

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 №6
по дисциплине
«Информатика и программирование»

Студент
гр. БИН-25-3 _____ Герцов Д.Е.
Ассистент
преподавателя _____ М.В. Водяницкий

Задание

Выполнить задания на Python и оформить отчет по стандартам ВВГУ.

Задание 1. Написать функцию, которая конвертирует время из одной величины в другую. На вход подаются: число (величина времени), исходная единица измерения, единица измерения, в которую нужно перевести. Функция должна вернуть конвертированное значение.

Задание 2. Пользователь делает вклад в банке в размере a рублей сроком на n лет. Процент по вкладу зависит от суммы и срока. Необходимо рассчитать прибыль по сложным процентам (без учёта первоначальной суммы).

Задание 3. Написать функцию для вывода всех простых чисел в заданном диапазоне. Учитывать некорректные данные и отсутствие простых чисел.

Задание 4. Реализовать функцию сложения двух квадратных матриц. Складывать можно только матрицы размера строго больше 2. При нарушении условий — сообщение об ошибке.

Задание 5. Написать функцию, определяющую, является ли строка палиндромом (игнорируя регистр, пробелы и знаки препинания).

Содержание

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

1 Выполнение работы

1.1 Задание 1

Реализована функция перевода времени между единицами измерения: часы ('h') и минуты ('m'). Ввод осуществляется в формате «значение_единица целевая_единица». Поддерживаются преобразования: часы → минуты и минуты → часы. На рисунке 1 представлен код программы.

```

1 def bank_sistem():
2     god_stavka = 0
3     bonus_stavki = 0
4     print("минимальный вклад 30000 рублей.")
5     try:
6         vklad = input("Сделайте вклад указав срок в годах, на
который собираетесь его класть: ").split(' ')
7         vkladint0 = int(vklad[0])
8         vkladint1 = int(vklad[1])
9     except:
10        print("Введите значения через пробел")
11
12    if vkladint0 < 30000:
13        print("Вклад не может быть меньше 30000 рублей")
14    else:
15
16        c = vkladint0 - 30000
17        god_stavka = 0
18        while (c >= 10000) and (god_stavka < 0.05):
19            c -= 10000
20            god_stavka += 0.003
21
22        if vkladint1 <= 3:
23            bonus_stavki = 0.03
24        elif 4 <= vkladint1 <= 6:
25            bonus_stavki = 0.05
26        elif vkladint1 >= 6:
27            bonus_stavki = 0.02
28
29        summa_stavok = god_stavka + bonus_stavki
30        vkladint2 = vkladint0
31
32        for god in range(vkladint1):
33            vkladint2 = vkladint2 * (1+summa_stavok)
34
35        vkladint3 = vkladint2 - vkladint0
36
37        print(vkladint3)
38
39 bank_sistem()

```

Рисунок 1 – Листинг программы для задания 1

- 1) Ввод строки с параметрами
- 2) Разделение на исходное значение и целевую единицу
- 3) Проверка типа исходной единицы («h» или «m»)
- 4) Выполнение перевода с помощью арифметических операций
- 5) Вывод результата

После выполнение программы в зависимости от введеного числа, будет получен нужный результат

1.2 Задание 2

Реализована система расчёта банковского вклада с учётом базовой ставки (зависит от срока) и бонусной ставки (зависит от суммы). Используется сложный процент: каждый год процент начисляется на текущую сумму. На рисунке 2 представлен код программы.

```

1 def bank_sistem():
2     god_stavka = 0
3     bonus_stavki = 0
4     print("минимальный вклад 30000 рублей.")
5     try:
6         vklad = input("Сделайте вклад указав срок в годах, на
который собираетесь его класть: ").split(' ')
7         vkladint0 = int(vklad[0])
8         vkladint1 = int(vklad[1])
9     except:
10        print("Ведите значения через пробел")
11
12    if vkladint0 < 30000:
13        print("Вклад не может быть меньше 30000 рублей")
14    else:
15
16        c = vkladint0 - 30000
17        god_stavka = 0
18        while (c >= 10000) and (god_stavka < 0.05):
19            c -= 10000
20            god_stavka += 0.003
21
22        if vkladint1 <= 3:
23            bonus_stavki = 0.03
24        elif 4 <= vkladint1 <= 6:
25            bonus_stavki = 0.05
26        elif vkladint1 >= 6:
27            bonus_stavki = 0.02
28
29        summa_stavok = god_stavka + bonus_stavki
30        vkladint2 = vkladint0
31
32        for god in range(vkladint1):
33            vkladint2 = vkladint2 * (1+summa_stavok)
34
35        vkladint3 = vkladint2 - vkladint0
36
37        print(vkladint3)
38
39 bank_sistem()

```

Рисунок 2 – Листинг программы для задания 2

- 1) Ввод суммы вклада и срока
- 2) Проверка минимального вклада (30 000 руб.)
- 3) Расчёт бонусной ставки (по 0.3% за каждые 10 000 руб., максимум 5%)
- 4) Определение базовой ставки по сроку (3%, 5% или 2%)
- 5) Циклический расчёт итоговой суммы по сложному проценту
- 6) Вывод прибыли (итоговая сумма минус первоначальный вклад)

После выполнение программы в зависимости от введеного числа, будет получен нужный результат

1.3 Задание 3

Реализована функция поиска всех простых чисел в заданном диапазоне. Простое число — натуральное число больше 1, имеющее ровно два делителя. Диапазон вводится как два целых числа. На рисунке 3 представлен код программы.

```

1 def simple_number(start, end):
2     simple_numbers = []
3     errors= []
4     def is_simple(num):
5         if num <2:
6             return errors.append(num)
7             for i in range(2, int(num**0.5)+1):
8                 if num%i == 0:
9                     return False
10                return True
11
12    for n in range(start, end + 1):
13        if is_simple(n):
14            simple_numbers.append(n)
15    return simple_numbers
16
17 print(simple_number(1,100))

```

Рисунок 3 – Листинг программы для задания 3

- 1) Перебор всех чисел в диапазоне [start, end].
- 2) Сбор простых чисел в список.
- 3) Возврат результата (список или пустой список при отсутствии простых).
- 4) Вывод результата.

После выполнение программы в зависимости от введеного числа, будет получен нужный результат

1.4 Задание 4

Реализована функция сложения двух квадратных матриц. Программа проверяет, что размер матрицы больше 2, и что обе матрицы имеют корректный ввод. На рисунке 4 представлен код программы.

```

1 def add_matrices(m1, m2, n):
2     result = []
3     for i in range(n):
4         row = []
5         for j in range(n):
6             row.append(m1[i][j] + m2[i][j])
7         result.append(row)
8     return result
9
10 def read_matrix(n):
11     matrix = []
12     for _ in range(n):
13         row = list(map(int, input().split()))
14         if len(row) != n:
15             raise ValueError("Неверное количество элементов в
 строке")
16         matrix.append(row)
17     return matrix
18
19 try:
20     n = int(input().strip())
21
22     if n <= 2:
23         print("Ошибка: размер матрицы должен быть больше 2")
24         exit()
25
26     a = read_matrix(n)
27     b = read_matrix(n)
28
29     result = add_matrices(a, b, n)
30
31     for row in result:
32         print(' '.join(map(str, row)))
33
34 except Exception as e:
35     print("Ошибка: некорректный ввод")

```

Рисунок 4 – Листинг программы для задания 4

- 1) Ввод размера матрицы n
- 2) Проверка условия $n > 2$
- 3) Чтение элементов первой и второй матриц построчно
- 4) Проверка корректности количества элементов в строках
- 5) Поэлементное сложение матриц
- 6) Вывод результирующей матрицы или сообщения об ошибке

После выполнение программы в зависимости от введеного числа, будет получен нужный результат

1.5 Задание 5

Реализована функция проверки строки на палиндром. При проверке игнорируются регистр, пробелы, знаки препинания и любые не-буквенно-цифровые символы. На рисунке 5 представлен код программы.

```
1 def is_poly():
2     s = input("Введите строку: ")
3     clean = ''.join(ch for ch in s if ch.isalnum()).upper()
4     if clean == clean[::-1]:
5         print("Все верно!")
6     else:
7         print("Все плохо!")
8
9 is_poly()
```

Рисунок 5 – Листинг программы для задания 5

- 1) Ввод строки от пользователя
- 2) Фильтрация строки: остаются только буквы и цифры
- 3) Приведение к верхнему регистру
- 4) Сравнение чистой строки с её реверсом
- 5) Вывод результата: «Все верно!» или «Все плохо!»

После выполнение программы в зависимости от введеного числа, будет получен нужный результат