ФГБОУ ВО «Сочинский государственный университет»

Факультет Инновационных, Инженерных и Цифровых Технологий

Кафедра информационных технологий

# **ОТЧЕТ**

о выполнении лабораторной работы №3

*«РЕАЛИЗАЦИЯ АЛГОРИТМОВ*

*С ЦИКЛИЧЕСКОЙ СТРУКТУРОЙ. ЦИКЛЫ WHILE И FOR»*

по дисциплине **«Алгоритмизация и программирование»**

***Выполнил:***

студент 1 курса гр.24-ПИЦ

Нерсесян С. В.

«\_*20*\_»\_\_\_\_*Октября*\_\_\_\_2024г

***Проверил:***

проф. каф. инф. тех., д.т.н.

Попов Дмитрий Иванович

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2024г

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

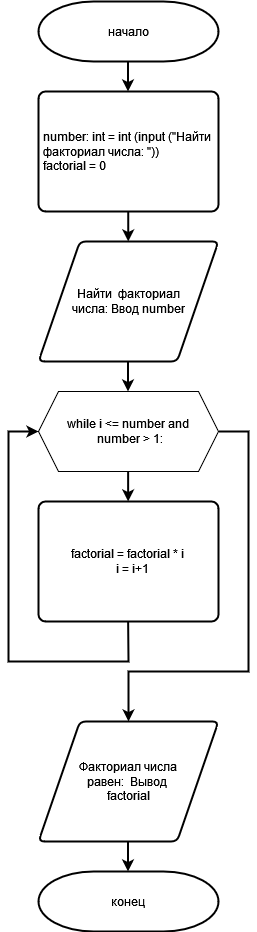
Отзыв, замечания\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Сочи 2024

Вариант 19

# Общие задания для всех вариантов

1. Нарисовать блок-схему алгоритма, написать и отладить программу для примера 1
2. Блок-схема



1. Код программы

number: int = int(input("Найти факториал числа: "))

i=1

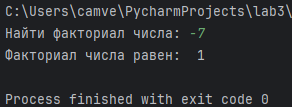
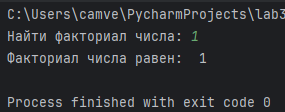
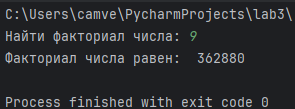
factorial=1

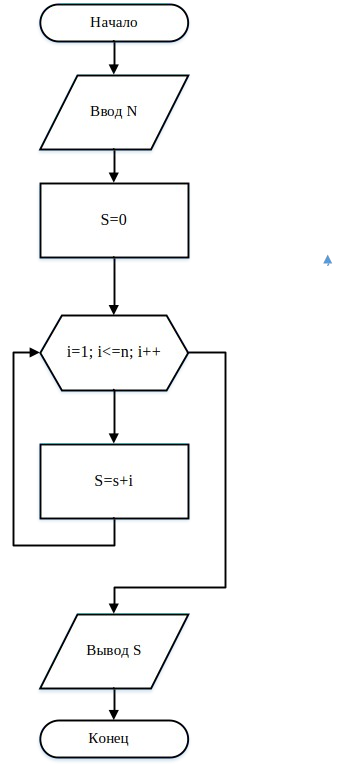
while i <= number and number>0 :

factorial=factorial\*i

i=i+1

print("Факториал числа равен: ", factorial)

1. Скриншоты выполнения программ  
     
     
     
     
     
   
2. Написать программу для блок-схемы алгоритма из примера 2.
3. Блок-схема



1. Код программы

n=int(input())

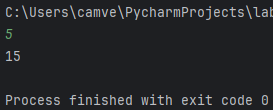
s=0

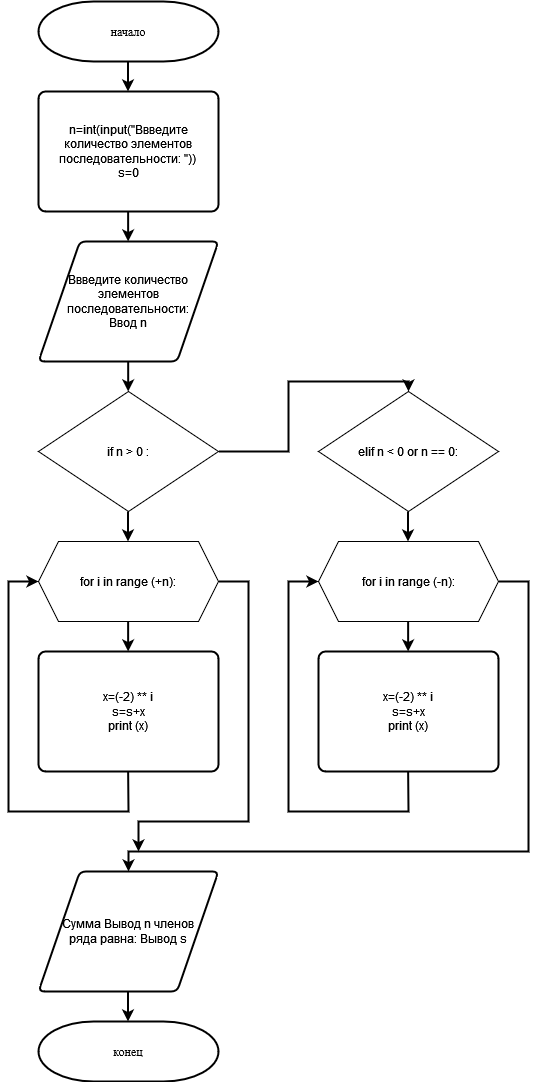
for i in range(1,n+1):

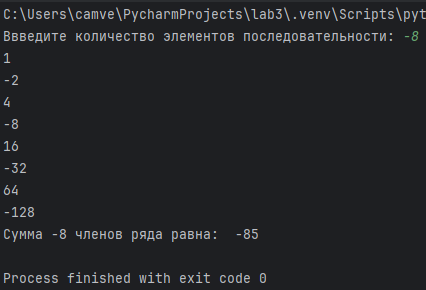
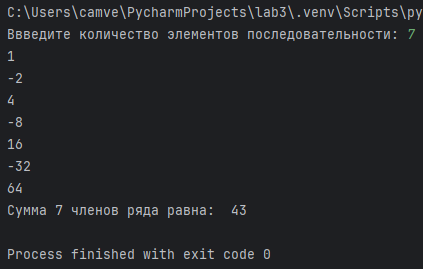
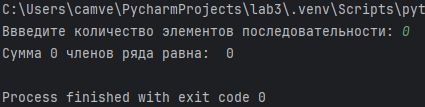
s=s+i

print(s)

1. Скриншоты выполнения программы



1. Нарисовать блок-схему алгоритма, написать и отладить программу для примера 3.
2. Блок-схема  
   

1. Код программы  
     
   n=int(input("Введите количество элементов последовательности: "))  
   s=0  
   if n > 0:  
    for i in range (+n):  
    x=(-2) \*\* i  
    s=s+x  
    print (x)  
   elif n < 0 or n == 0:  
    for i in range (-n):  
    x=(-2) \*\* i  
    s=s+x  
    print(x)  
   print ("Cyммa", n, "членов ряда равна: ", s)
2. Скриншоты выполнения программы  
     
     
     
     
     
   
3. Посчитать сумму ряда *N* √  для **N** членов. Распечатывать номер шага

∑

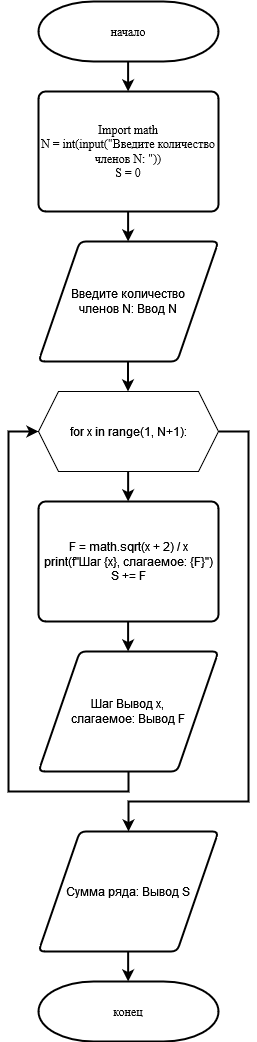
*x*

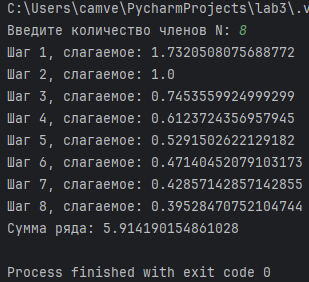
+

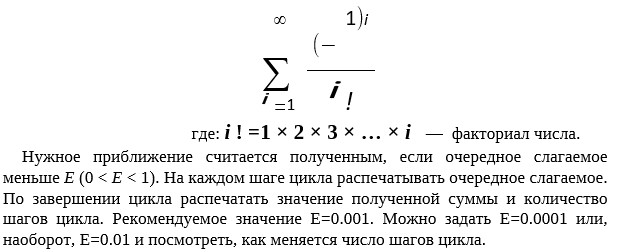
2

*x*=1 *x*

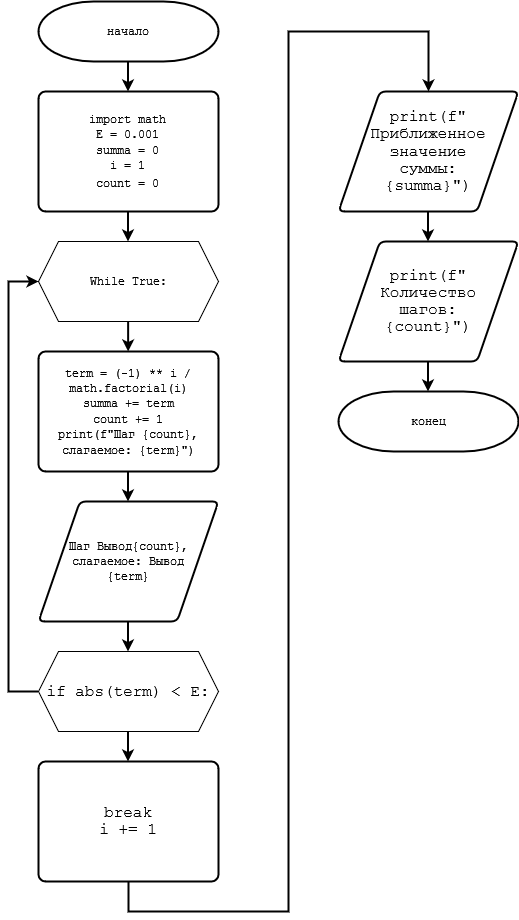
цикла и каждое слагаемое, а по завершении цикла - посчитанную сумму.

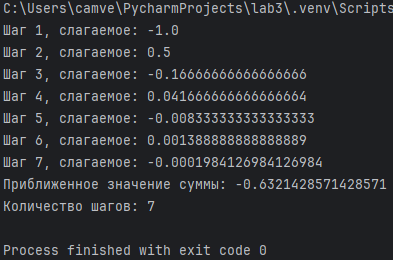
1. Блок-схема   
   
2. Код программы  
     
   import math  
   N = int(input("Введите количество членов N: "))  
   S = 0  
   for x in range(1, N+1):  
    F = math.sqrt(x + 2) / x  
    print(f"Шаг {x}, слагаемое: {F}")  
    S += F  
   print(f"Сумма ряда: {S}")

1. Скриншоты выполнения программы  
     
   
2. Рассчитать приближенное значение бесконечной суммы:



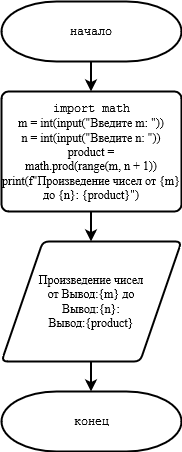
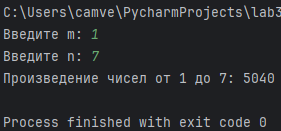
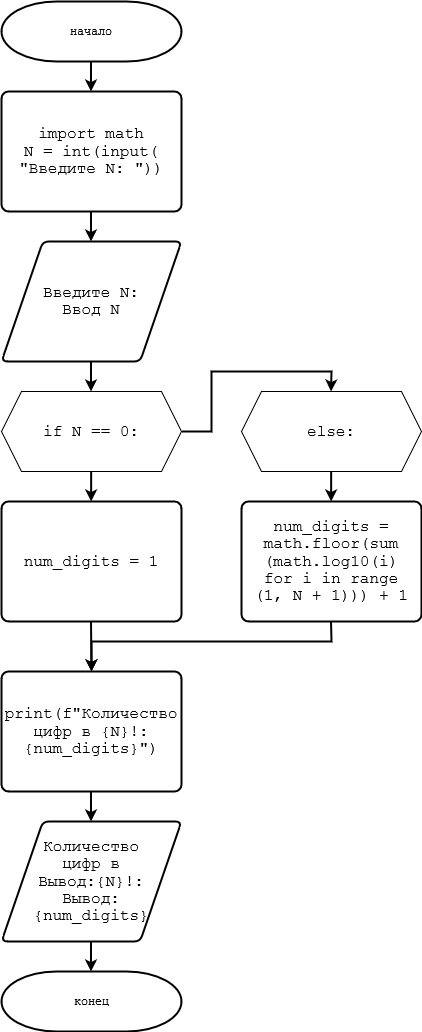
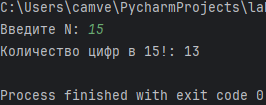
1. Блок-схема

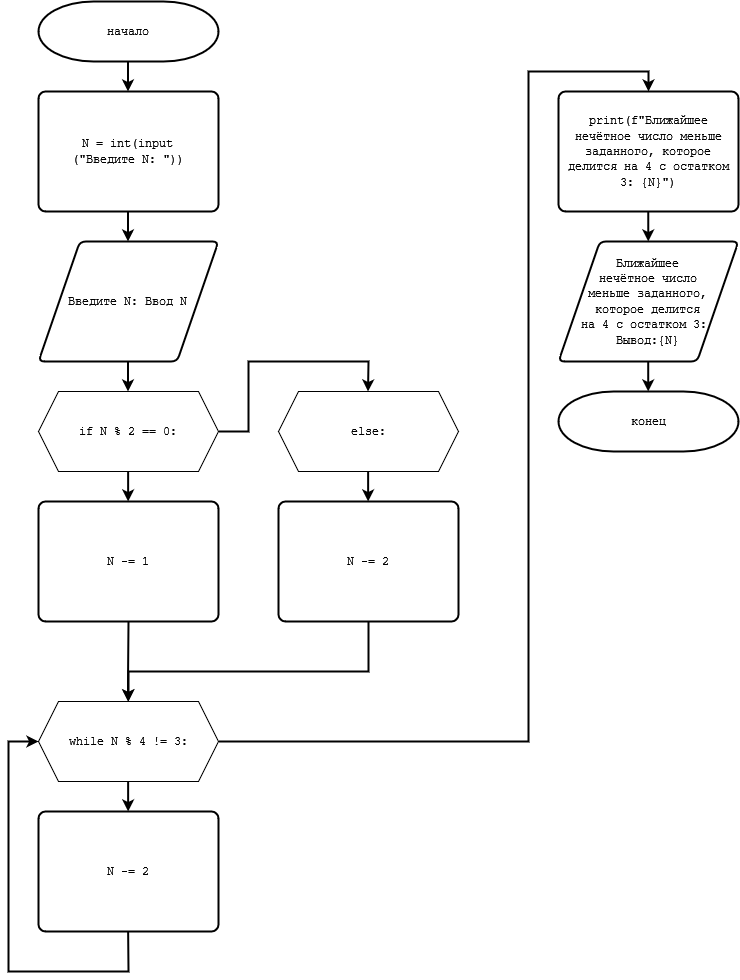


1. Код программы  
     
   import math  
     
   E = 0.001  
   summa = 0  
   i = 1  
   count = 0  
     
   while True:  
    term = (-1) \*\* i / math.factorial(i)  
    summa += term  
    count += 1  
    print(f"Шаг {count}, слагаемое: {term}")  
     
    if abs(term) < E:  
    break  
    i += 1  
     
   print(f"Приближенное значение суммы: {summa}")  
   print(f"Количество шагов: {count}")
2. Скриншоты выполнения программы  
     
   

# Задания по вариантам

Вариант 19

1. Перемножить все числа от **m** до **n**, где **m, n** — целые числа, **m < n**.
2. Блок-схема   
   
3. Код программы  
     
   import math  
     
   m = int(input("Введите m: "))  
   n = int(input("Введите n: "))  
     
   product = math.prod(range(m, n + 1))  
     
   print(f"Произведение чисел от {m} до {n}: {product}")
4. Скриншоты выполнения программы  
     
   
5. Определить, сколько цифр в числе **N! (N!=1\*2\*3\*…\*N). N –** вводится с консоли.
6. Блок-схема   
   
7. Код программы  
     
   import math  
     
   N = int(input("Введите N: "))  
     
   if N == 0:  
    num\_digits = 1  
   else:  
    num\_digits = math.floor(sum(math.log10(i) for i in range(1, N + 1))) + 1  
     
   print(f"Количество цифр в {N}!: {num\_digits}")
8. Скриншоты выполнения программы  
     
   
9. Определить ближайшее нечетное число меньше заданного числа **N**, которое делится на 4 с остатком 3. **N –** вводится с консоли
10. Блок-схема



1. Код программы  
     
   N = int(input("Введите N: "))

if N % 2 == 0:

N -= 1

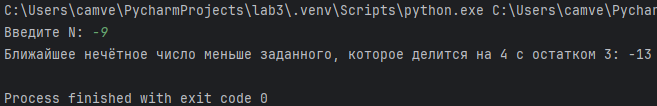
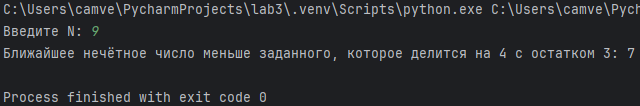
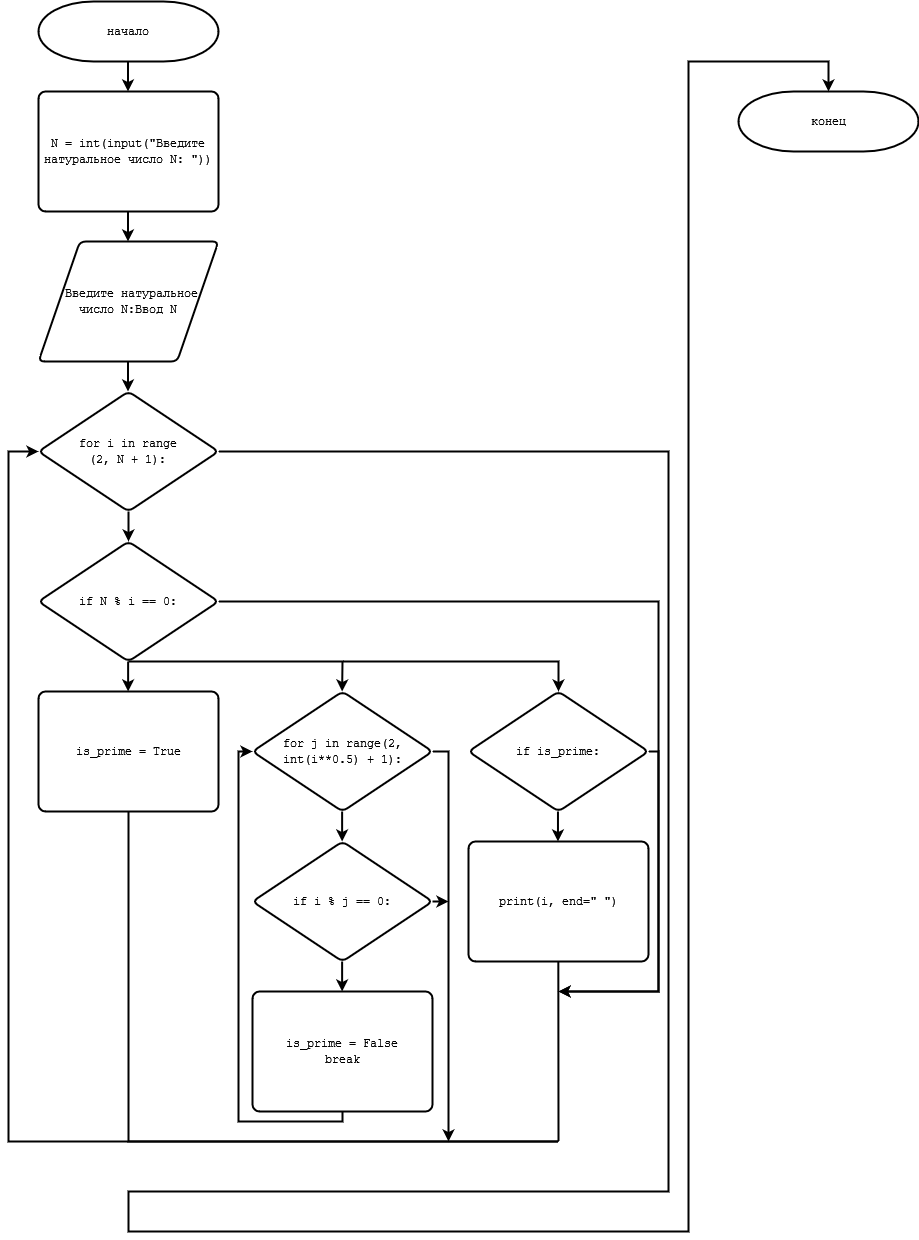
else:

N -= 2

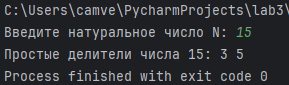
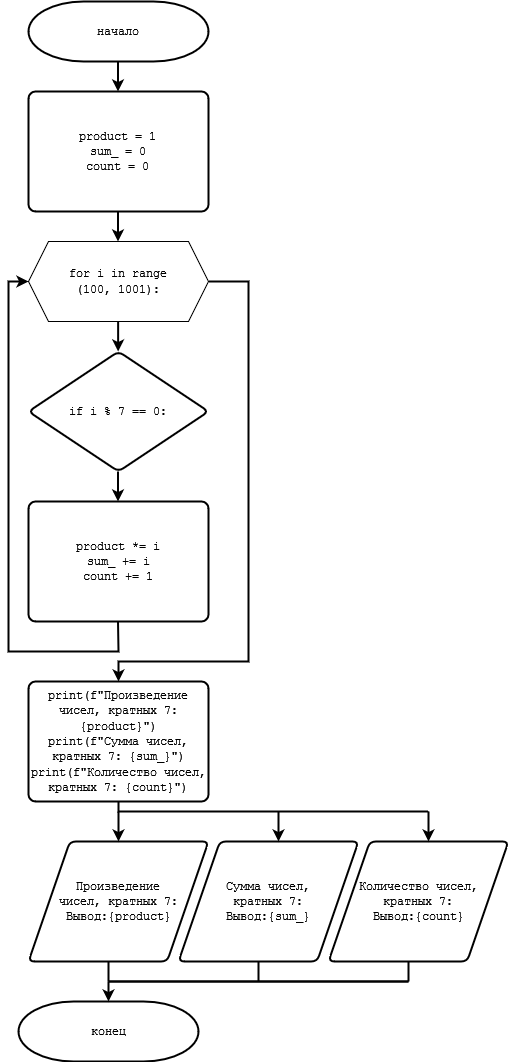
while N % 4 != 3:

N -= 2

print(f"Ближайшее нечётное число меньше заданного, которое делится на 4 с остатком 3: {N}")

1. Скриншоты выполнения программы  
     
     
     
   
2. Вводится натуральное число N. Напечатать все простые делители этого числа. Простые делители – это числа, на которые N делится без остатка.
3. Блок-схема  
   
4. Код программы  
     
   N = int(input("Введите натуральное число N: "))  
     
   print(f"Простые делители числа {N}:", end=" ")  
     
   for i in range(2, N + 1):  
    if N % i == 0:

is\_prime = True  
 for j in range(2, int(i\*\*0.5) + 1):  
 if i % j == 0:  
 is\_prime = False  
 break  
 if is\_prime:  
 print(i, end=" ")

1. Скриншоты выполнения программы  
     
   
2. Вычислить произведение чисел кратных 7 от 100 до 1000. Итоги вывода: сумма и количество чисел.
3. Блок-схема  
   
4. Код программы  
     
   product = 1

sum\_ = 0

count = 0

for i in range(100, 1001):

if i % 7 == 0:

product \*= i

sum\_ += i

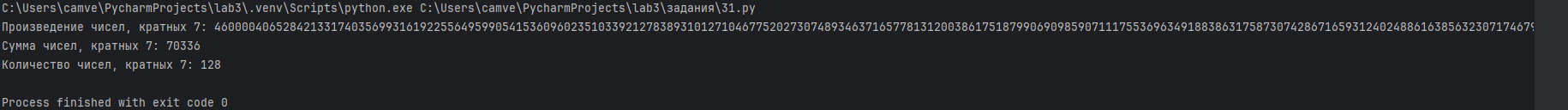
count += 1

print(f"Произведение чисел, кратных 7: {product}")

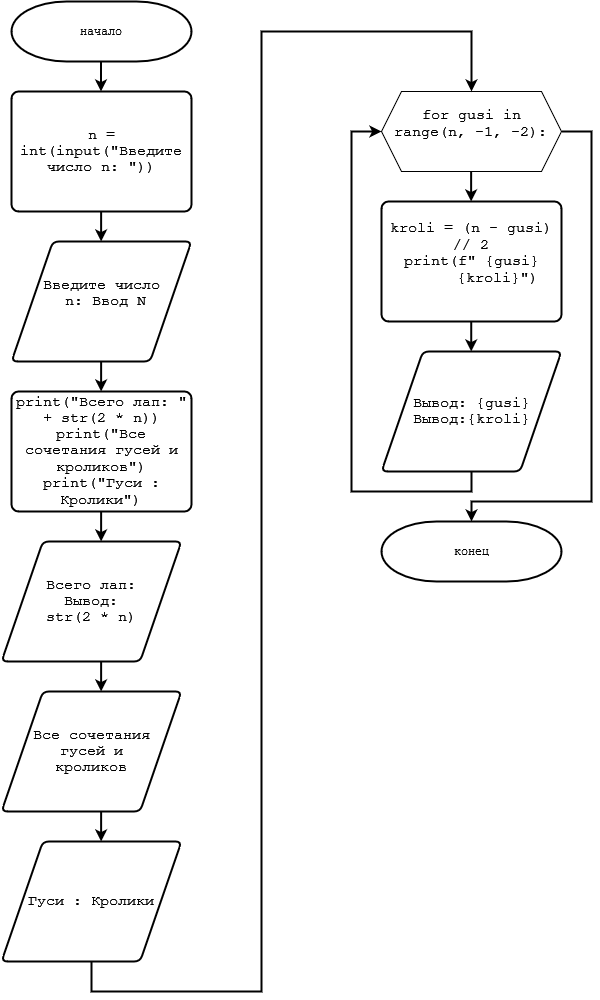
print(f"Сумма чисел, кратных 7: {sum\_}")

print(f"Количество чисел, кратных 7: {count}")

1. Скриншоты выполнения программы:



# Бонусное задание

1. У гусей и кроликов вместе 2\*N лап. N – вводится с консоли. Сколько может быть гусей и кроликов (вывести все возможные сочетания).
2. Блок-схема  
   
3. Код программы  
     
   n = int(input("Введите число n: "))  
   print("Всего лап: " + str(2 \* n))  
   print("Все сочетания гусей и кроликов")  
   print("Гуси : Кролики")  
   for gusi in range(n, -1, -2):  
    kroli = (n - gusi) // 2  
    print(f" {gusi} {kroli}")
4. Скриншоты выполнения программы:  
     
   