

Міністерство освіти і науки України

Донецький національний університет імені Василя Стуса Факультет

інформаційних і прикладних технологій

Кафедра інформаційних технологій

## **ЗВІТ**

з лабораторної роботи № 7

з дисципліни «Основи програмування»

на тему:

«Створення користувацьких функцій у мові Python»

Виконав: студент гр. Б25\_д/F3

Кручиківський Ю.О.

Перевірив: доц. Бабаков Р. М.

Вінниця – 2025

## Завдання 1.

7

Деяке тризначне число склали з його числом-паліндромом і отримали число N. Знайти для заданого числа N відповідне тризначне число

Лістинг до завдання 1.

```
while True:
    try:
        number = input("Введіть тризначне число: ")

        if len(number) != 3:
            raise ValueError("Введено некоректне число")

        number = int(number)
        break

    except ValueError as error:
        print(error)

def find_palindrome(n: int) -> int: # задамо ф-цію для пошуку паліндрому
    числа
    return int(str(n)[::-1]) # Перетворимо число в текст інвертуємо його
    після чого повернемо в число

def brut_force_equation(
    func): # задамо ф-цію для перебору значень з заданим рівнянням
    котре задається безіменною ф-цією
    for x in range(1, 1000): # переберемо всі можливі числа
        if func(x): # перевіримо відповідність числа до рівняння
            return x

print("Початкове число сума якого з паліндромом дорівнює
{}".format(number))
palindrome = brut_force_equation( # виклечемо ф-цію
    lambda x: (x + find_palindrome(x)) == number # задамо рівняння у
    форматі безіменної ф-ції
)

if palindrome:
    print(palindrome)
else:
    print("Не знайдено")
```

Результат роботи програми 1.

```
Введіть тризначне число: 565
Початкове число сума якого з паліндромом дорівнює 565
134
```

Блок схема до програми 1.

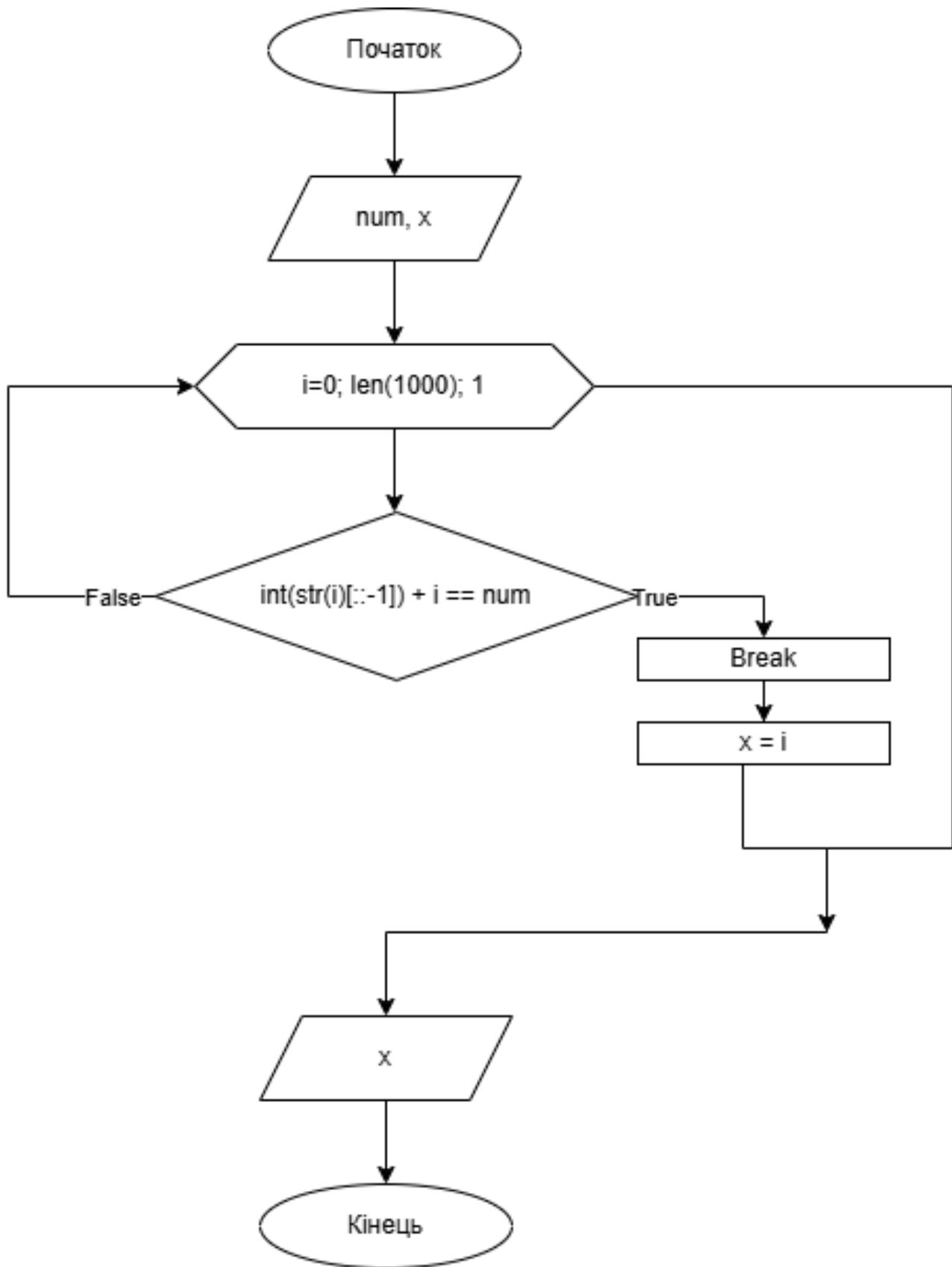


Рисунок 1 – Блок схема до завдання 1

## Завдання 2.

7

Початковими даними для цього завдання є лабораторна робота № 4. Вона містить три завдання, присвячені операціям розгалуження. Необхідно реалізувати кожне завдання у вигляді окремої функції. Треба самостійно обрати кількість та імена аргументів функцій, а також продумати, чи можна зробити значення якихось аргументів «за замовчуванням». Усі функції мають бути самостійні і незалежні одна від одної. В основній програмі продемонструвати декілька прикладів виклику функцій.

Лістинг до завдання 2.

```
"""
Завдання взяте з лабораторної роботи номер 4
"""

# Переробка завдання 1
class BirthDate:
    def __init__(self, birthday_string: str):
        if len(birthday_string) != 8:
            raise ValueError

        self.birthday = birthday_string
        self.day = int(birthday_string[:2:])
        self.month = int(birthday_string[2:4:])
        self.year = int(birthday_string[4::])

def solution1(A, B, birthday: BirthDate):
    if A == birthday.day:
        print("A == дню народження")
    else:
        print("A не != дню народження")

    if B == birthday.day:
        print("B == дню народження")
    else:
        print("B не != дню народження")

    if A == birthday.month:
        print("A == місяцю народження")
    else:
        print("A не != місяцю народження")

    if B == birthday.month:
        print("B == місяцю народження")
    else:
        print("B не != місяцю народження")

    if A == birthday.year:
        print("A == року народження")
```

```

else:
    print("A != року народження")

if B == birthday.year:
    print("B == року народження")
else:
    print("B != року народження")


while True:
    try:
        A = int(input("Введіть A: "))
        B = int(input("Введіть B: "))

        birthday = BirthDate(input("Введіть дату народження (ddmmyyyy) :"))
        solution1(A, B, birthday)
        break

    except ValueError:
        print("Спробуйте знову")

```

# переробка завдання 2

```

def check_condition_2(variables: list):
    return len(variables) == 5

def solution2(variables: list):
    if check_condition_2(variables):
        count = 0

        for i in range(5):
            for j in range(5):
                if i != j and variables[j] != 0:
                    if variables[i] % variables[j] == 0:
                        count += 1
        return count

```

while True:

```

    try:
        variables = []

        for i in range(5):
            variables.append(int(input("Введіть значення для x{}: ".format(i + 1))))

        print(solution2(variables), "чисел, є дільниками інших чисел масиву")
        break

    except ValueError:
        print("Спробуйте знову")

```

# переробка завдання 3

```

def str_to_digit_list(num):
    return list(map(int, list(str(num))))

```

```

while True:
    try:
        A = input("Введіть А: ")
        B = input("Введіть В: ")

        a_base = max(str_to_digit_list(A)) + 1
        b_base = max(str_to_digit_list(B)) + 1

        print("Мінімально можлива числення для числа А:", a_base)
        print("Мінімально можлива числення для числа В:", b_base)

        print("Число А в десятковій системі:", int(A, a_base))
        print("Число В в десятковій системі:", int(B, b_base))
        break

    except ValueError:
        print("Спробуйте знову")

```

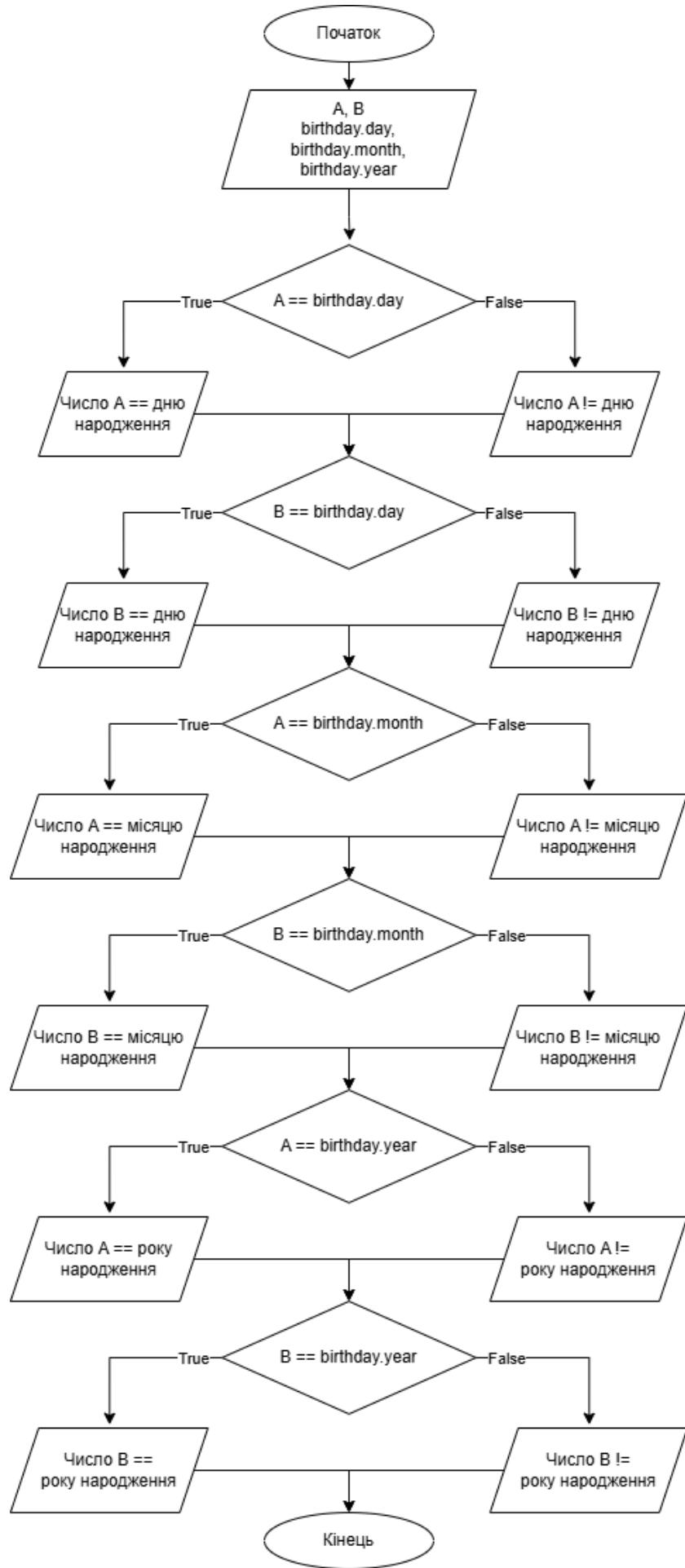
Результат виконання програми завдання 2.

```

Введіть А: 4
Введіть В:
Спробуйте знову
Введіть А: 4
Введіть В: 5
Введіть дату народження (ddmmyyyy) :01022006
А не != дню народження
В не != дню народження
А не != місяцю народження
В не != місяцю народження
А != року народження
В != року народження
Введіть значення для x1: 1
Введіть значення для x2: 2
Введіть значення для x3: 3
Введіть значення для x4: 4
Введіть значення для x5:
Спробуйте знову
Введіть значення для x1: 1
Введіть значення для x2: 2
Введіть значення для x3: 5
Введіть значення для x4: 8
Введіть значення для x5: 6
6 чисел, є дільниками інших чисел масиву
Введіть А: 1
Введіть В: 5
Мінімально можлива числення для числа А: 2
Мінімально можлива числення для числа В: 6
Число А в десятковій системі: 1
Число В в десятковій системі: 5

```

Блок схеми до завдання 2.



Блок-схема 2.1

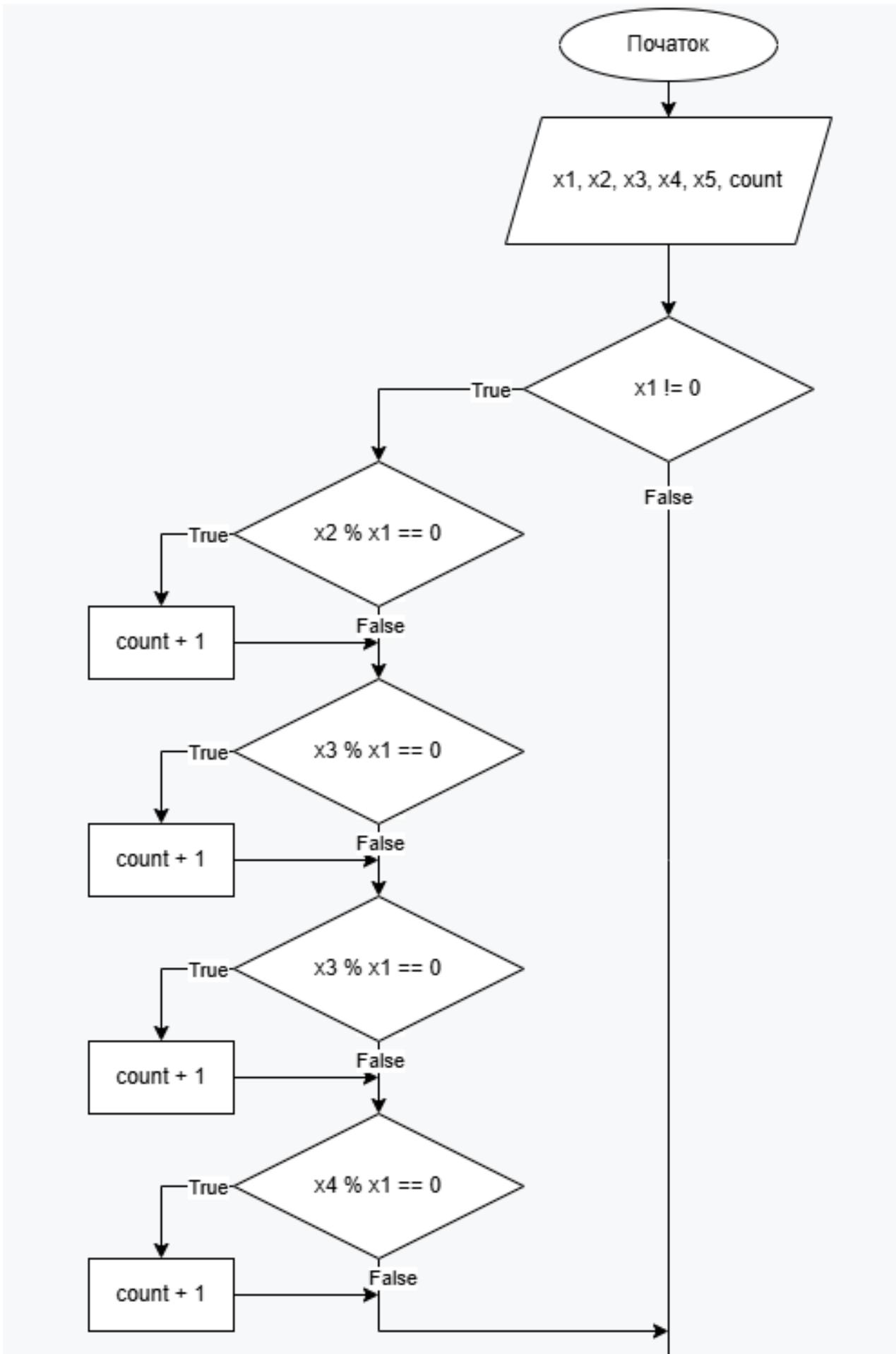


Рисунок 2 - 2.2 ч1

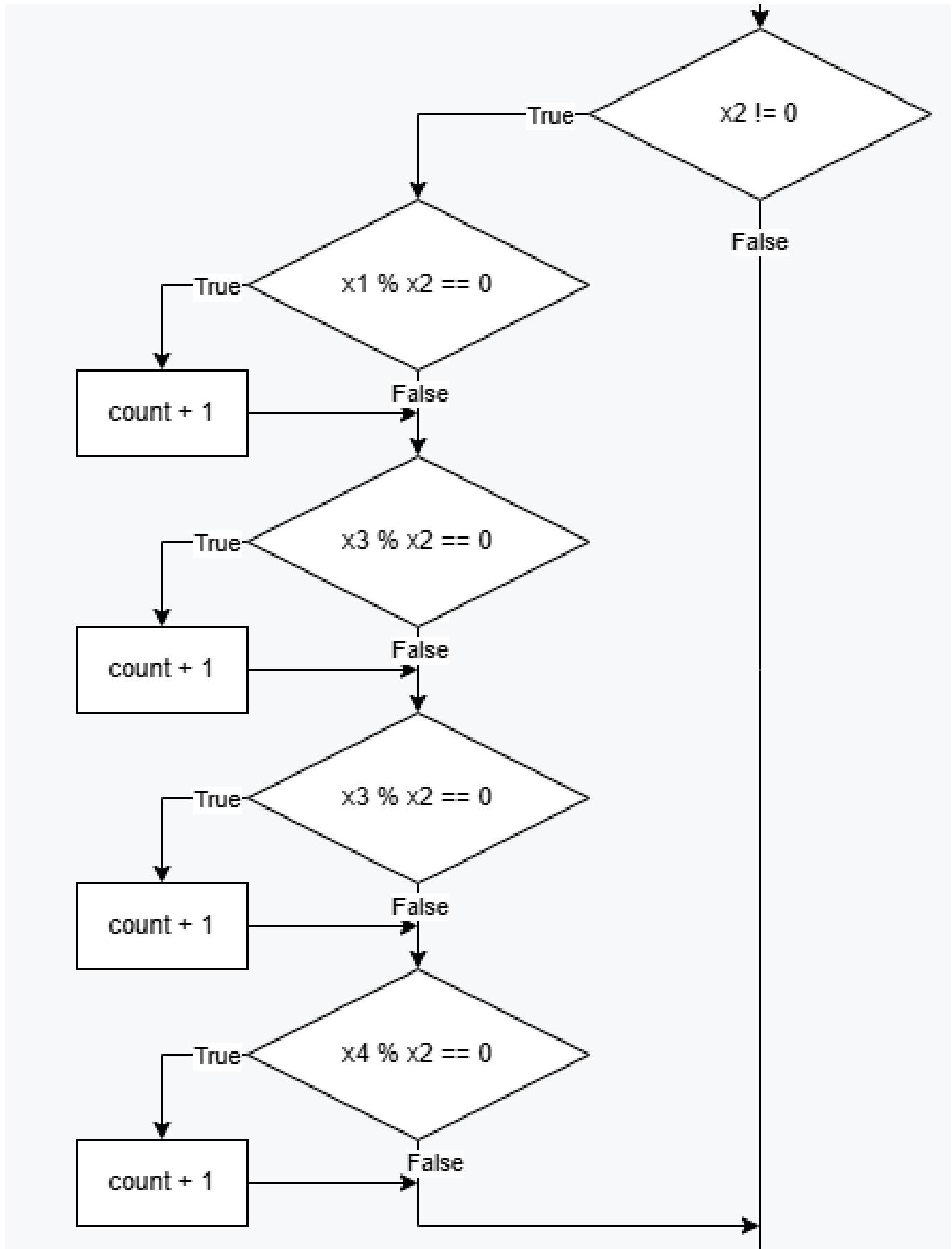


Рисунок 3 - 2.2 ч2

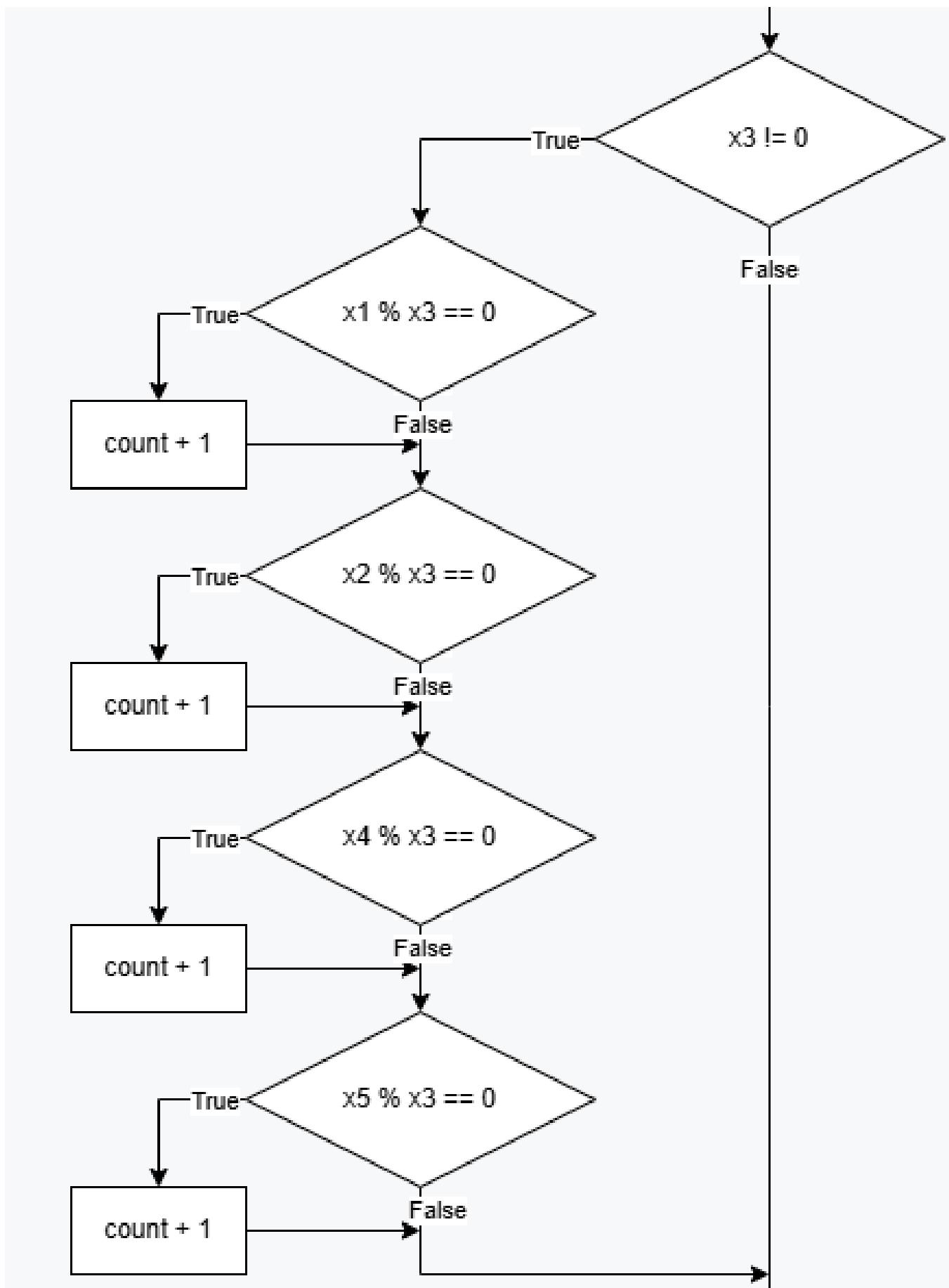


Рисунок 4 - 2.2 ч3

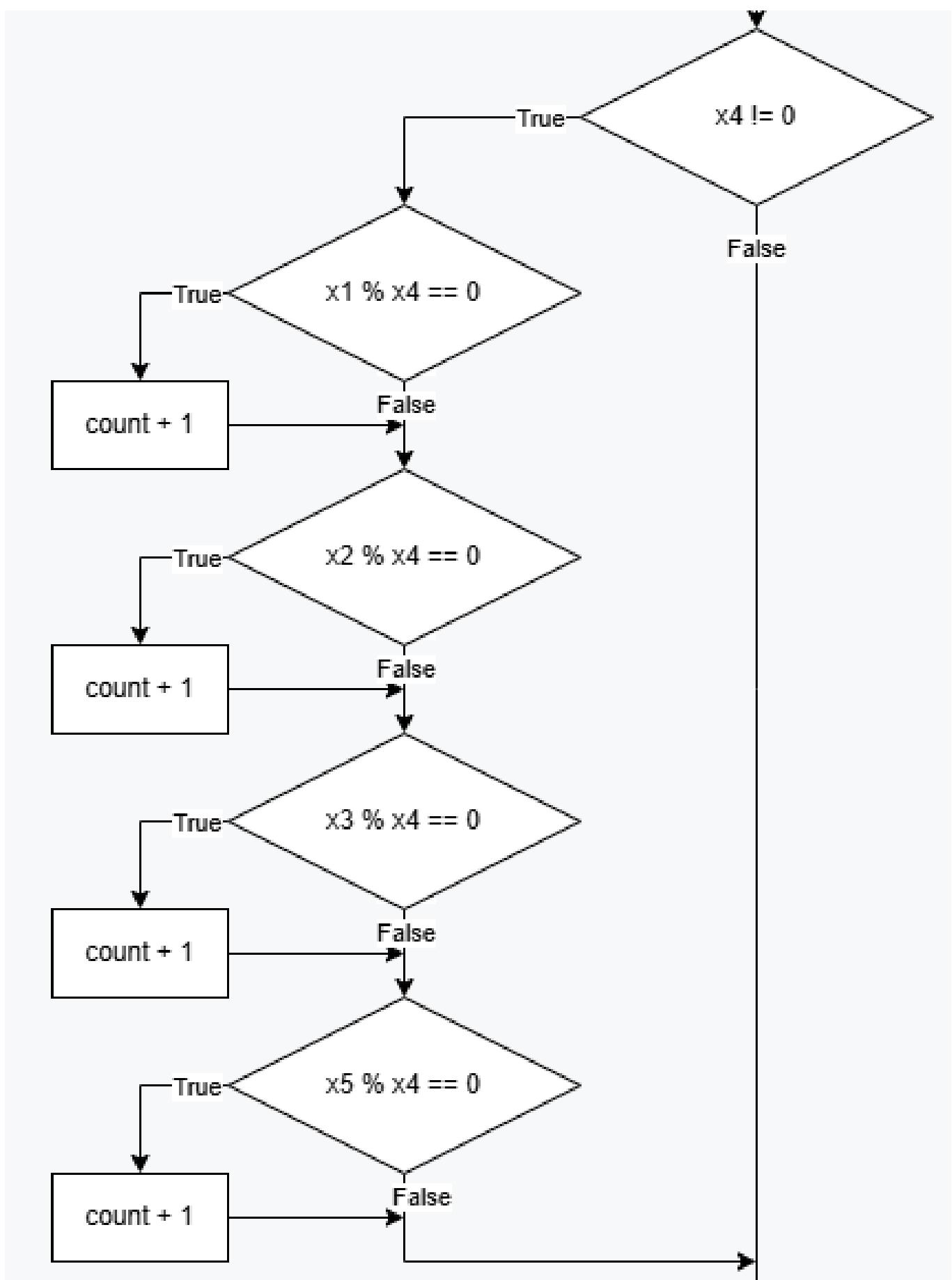


Рисунок 5 - 2.2 ч4

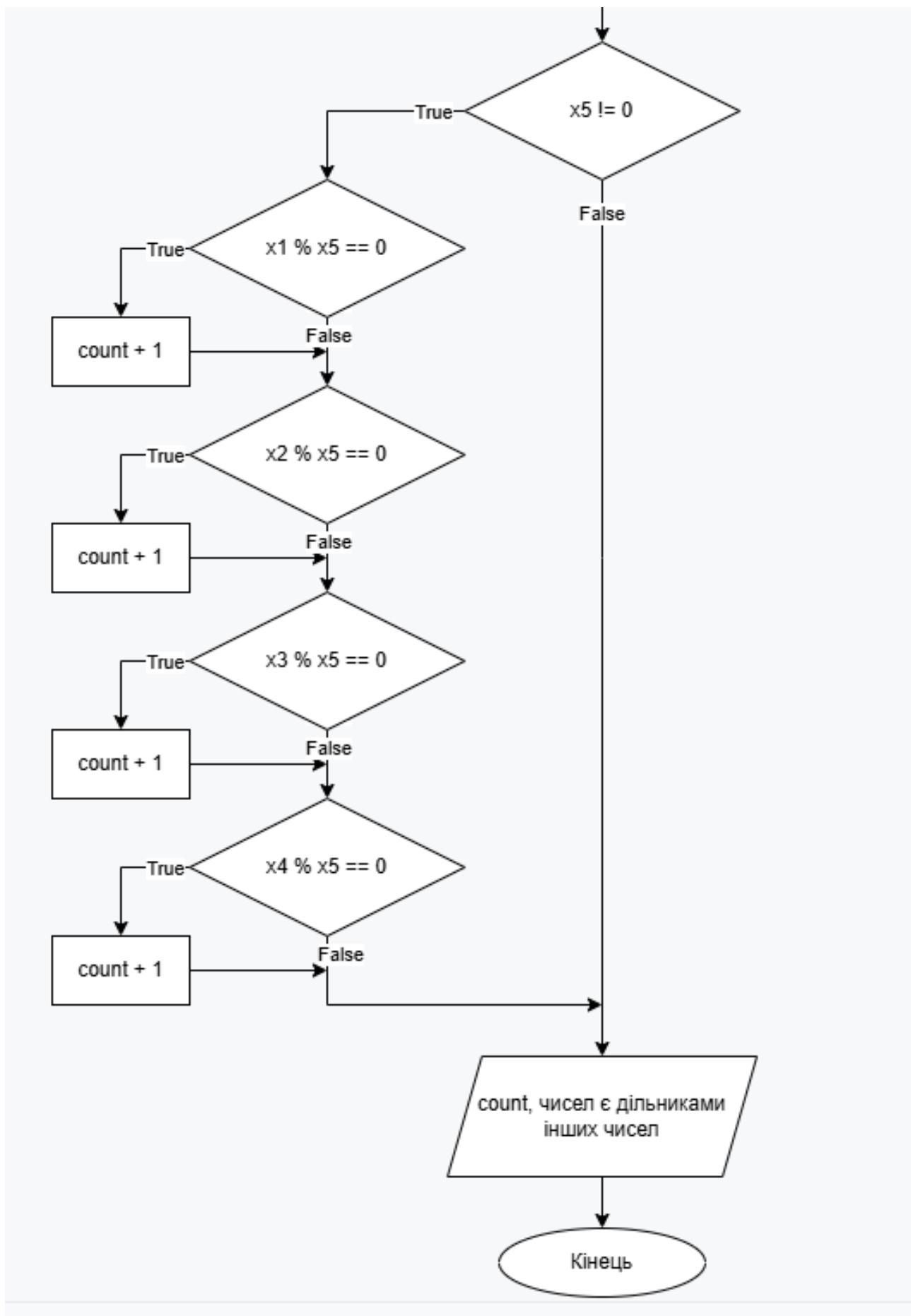


Рисунок 6 - 2.2 ч5

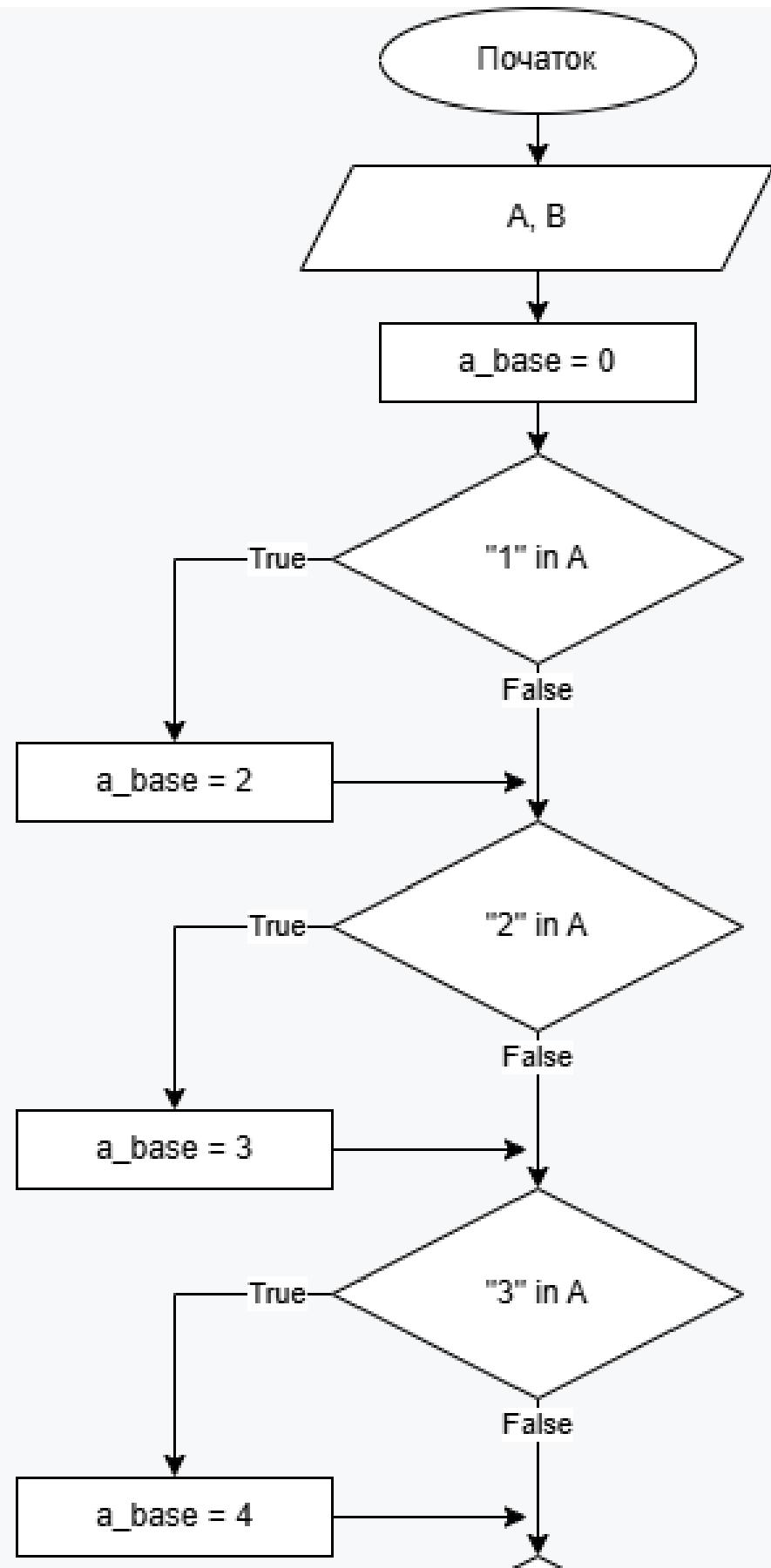


Рисунок 7 - 2.3 ч1

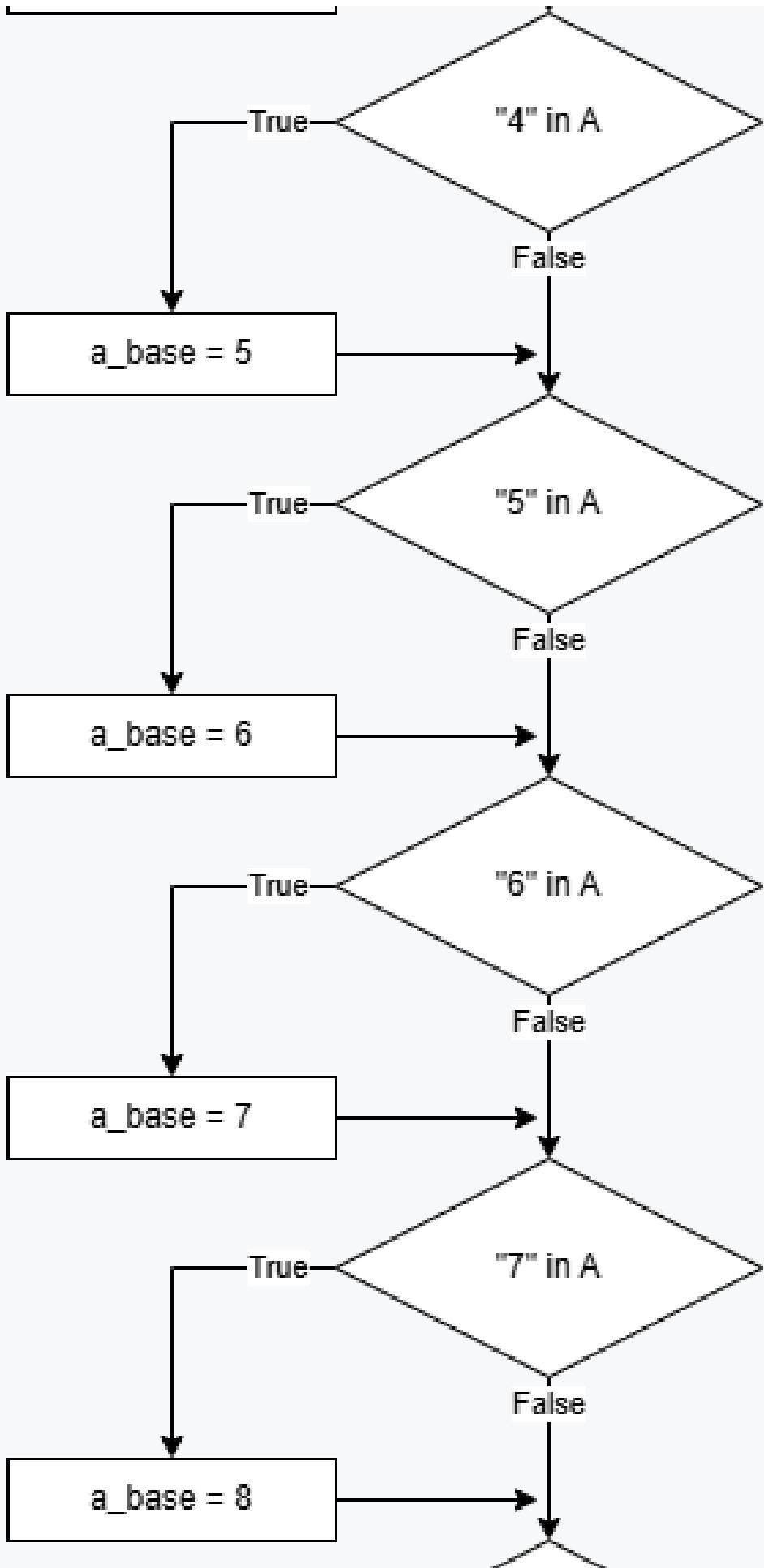
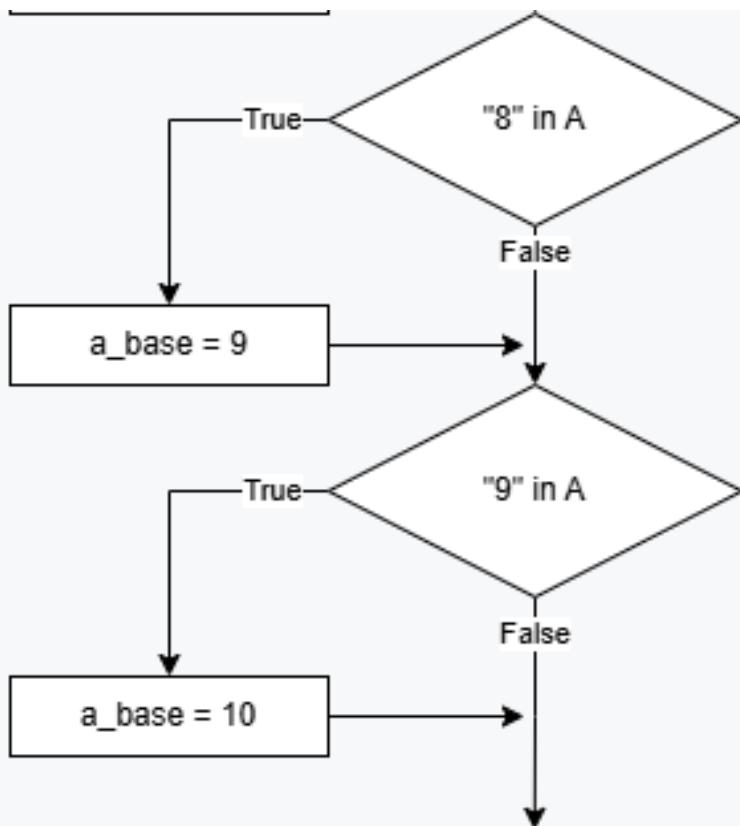


Рисунок 8 - 2.3 ч2



Мінімально можлива числення для числа A: a\_base,  
Число A в десятковій системі: int(A, a\_base)

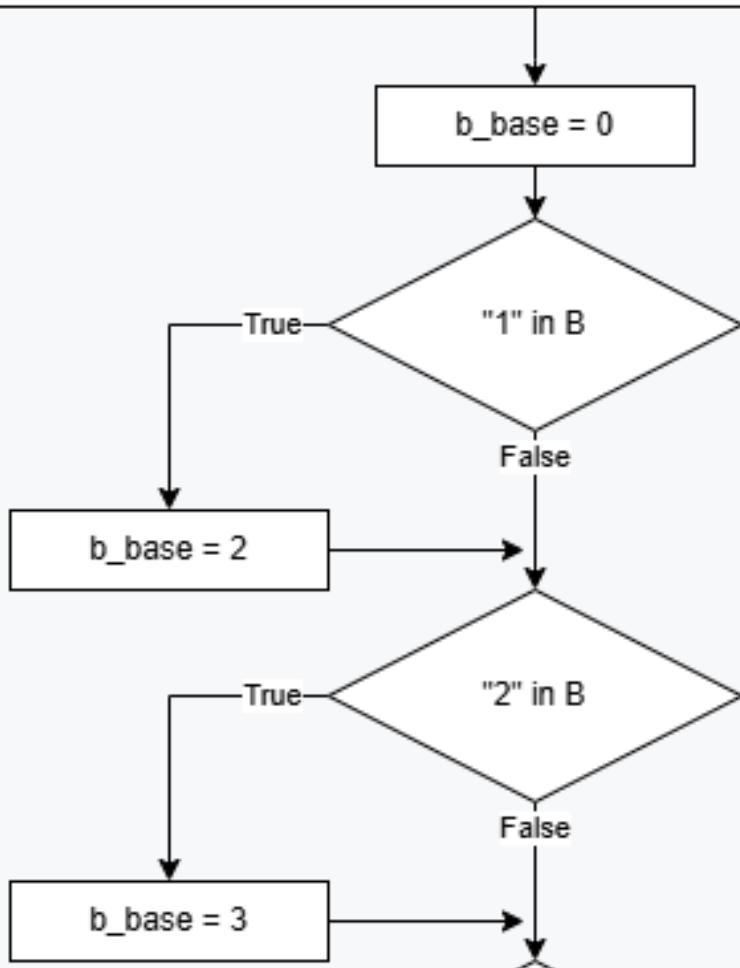


Рисунок 9 - 2.3 ч3

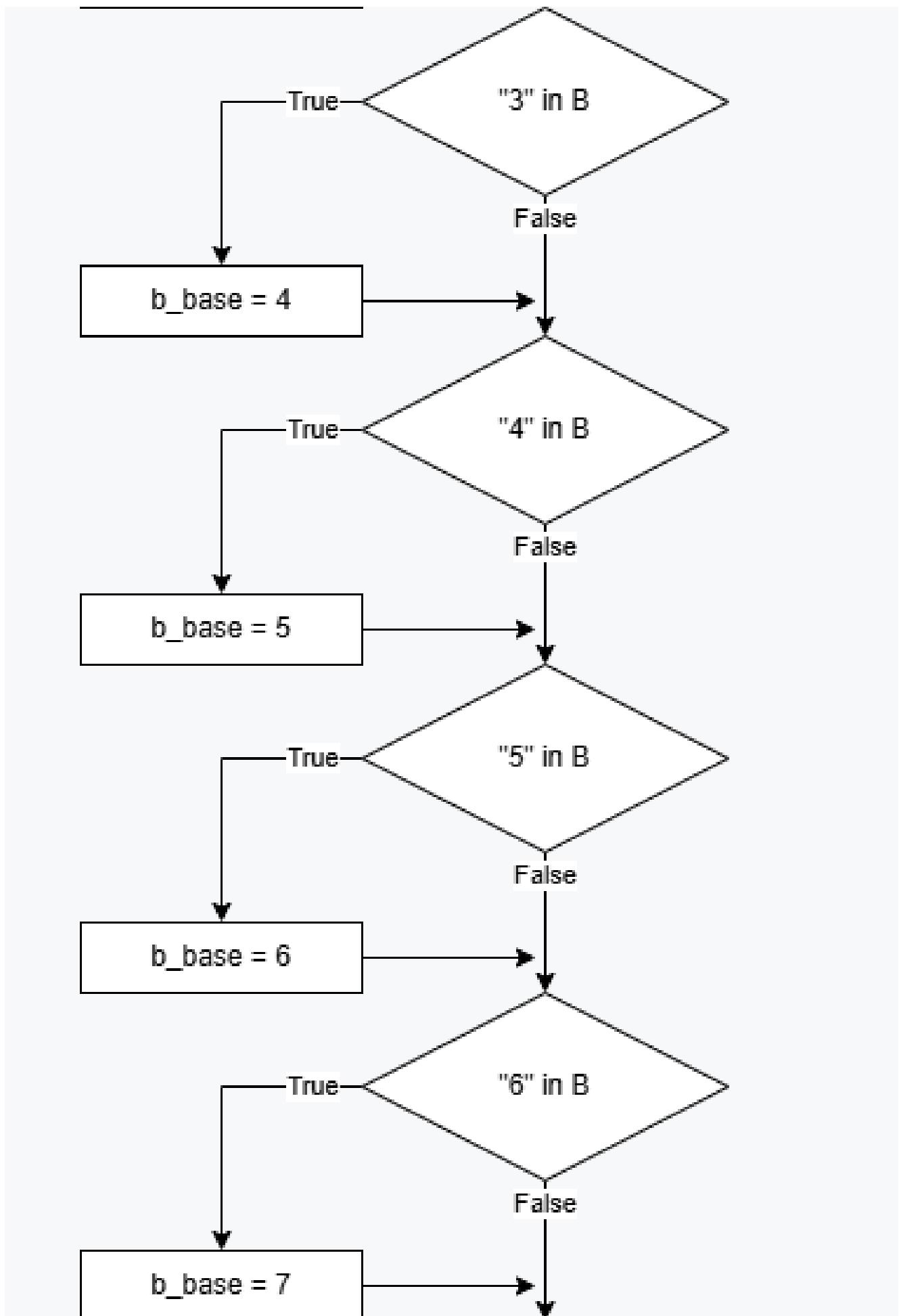


Рисунок 10 - 2.3 ч4

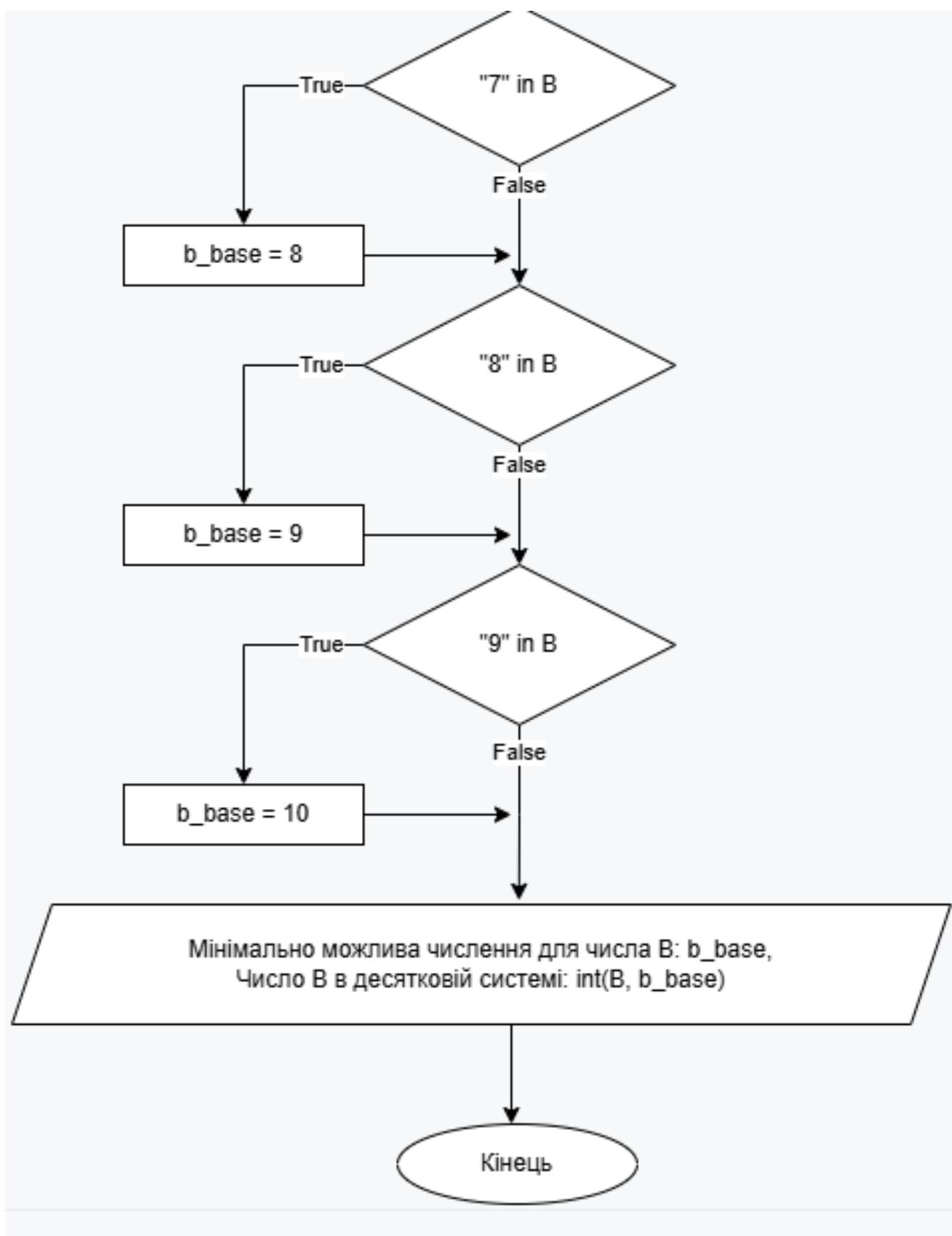


Рисунок 11 - 2.3 ч5

Висновок: під час виконання лабораторної роботи 7 було підвищено рівень використання всіх типів ф-цій в мові програмування пайтон. Реалізовано деякі ф-ції для вирішення різних задач.