

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

Основы кроссплатформенного программирования

Отчет по лабораторной работе №2.2

Условные операторы и циклы в языке Python

Выполнил студент группы

ИВТ-б-о-21-1

Мальцев Н.А. « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил доцент

Кафедры инфокоммуникаций, старший
преподаватель

Воронкин Р.А.

(подпись)

Ставрополь 2022

Условные операторы и циклы в языке Python.

Цель работы: приобретение навыков программирования разветвляющихся алгоритмов и алгоритмов циклической структуры. Освоить операторы языка Python версии 3.10 if, while, for, break и continue, позволяющих реализовывать разветвляющиеся алгоритмы и алгоритмы циклической структуры.

Порядок выполнения работы:

1) Проработка примеров:

Пример 1:

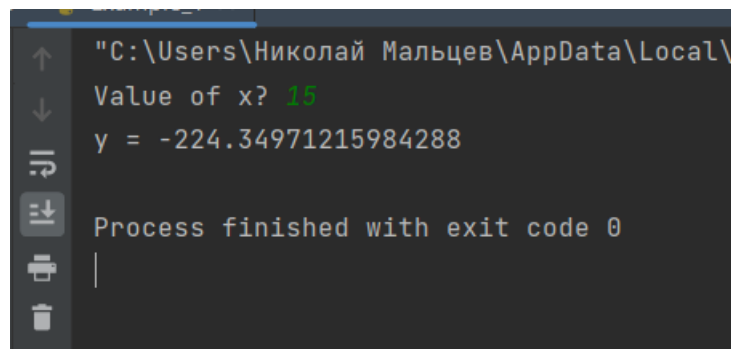
Код программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import math

if __name__ == '__main__':
    x = float(input("Value of x? "))
    if x <= 0:
        y = 2 * x * x + math.cos(x)
    elif x < 5:
        y = x + 1
    else:
        y = math.sin(x) - x * x
    print(f"y = {y}")
```

Результат работы программы:



```
"C:\Users\Николай Мальцев\AppData\Local\...
Value of x? 15
y = -224.34971215984288
Process finished with exit code 0
```

Рисунок 1. Результат работы программы из примера 1

UML-диаграмма деятельности:

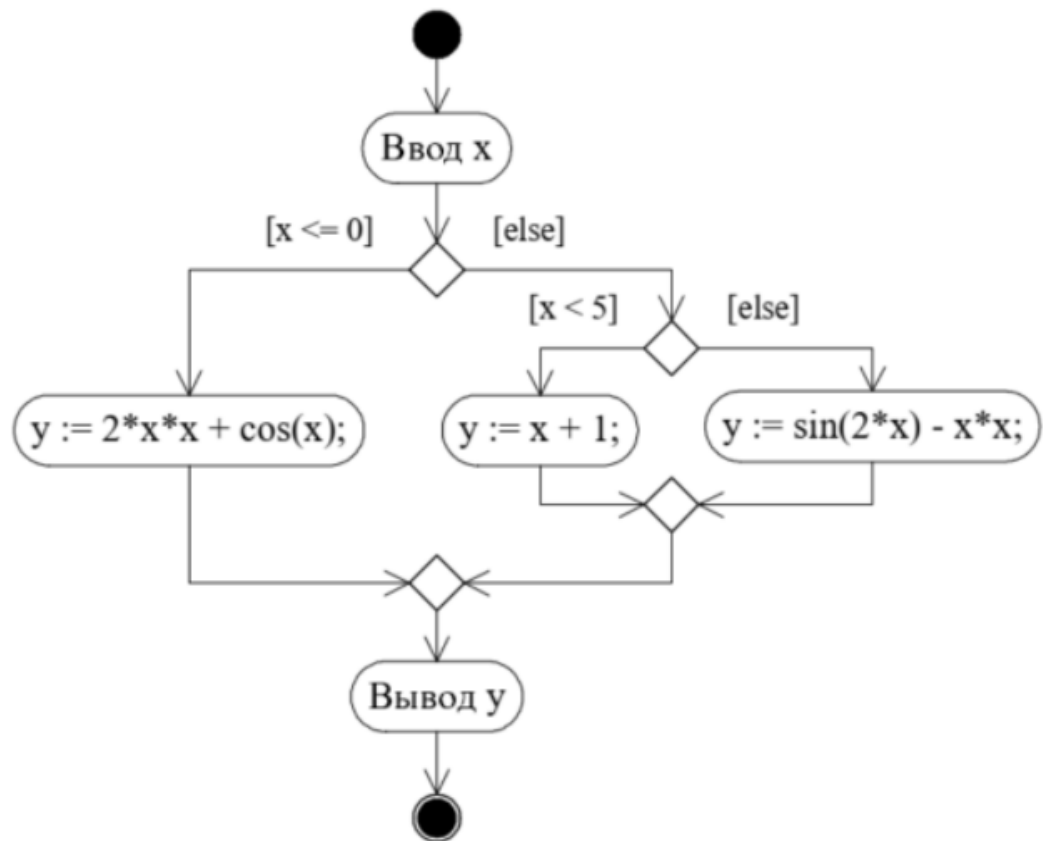


Рисунок 2. UML-диаграмма деятельности из примера 1

Пример 2:

Код программы:

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import sys

if __name__ == '__main__':
    n = int(input("Введите номер месяца: "))

    if n == 1 or n == 2 or n == 12:
        print("Зима")
    elif n == 3 or n == 4 or n == 5:
        print("Весна")
    elif n == 6 or n == 7 or n == 8:
        print("Лето")
    elif n == 9 or n == 10 or n == 11:
        print("Осень")
    else:
        print("Ошибка!", file=sys.stderr)
        exit(1)
  
```

Результат работы программы:

```
↑ "C:\Users\Николай Мальцев\AppData\Local
↓ Введите номер месяца: 10
↕ Осень
↔ Process finished with exit code 0
✕
```

Рисунок 3. Результат работы программы из примера 2

UML-диаграмма деятельности:

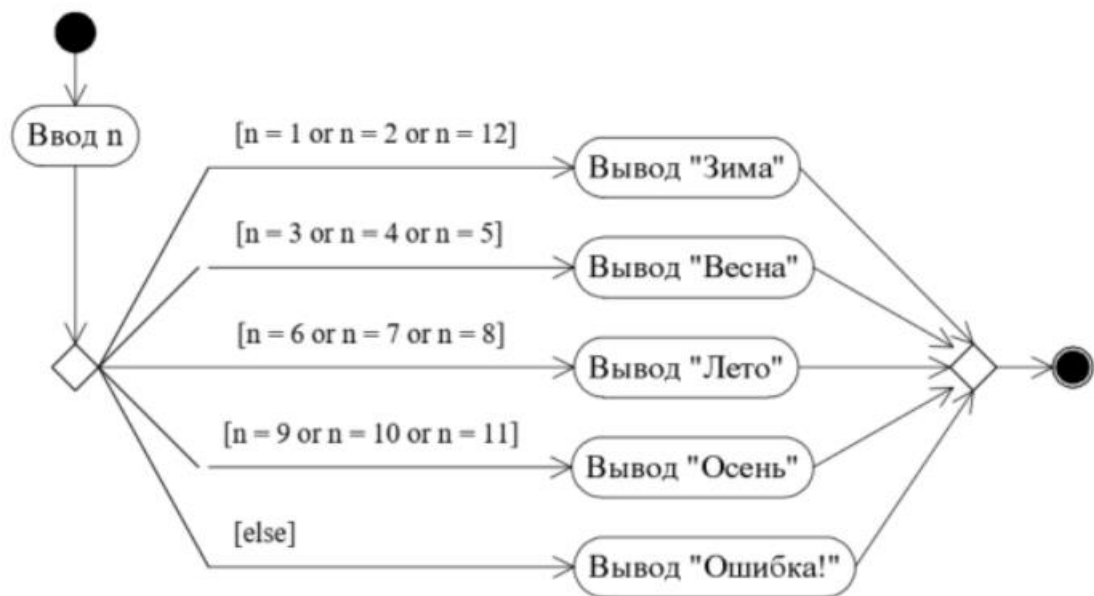


Рисунок 4. UML-диаграмма деятельности из примера 2

Пример 3:

Код программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import math

if __name__ == '__main__':
    n = int(input("Value of n? "))
    x = float(input("Value of x? "))

    S = 0.0
    for k in range(1, n + 1):
        a = math.log(k * x) / (k * k)
        S += a

    print(f"S = {S}")
```

Результат работы программы:

```
"C:\Users\Николай Мальцев\AppData\Loc
Value of n? 4
Value of x? 5
S = 2.6732119195688706
Process finished with exit code 0
```

Рисунок 5. Результат работы программы из примера 3

UML-диаграмма деятельности:

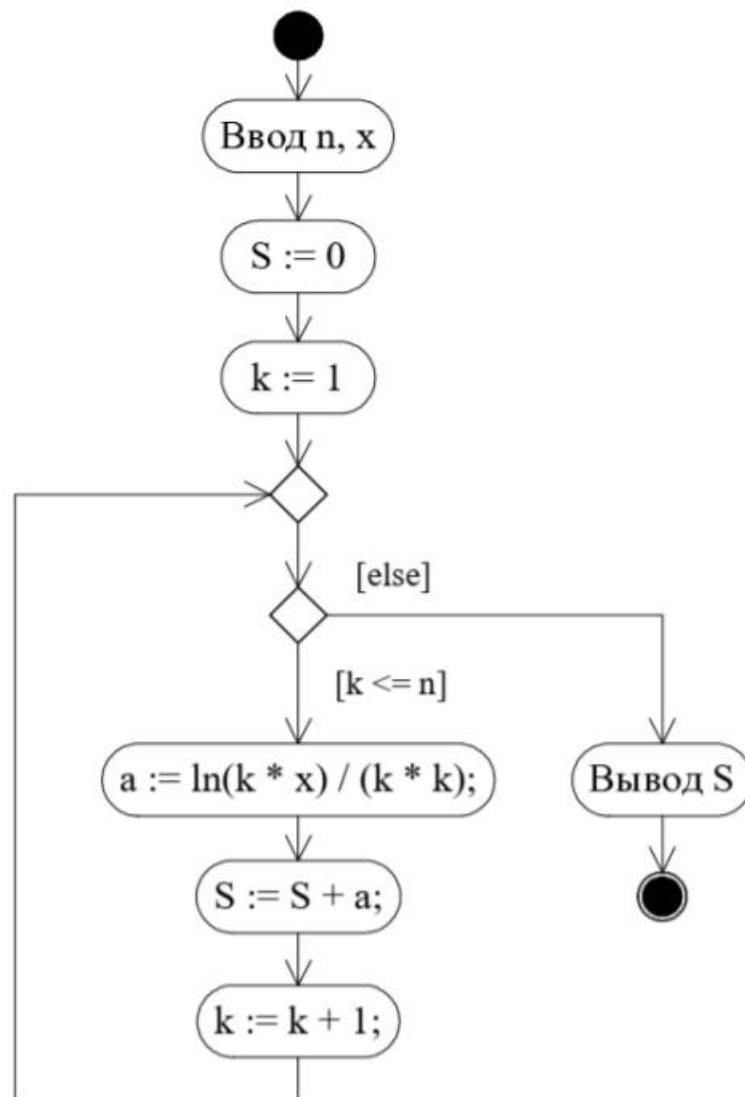


Рисунок 6. UML-диаграмма деятельности из примера 3

Пример 4:

Код программы:

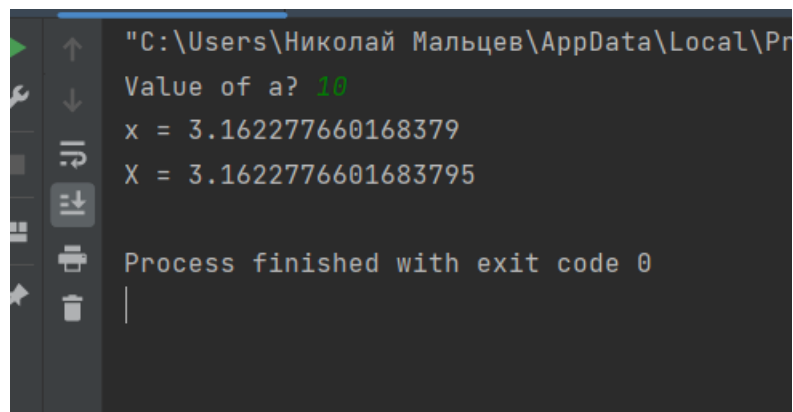
```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import math
import sys

if __name__ == '__main__':
    a = float(input("Value of a? "))
    if a < 0:
        print("Illegal value of a", file=sys.stderr)
        exit(1)
    x = 1
    eps = 1e-10
    while True:
        xp = x
        x = (x + a / x) / 2
        if math.fabs(x - xp) < eps:
            break

    print(f"x = {x}\nX = {math.sqrt(a)}")
```

Результат работы программы:



```
"C:\Users\Николай Мальцев\AppData\Local\Pr
Value of a? 10
x = 3.162277660168379
X = 3.1622776601683795
Process finished with exit code 0
```

Рисунок 7. Результат работы программы из примера 4

Пример 5:

Код программы:

```
import math
import sys

# Постоянная Эйлера.
EULER = 0.5772156649015328606
# Точность вычислений.
EPS = 1e-10

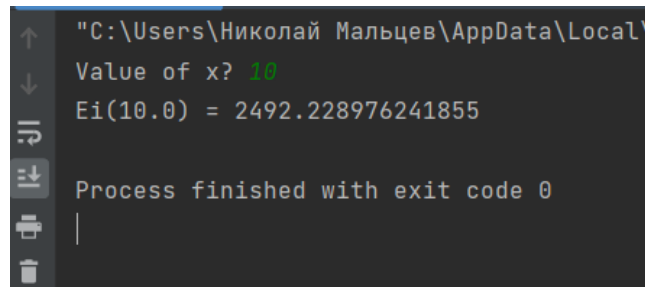
if __name__ == '__main__':
    x = float(input("Value of x? "))
    if x == 0:
        print("Illegal value of x", file=sys.stderr)
        exit(1)

    a = x
    S, k = a, 1
```

```
# Найти сумму членов ряда.
while math.fabs(a) > EPS:
    a *= x * k / (k + 1) ** 2
    S += a
    k += 1

# Вывести значение функции.
print(f"Ei({x}) = {EULER + math.log(math.fabs(x)) + S}")
```

Результат работы программы:



```
"C:\Users\Николай Мальцев\AppData\Local\
Value of x? 10
Ei(10.0) = 2492.228976241855
Process finished with exit code 0
```

Рисунок 8. Результат работы программы из примера 5

2) Выполнение индивидуального задания (Вариант 11):

Задание 1:

Код программы:

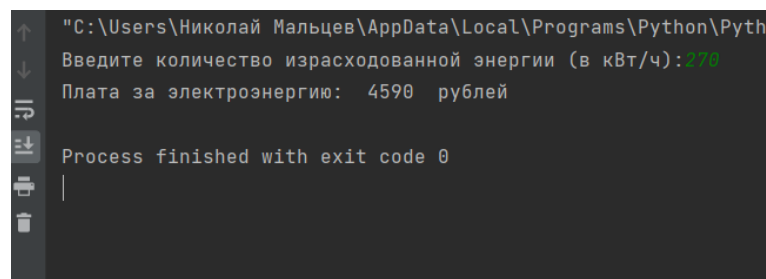
```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == '__main__':

    n = int(input("Введите количество израсходованной энергии (в кВт/ч):"))

    if n <= 250:
        print("Плата за электроэнергию: ", n*7, " рублей")
    elif (n > 250) and (n < 300):
        print("Плата за электроэнергию: ", n * 17, " рублей")
    elif n >= 300:
        print("Плата за электроэнергию: ", n * 20, " рублей")
```

Результат работы программы:



```
"C:\Users\Николай Мальцев\AppData\Local\Programs\Python\Pyth
Введите количество израсходованной энергии (в кВт/ч):270
Плата за электроэнергию: 4590 рублей
Process finished with exit code 0
```

Рисунок 9. Результат выполнения программы к заданию 1

UML-диаграмма деятельности:

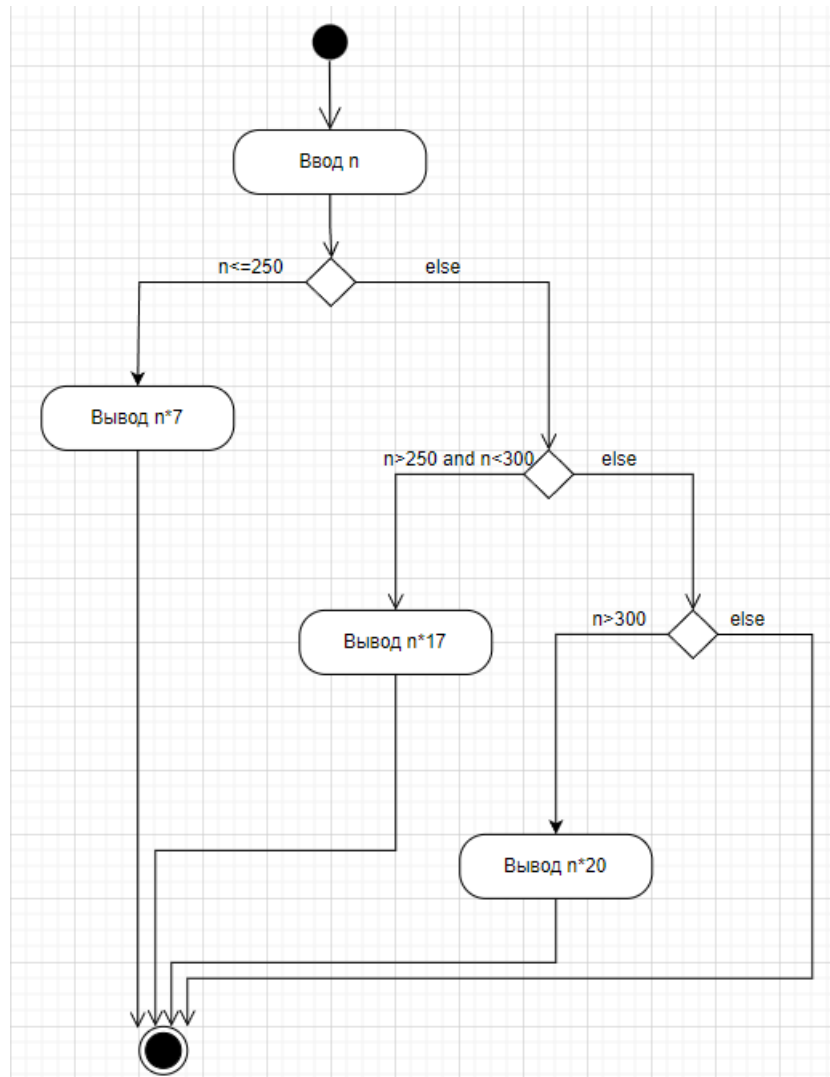


Рисунок 10. UML-диаграмма деятельности к заданию 1

Задание 2:

Код программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == '__main__':
    a = int(input("Введите три числа: "))
    b = int(input())
    c = int(input())

    if (a % 2 == 0) or (b % 2 == 0) or (c % 2 == 0):
        print("Однo из чисел четное.")
    else:
        print("Среди введенных чисел нет четных чисел.")
```

Результат работы программы:


```
"C:\Users\Николай Мальцев\AppData\Local\Pr
Введите три числа: 2
5
10
Одо из чисел четное.

Process finished with exit code 0
```

Рисунок 11. Результат работы программы к заданию 2

UML-диаграмма деятельности:

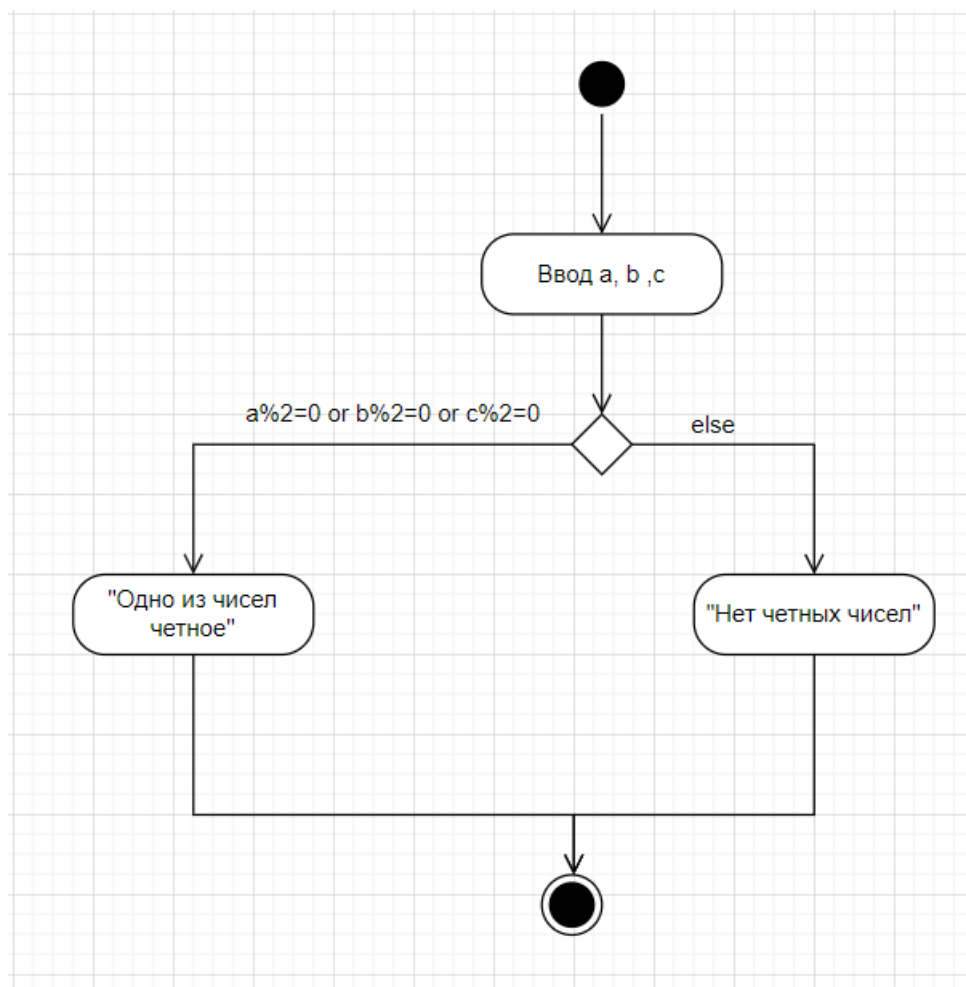


Рисунок 12. UML-диаграмма деятельности к заданию 2

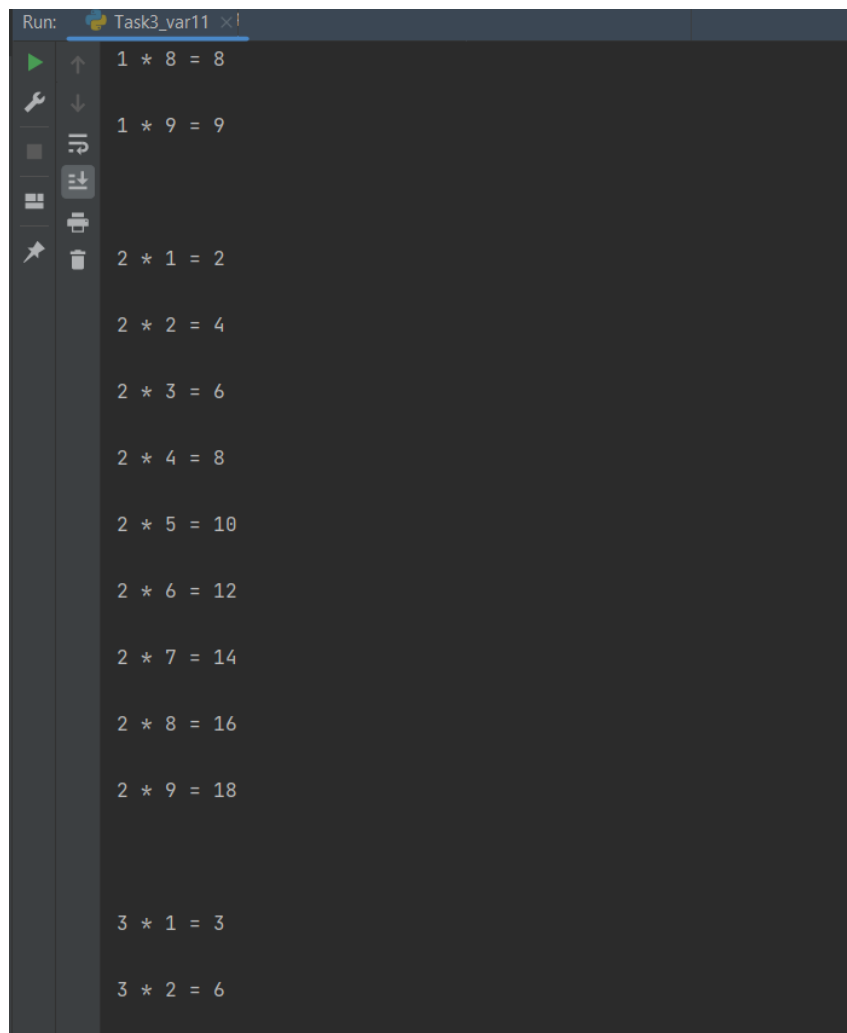
Задание 3:

Код программы:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
```

```
if __name__ == '__main__':  
    for i in range(1, 10):  
        for j in range(1, 10):  
            print(i, "*", j, "=", i*j, "\n")  
        print("\n")
```

Результат работы программы:



```
Run: Task3_var11 x!  
↑ 1 * 8 = 8  
↓ 1 * 9 = 9  
↕  
↓ 2 * 1 = 2  
2 * 2 = 4  
2 * 3 = 6  
2 * 4 = 8  
2 * 5 = 10  
2 * 6 = 12  
2 * 7 = 14  
2 * 8 = 16  
2 * 9 = 18  
3 * 1 = 3  
3 * 2 = 6
```

Рисунок 3. Результат работы программы к заданию 3

UML-диаграмма деятельности:

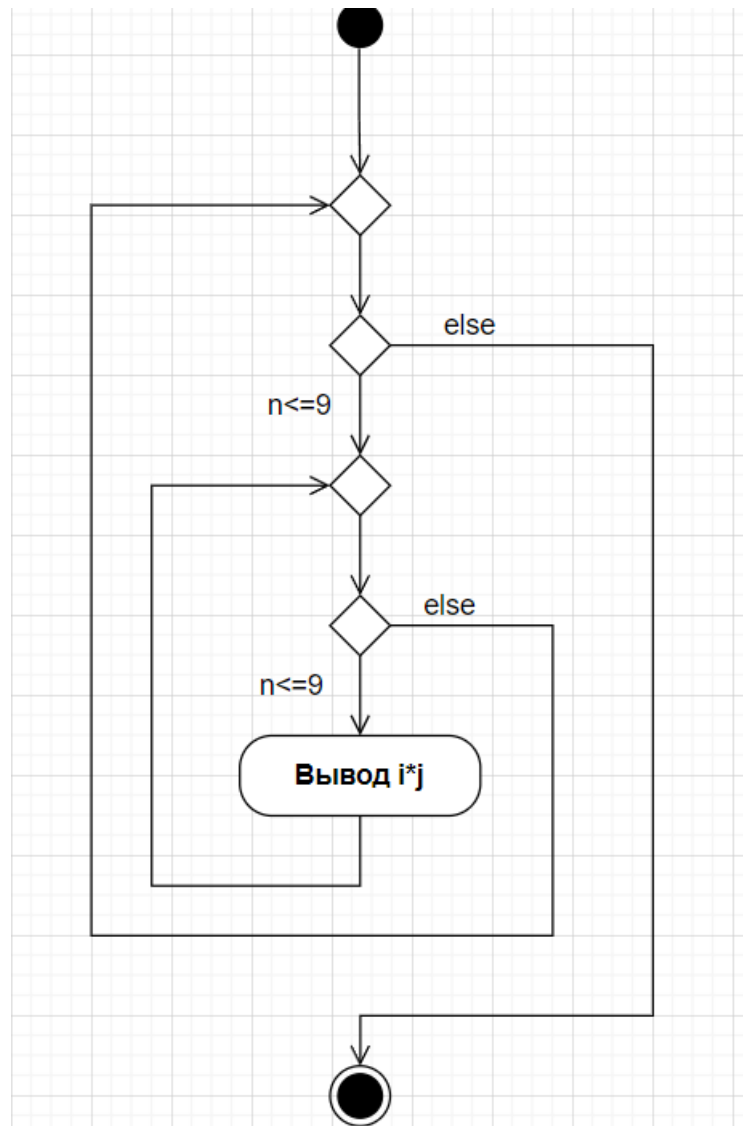


Рисунок 4. UML-диаграмма деятельности к заданию 3

Вывод: в ходе работы были приобретены навыки программирования разветвляющихся алгоритмов и алгоритмов циклической структуры. Освоены операторы языка Python версии 3.10 if, while, for, break и continue, позволяющих реализовать разветвляющиеся алгоритмы и алгоритмы циклической структуры.