

Міністерство освіти і науки України  
Національний технічний університет України «Київський політехнічний  
інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 3 з дисципліни  
«Алгоритми та структури даних-1.  
Основи алгоритмізації»

«Дослідження ітераційних циклічних алгоритмів»

Варіант 2

Виконав студент                    ПІ-15, Богун Даниїл Олександрович  
(шифр, прізвище, ім'я, по батькові)

Перевірив \_\_\_\_\_  
( прізвище, ім'я, по батькові)

Київ 2021

## Лабораторна робота 3

### Дослідження ітераційних циклічних алгоритмів

**Мета** – дослідити подання операторів повторення дій та набути практичних навичок їх використання під час складання циклічних програмних специфікацій.

#### Варіант 2

**Задача:** З точністю  $\varepsilon = 10^{-6}$  обчислити значення функції  $\ln a$  :

$$\ln a = (a - 1) - \frac{(a - 1)^2}{2} + \frac{(a - 1)^3}{3} - \dots, \quad \text{для } 0 \leq a \leq 2.$$

Порівняти одержане за допомогою ряду значення зі значенням, отриманим стандартною функцією.

#### **Постановка задачі:**

Нам дані значення аргумента  $a$  і значення  $\varepsilon$ .  $\ln a$  – це сума елементів послідовності, яка прямує до якогось числа. Виришувати задачу будемо використовуючи ряд Тейлора. Процес ітерації треба продовжувати, поки модуль різниці двох сусідніх елементів послідовності не буде менше за  $\varepsilon$ . Потім треба результат порівняти зі значенням, отриманим стандартною функцією.

## Побудова математичної моделі:

Змінна	Тип	Ім'я	Призначення
Точність	Дійсний	$\varepsilon$	Початкове дане
Аргумент	Дійсний	$a$	Початкове дане
Член послідовності	Дійсний	$x$	Проміжне дане
Наступний член послідовності	Дійсний	$x_n$	Проміжне дане
Лічильник ітерації	Цілий	$n$	Проміжне дане
Функція $\ln a$ за формулою	Дійсний	sum	Результат
Чисельник у формулі	Дійсний	$p$	Проміжне дане
Стандартна функція $\ln a$	Дійсний	$Y$	Проміжне дане
Порівняння функцій	Дійсний	res	Результат

$Y$  = стандартна функція  $\ln a$ ;

$Res = sum / Y$

Змінній  $n$  присвоюємо значення 1;

Змінній  $p$  присвоюємо значення -1;

Змінній  $x_n$  присвоюємо значення 1;

Змінній  $x$  присвоюємо значення 0;

Змінній sum присвоюємо значення 0;

## Розв'язання

Програмні специфікації запишемо у псевдокоді та графічній формі у вигляді блок-схеми.

Крок 1: Визначимо основні дії

Крок 2: Введення умови для  $a$ .

Крок 3: Деталізуємо знаходження функції  $\ln a$ .

Крок 4: Знаходимо  $Y$  і res.

*Крок 1*

**Початок**

Введення  $\varepsilon$

Введення  $a$

**Умова для  $a$**

Знаходження  $L_n$   $a$  з  
точністю  $\varepsilon$

Знаходження  $Y$

Знаходження  $res$

Виведення  $sum$

Виведення  $res$

**Кінець**

*Крок 2*

**Початок**

Введення  $\varepsilon$

Введення  $a$

**Якщо  $a \geq 2$  або  $a \leq 0$**

Виведення “Невірне  
значення  $a$ . ”

**Інакше**

**Знаходження  $L_n$   $a$  з  
точністю  $\varepsilon$**

Знаходження  $Y$

Знаходження  $res$

Виведення  $sum$

Виведення  $res$

**Кінець**

*Крок 3*

**Початок**

Введення  $\varepsilon$

Введення  $a$

**Якщо  $a \geq 2$  або  $a \leq 0$**

Виведення “Невірне  
значення  $a$ . ”

**Інакше**

**Повторити**

$x = x_n$

$p = p * (-(a-1))$

$x_n = p / n$

$n = n+1$

$sum = sum + x_n$

**Поки  $|x_n - x_{n-1}| >$   
 $\varepsilon$**

**Знаходження  $Y$**

Знаходження  $res$

Виведення  $sum$

Виведення  $res$

**Кінець**

*Крок 4*

**Початок**

Введення  $\varepsilon$

Введення  $a$

**Якщо**  $a \geq 2$  або  $a \leq 0$

Виведення “Невірне значення  $a$ . ”

**Інакше**

**Повторити**

$x = xn$

$p = p * (-(a-1))$

$xn = p / n$

$n = n+1$

$sum = sum + xn$

**Поки**  $|x_n - x_{n-1}| > \varepsilon$

$Y = \ln a$

**Знаходження**  $res$

Виведення  $sum$

Виведення  $res$

**Кінець**

*Крок 5*

**Початок**

Введення  $\varepsilon$

Введення  $a$

**Якщо**  $a \geq 2$  або  $a \leq 0$

Виведення “Невірне значення  $a$ . ”

**Інакше**

**Повторити**

$x = xn$

$p = p * (-(a-1))$

$xn = p / n$

$n = n+1$

$sum = sum + xn$

**Поки**  $|x_n - x_{n-1}| > \varepsilon$

$Y = \ln a$

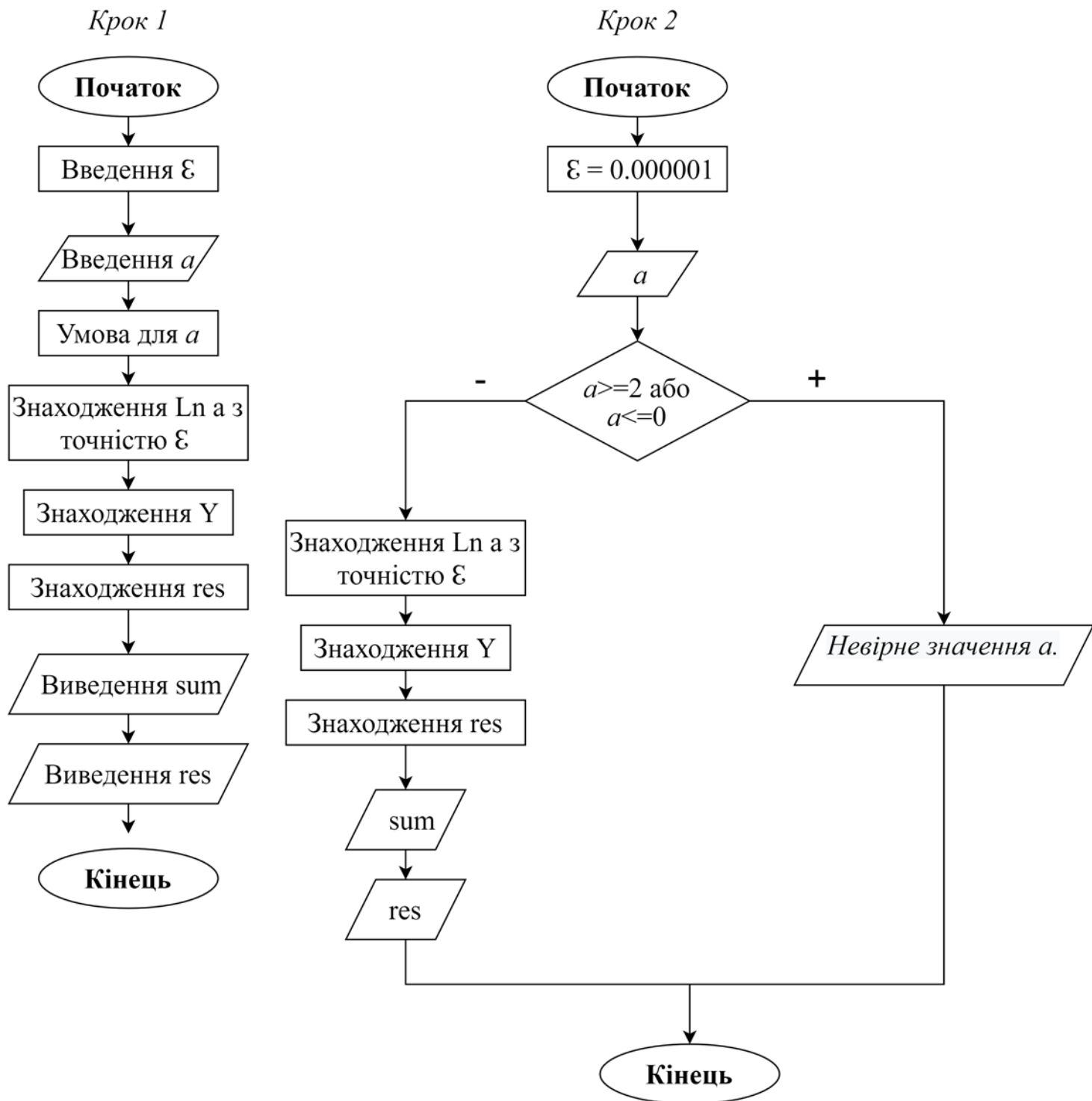
$Res = sum / Y$

Виведення  $sum$

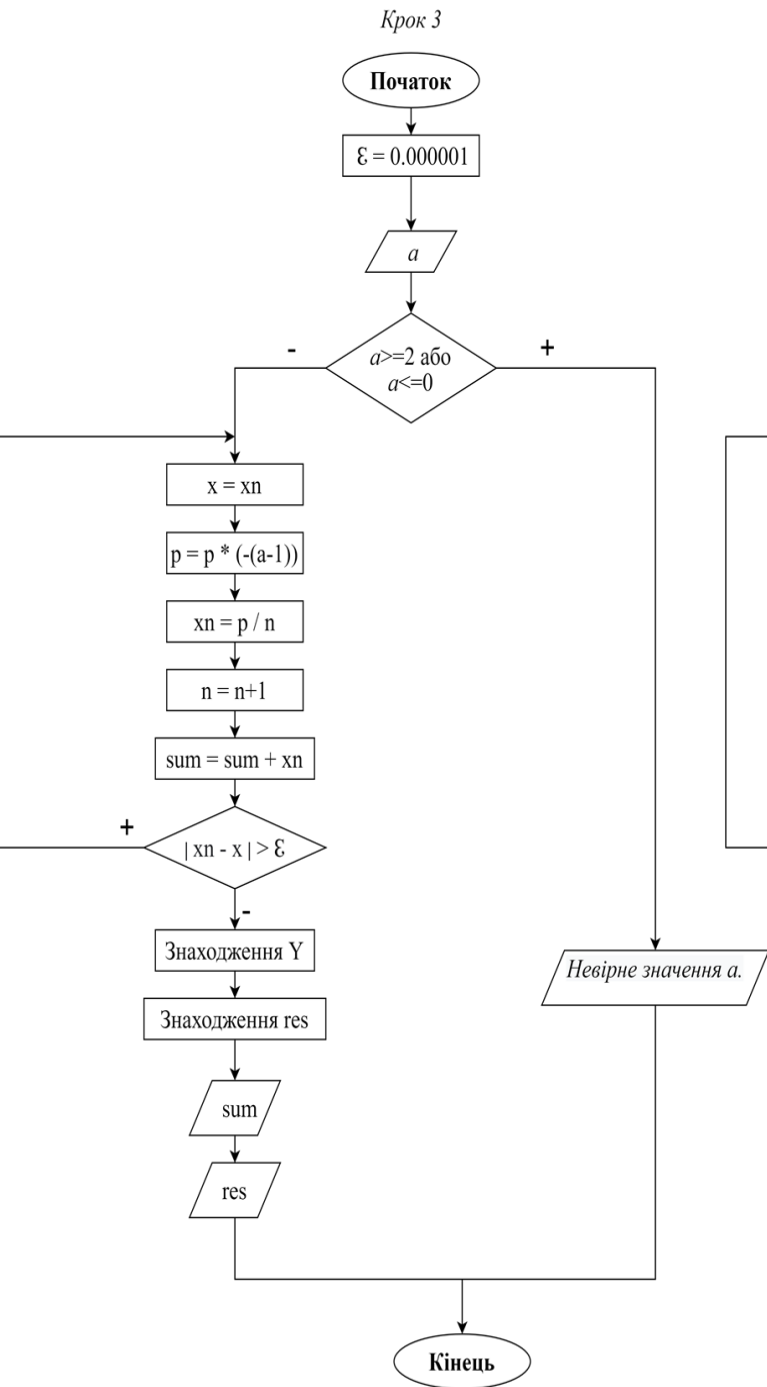
Виведення  $res$

**Кінець**

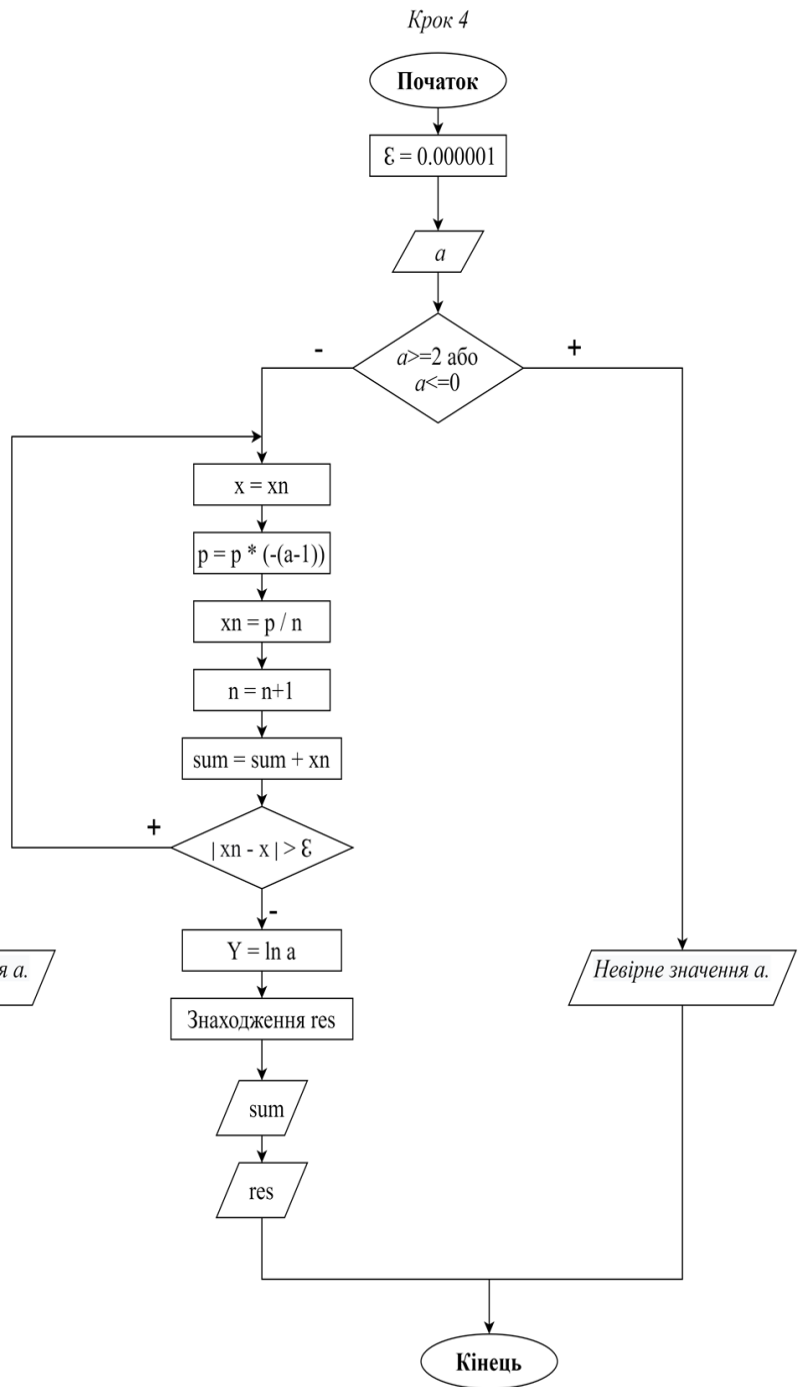
## Побудова блок–схеми алгоритму



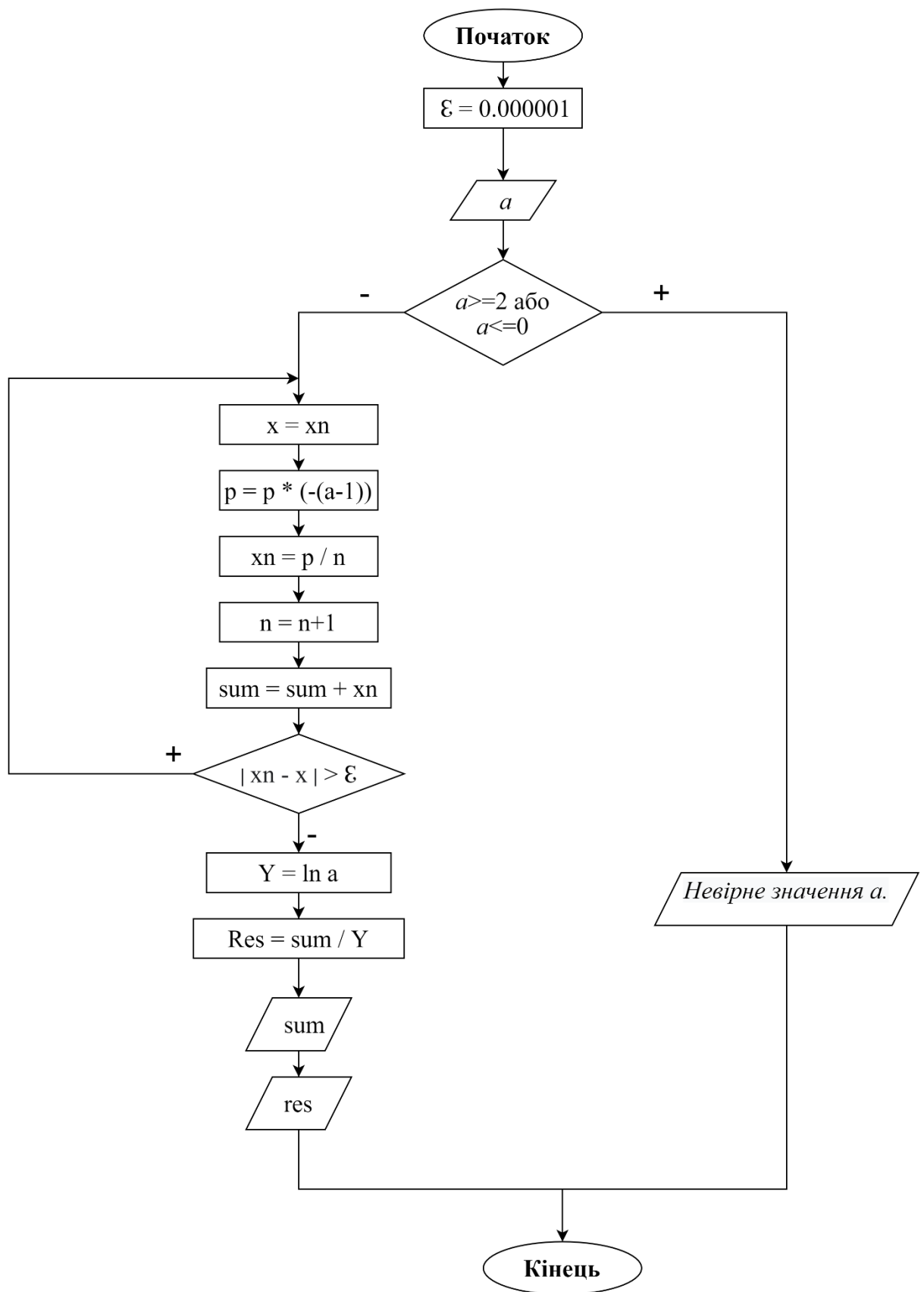
Крок 3



Крок 4



Крок 5





## Випробування алгоритму

<b>Початок</b>
a=1.6
n= 24
Sum=0.470004
res=1
<b>Кінець</b>

```
C:\Users\38096\Desktop\New Folder\ОП\Лаб 3\Lab3_OP_Bohun\Lab_3_Bohun\Debug\ConsoleApplication2.exe
0<a<=2  a=1.6
a=1.6  Sum=0.470004  res=1  n=24
Для продолжения нажмите любую клавишу . . .
```

<b>Початок</b>
a=0.67
n=13
Sum=-0.400478
res=1
<b>Кінець</b>

```
C:\Users\38096\Desktop\New Folder\ОП\Лаб 3\Lab3_OP_Bohun\Lab_3_Bohun\Debug\ConsoleApplication2.exe
0<a<=2  a=0.67
a=0.67  Sum=-0.400478  res=1  n=13
Для продолжения нажмите любую клавишу . . .
```

## Висновки

Протягом третьої лабораторної роботи ми дослідили подання операторів повторення дій та набули практичних навичок їх використання під час складання циклічних програмних специфікацій.