimpleNumber(A,B):
        Kop = []
        if A<B:
            for i in range(A,B+1):
                if i>1:
                    flagSimple = True
                    for j in range(2,B):
                        if i%j==0 and j!=i:
                            flagSimple = False
                            break
                    if flagSimple:
                        Kop.append(i)
        if len(Kop)!=0:
            print(*Kop)
        else:
            print("Error")

    SimpleNumber( A: 2, B: 13)
    SimpleNumber( A: 1, B: 10)
    SimpleNumber( A: 15, B: 120)
    SimpleNumber( A: 0, B: 1)

# Lab6Zad3()

```

Рисунок 3 – Листинг программы для задания 3

Пояснение работы программы:

- 1) Проверяется, что начало диапазона меньше или равно концу.
- 2) Для каждого числа в диапазоне проверяется, является ли оно простым.
- 3) Проверка простоты осуществляется путём деления числа на все числа от 2 до \sqrt{n} .
- 4) Найденные простые числа добавляются в список.
- 5) Если список простых чисел пуст, выводится сообщение об ошибке.
- 6) В противном случае выводятся все найденные простые числа.

1.4 Задание 4

Функция Lab6Zad4 реализует сложение двух квадратных матриц одинакового размера. Программа проверяет корректность ввода данных и размеров матриц, выводя сообщение об ошибке при нарушении условий.

На рисунке 4 предоставлен код данной программы.

```
def Lab6Zad4():
    def SumMatrix():
        # Читаем размер матрицы
        n_str = input()

        # Попробуем преобразовать в число
        try:
            n = int(n_str)
        except:
            print("Error!")
            return

        # Проверяем размер
        if n <= 2:
            print("Error!")
            return

        # Читаем первую матрицу
        matrix1 = []
        i = 0
        while i < n:
            try:
                row_str = input()
            except:
                print("Error!")
                return

            parts = row_str.split()

            # Проверяем количество чисел
            if len(parts) != n:
                print("Error!")
                return

            # Попробуем сделать числа из строк
            ch2 = []
            j = 0
            while j < n:
                try:
```

Пояснение работы программы:

- 1) Считывается размер матрицы n (должен быть больше 2).
- 2) Проверяется корректность ввода (целое число).
- 3) Последовательночитываются две матрицы размером $n \times n$.
- 4) Проверяется, что каждая строка содержит ровно n чисел.
- 5) Проверяется, что все введённые значения являются целыми числами.
- 6) При обнаружении ошибки выводится сообщение `Error!`.
- 7) Матрицы поэлементно складываются.
- 8) Результат выводится в том же формате, что и ввод.

1.5 Задание 5

Функция `Lab6Zad5` определяет, является ли введённая строка палиндромом. Программа игнорирует регистр букв, пробелы и знаки препинания при проверке.

На рисунке 5 предоставлен код данной программы.

```
def Lab6Zad5():
    def palindrome():
        # Получаем строки
        text = input()

        # Приводим к нижнему регистру
        text = text.lower()

        # Удаляем все не-буквы (пробелы, знаки препинания)
        letters = "абвгдеёжзийклмнопрстуфцчшъэюяаbcdeffghijklmnoprstuvwxyz0123456789"
        clean_text = ""

        for char in text:
            if char in letters:
                clean_text += char

        # Проверяем палиндром
        n = len(clean_text)
        is_pal = True

        for i in range(n // 2):
            if clean_text[i] != clean_text[n - i - 1]:
                is_pal = False
                break

        if is_pal:
            print("Да")
        else:
            print("Нет")

    palindrome()

# Borrow or rob Да
# Алфавитный порядок     Нет
# Lab6Zad5()
```

Рисунок 5 – Листинг программы для задания 5

Пояснение работы программы:

- 1) Считывается строка от пользователя.
 - 2) Стока приводится к нижнему регистру.
 - 3) Удаляются все символы, не являющиеся буквами или цифрами (пробелы, знаки препинания).
 - 4) Проверяется, является ли очищенная строка палиндромом (читается одинаково слева направо и справа налево).
 - 5) Сравнение выполняется путём попарного сравнения символов с начала и конца строки.
 - 6) Выводится , если строка палиндром, или в противном случае.

Список использованных источников