

*Додаток 1*

Міністерство освіти і науки України  
Національний технічний університет України «Київський політехнічний  
інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки  
Кафедра автоматизованих систем обробки інформації  
і управління

Звіт

з лабораторної роботи № 5  
з дисципліни «Алгоритми та структури даних-1. Основи  
алгоритмізації»  
«Дослідження лінійних алгоритмів »  
Варіант 8

Виконав

\_\_\_\_\_  
ІІ-15, Дацьо Іван Іванович

студент

(шифр, прізвище, ім'я, по батькові)

Перевірів

\_\_\_\_\_  
Вечерковська Анастасія Сергіївна

( прізвище, ім'я, по батькові)

Київ 2021

## Лабораторна робота 5

### Дослідження складних циклічних алгоритмів

**Мета** – дослідити особливості роботи складних циклів та набути практичних навичок їх використання під час складання програмних специфікацій.

**Індивідуальне завдання :**

#### Варіант 8

#### Задача:

Цифровий корінь натурального числа – це одноцифрове значення, яке отримується із цифр числа шляхом ітераційного процесу знаходження спочатку суми цифр даного числа, а потім, якщо потрібно, суми цифр значень, отриманих на попередній ітерації знаходження відповідних сум (якщо значення суми не є цифрою). Цей процес триває до тих пір, поки не буде отримано однорозрядне число. Наприклад, цифровим коренем числа 65536 є 7, так як  $6+5+5+3+6=25$  і  $2+5=7$ . Знайти цифрові корені всіх простих чисел з інтервалу  $[100, 200]$ .

### 1. Постановка задачі.

Знайти прості числа з проміжку  $[100, 200]$  і для цих чисел знайти цифровий корінь використовуючи складні циклічні алгоритми .

Результатом буде виведення простого і числа і його цифрового кореня .

### 2. Побудова математичної моделі

Змінна	Тип	Ім'я	Призначення
--------	-----	------	-------------

Лічильник зовнішнього циклу	Цілочисельний	i	Проміжні дані
Лічильник внутрішнього циклу	Цілочисельний	j	Проміжні дані
Показник	Цілочисельний	k	Проміжні дані
Просте число з проміжку	Цілочисельний	a	Проміжні дані
Цифровий корінь простого числа	Цілочисельний	total	Кінцеві дані

Для позначення остачі від ділення використовуємо знак : “ % ”

Для позначення цілочисельного ділення використовуємо знак “ // ”

Для знаходження цифрового кореня від двоцифрового числа використовуємо формулу :

$total = total // 10 + total \% 10$

Для позначення порівняння використовуємо знак “ == ”

- 1) За допомогою зовнішнього циклу перебираємо числа від 100 до 200 включно
- 2) Присвоюємо показнику k значення 0
- 3) Для чисел від 1 до числа i виконуємо внутрішній цикл
- 4) У цьому внутрішньому циклі перевіряємо дію остачі від ділення
- 5) У разі якщо остача від ділення рівна 0 збільшуємо значення k на 1
- 6) Якщо після закінчення внутрішнього циклу із j отримаємо k яке рівне 2 то це число i є простим і для нього шукаємо цифровий корінь
- 7) Змінну a присвоюємо значення i для подальшої зручності при виконанні дій
- 8) Змінній total присвоюємо значення 0
- 9) Виконуємо цикл while до поки  $a > 0$
- 10) total знаходимо через суму total і останньої цифри числа a
- 11) a присвоюємо значення цілої частини від ділення на 10
- 12) Перевіряємо чи значення total є більшим за 9
- 13) Якщо все ж якщо total більша за 9 то знаходимо цифровий корінь двоцифрового числа як суму цілої частини від ділення на 10 і остачі ділення цього числа на 10
- 14) Вивидимо значення i(просте число) , виводимо “=”, виводимо total( цифровий корінь)

### 3.Розв'язання

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

Крок 1. Визначимо основні дії .

Крок 2. Деталізуємо дію зовнішнього циклу .

Крок 3. Присвоєння значення показнику .

Крок 4. Деталізуємо дію вкладеного циклу .

Крок 5. Деталізуємо дію перевірки ділення числа з проміжку  $j$  .

Крок 6. Деталізуємо дію визначення  $k$

Крок 7. Деталізуємо дію перевірки  $k$  на рівність двом .

Крок 8. Деталізуємо дію присвоєння значень змінним  $a$  і  $total$  .

Крок 9. Деталізуємо умову `while` циклу .

Крок 10. Деталізуємо дію знаходження  $total$  .

Крок 11 . Деталізуємо дію знаходження  $a$  .

Крок 12. Деталізуємо дію перевірки значення  $total$ .

Крок 13. Деталізуємо дію знаходження  $total$ .

### 4. Псевдокод

Крок 1 .

#### Початок

Дія зовнішнього циклу

Присвоєння значення показнику

Дія вкладеного циклу

Перевірка ділення числа на число з проміжку  $j$

Визначення  $k$

Перевірка  $k$  на рівність двом

Присвоєння значень змінним  $a$  і  $total$

Умова `while` циклу

Знаходження  $total$

Знаходження a

Перевірка значення total

Знаходження значення total

**Кінець**

Крок 2.

**Початок**

**повторити**

**для i від 100 до 200 :**

Присвоєння значення показнику

Дія вкладеного циклу

Перевірка ділення числа на число з проміжку j

Визначення k

Перевірка k на рівність двом

Присвоєння значень змінним a і total

Умова while циклу

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**Кінець**

Крок 3. .

**Початок**

**повторити**

**для i від 100 до 200 :**

k =0

Дія вкладеного циклу

Перевірка ділення числа на число з проміжку j

Визначення k

Перевірка k на рівність двом

Присвоєння значень змінним a і total

Умова while циклу

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**все повторити**

**Кінець**

Крок 4.

**Початок**

**повторити**

**для i від 100 до 200 :**

k = 0

**для j від 1 до i :**

Перевірка ділення числа на число з проміжку j

Визначення k

Перевірка k на рівність двом

Присвоєння значень змінним a і total

Умова while циклу

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**все повторити**

**Кінець**

Крок 5.

**Початок**

**повторити**

**для i від 100 до 200 :**

k = 0

**повторити**

**для j від 1 до i:**

**якщо i % j == 0 :**

Визначення k

**все повторити**

Перевірка k на рівність двом

Присвоєння значень змінним a і total

Умова while циклу

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**все повторити**

**Кінець**

Крок 6.

**Початок**

**повторити**

**для i від 100 до 200 :**

k = 0

**повторити**

**для j від 1 до i :**

**якщо  $i \% j == 0$  :**

$k=k+1$

**все якщо**

**все повторити**

Перевірка k на рівність двом

Присвоєння значень змінним a і total

Умова while циклу

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**все повторити**

**Кінець**

Крок 7.

**Початок**

**повторити**

**для i від 100 до 200 :**

$k = 0$

**повторити**

**для j від 1 до i :**

**якщо  $i \% j == 0$  :**

$k=k+1$

**все якщо**

**все повторити**



**якщо**  $k == 2$  :

Присвоєння значень змінним  $a$  і  $total$

Умова while циклу

Знаходження  $total$

Знаходження  $a$

Перевірка значення  $total$

Знаходження значення  $total$

**все якщо**

**все повторити**

**Кінець**

Крок 8.

**Початок**

**повторити**

**для  $i$  від 100 до 200 :**

$k = 0$

**повторити**

**для  $j$  від 1 до  $i$  :**

**якщо  $i \% j == 0$  :**

$k = k + 1$

**все якщо**

**все повторити**

**якщо  $k == 2$  :**

$a = i$

$total = 0$

Умова while циклу

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**все якщо**

**все повторити**

**Кінець**

Крок 9.

**Початок**

**повторити**

**для i від 100 до 200 :**

k = 0

**повторити**

**для j від 1 до i :**

**якщо i % j == 0 :**

k = k + 1

**все якщо**

**все повторити**

**якщо k == 2 :**

a = i

total = 0

**while a > 0 :**

Знаходження total

Знаходження a

Перевірка значення total

Знаходження значення total

**все повторити**

**все якщо**

**все повторити**

**Кінець**

Крок 10.

**Початок**

**повторити**

**для i від 100 до 200 :**

k = 0

**повторити**

**для j від 1 до i :**

**якщо i % j == 0 :**

k = k + 1

**все якщо**

**все повторити**

**якщо k == 2 :**

a = i

total = 0

**while a > 0 :**

total = total + (a % 10 )

Знаходження a

Перевірка значення total

Знаходження значення total

**все повторити**

**все якщо**

**все повторити**

**Кінець**

Крок 11.

**Початок**

**повторити**

**для i від 100 до 200 :**

$k = 0$

**повторити**

**для j від 1 до i:**

**якщо  $i \% j == 0$  :**

$k = k + 1$

**все якщо**

**все повторити**

**якщо  $k == 2$  :**

$a = i$

$total = 0$

**повторити**

**while  $a > 0$  :**

$total = total + (a \% 10)$

$a = a // 10$

Перевірка значення total

Знаходження значення total

**все повторити**

**все якщо**

**все повторити**

**Кінець**

Крок 12.

**Початок**

**повторити**

**для i від 100 до 200 :**

**k = 0**

**повторити**

**для j від 1 до i:**

**якщо i % j == 0 :**

**k = k + 1**

**все якщо**

**все повторити**

**якщо k == 2 :**

**a = i**

**total = 0**

**повторити**

**while a > 0 :**

**total = total + (a % 10 )**

**a = a // 10**

**якщо total > 9 :**

**Знаходження значення total**

**все якщо**

**все повторити**

**все якщо**

**все повторити**

Крок 13.

**Початок**

**повторити**

**для i від 100 до 200 :**

**k = 0**

**повторити**

**для j від 1 до i :**

**якщо i % j == 0 :**

**k = k + 1**

**все якщо**

**все повторити**

**якщо k == 2 :**

**a = i**

**total = 0**

**while a > 0 :**

**total = total + (a % 10 )**

**a = a // 10**

**якщо total > 9 :**

**total = total // 10 + total % 10**

**все якщо**

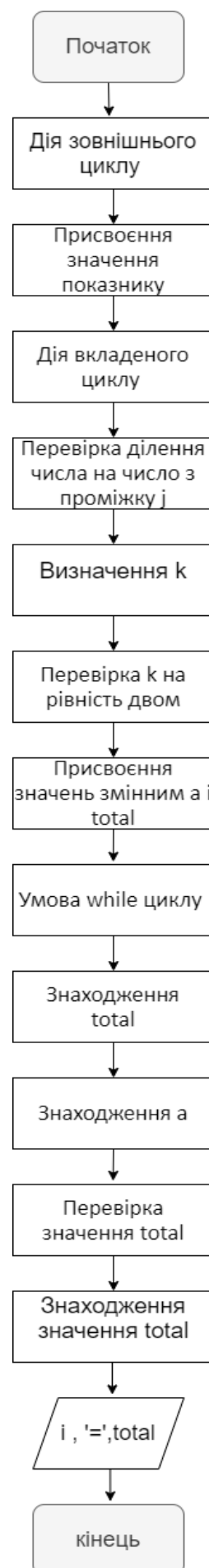
**все повторити**

**все якщо**

**все повторити**

## **5. Блок-схема алгоритму**

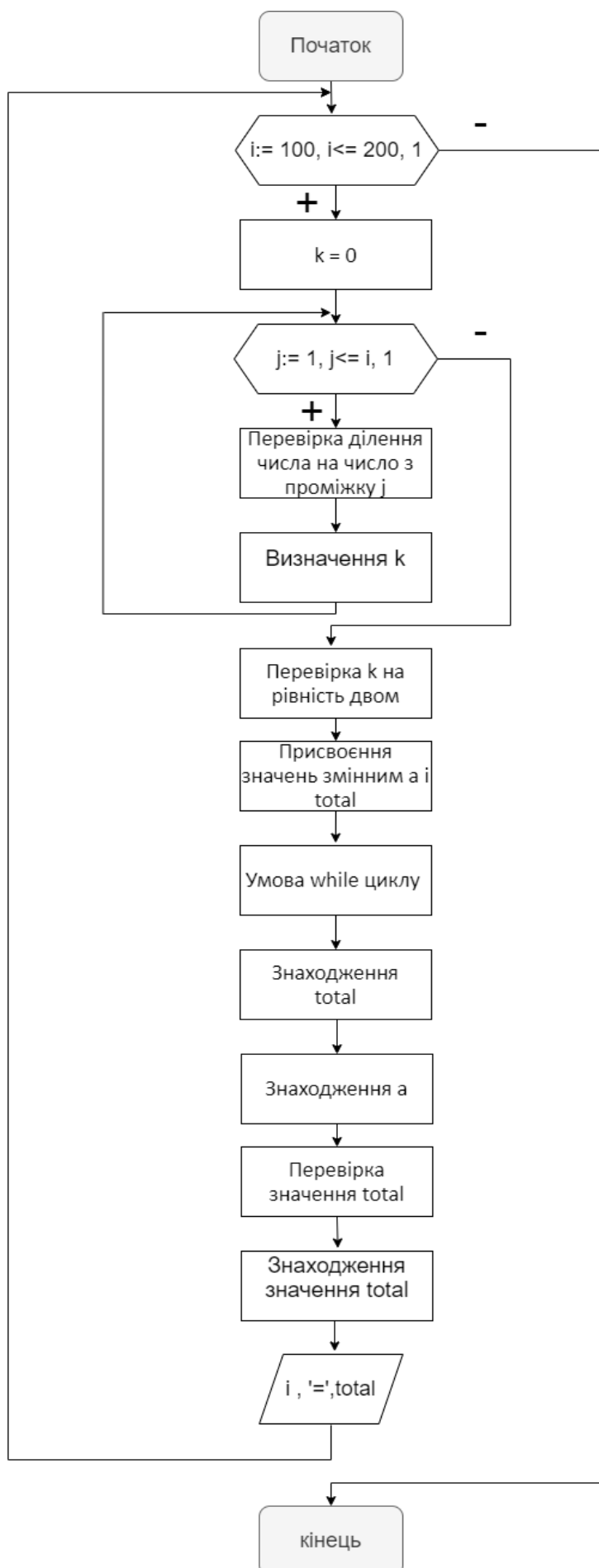
**Крок 1 .**

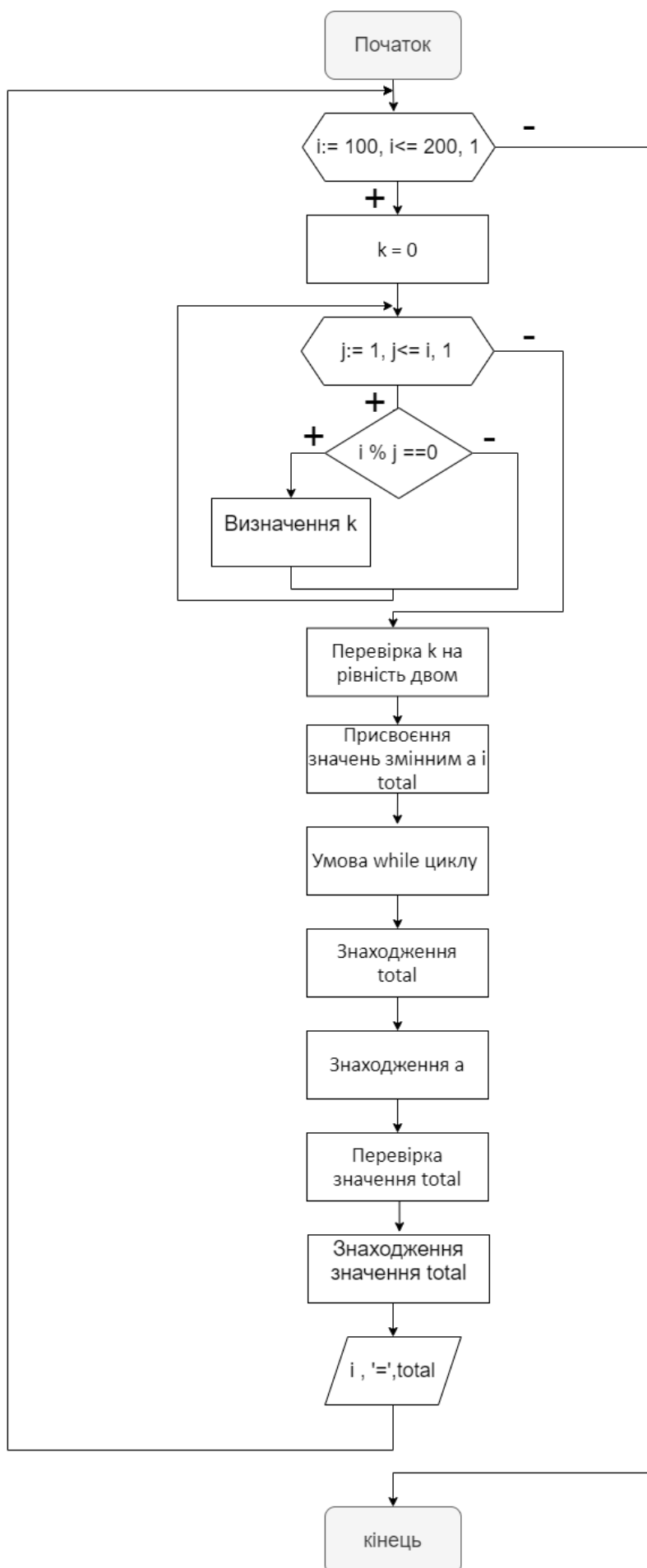


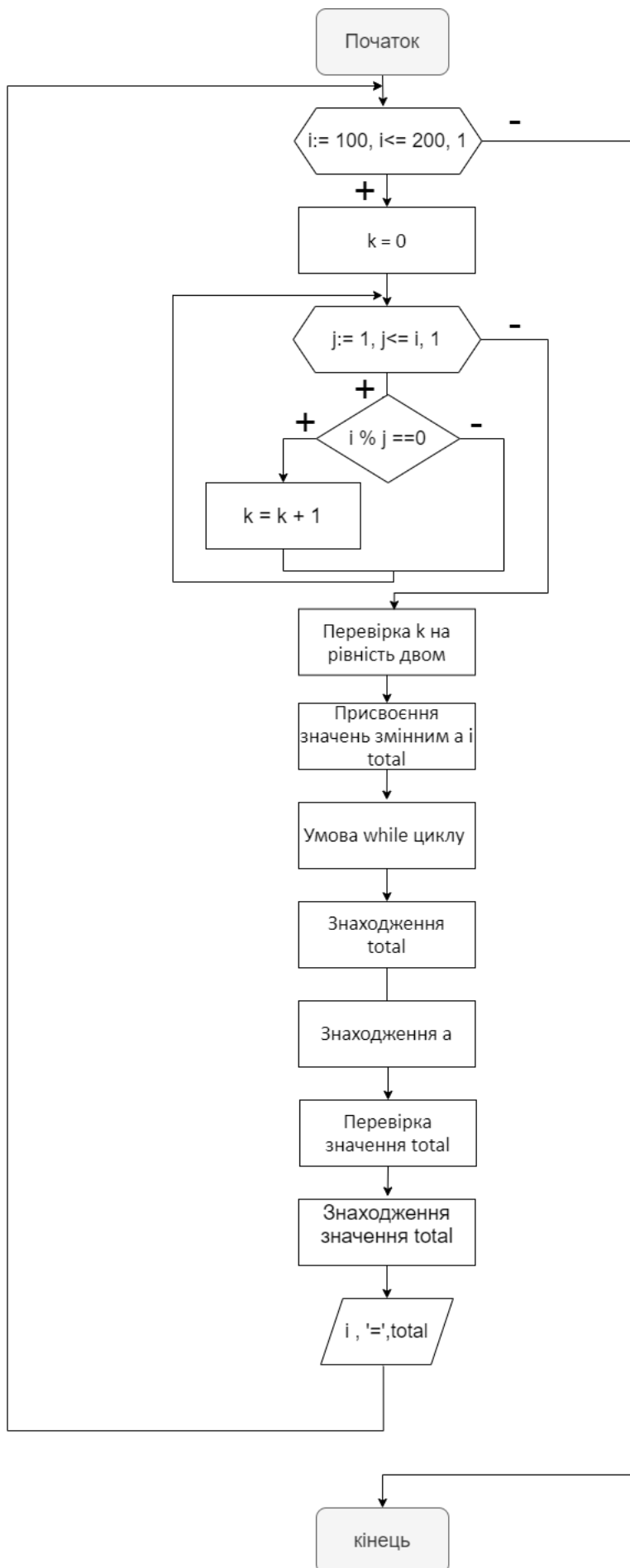


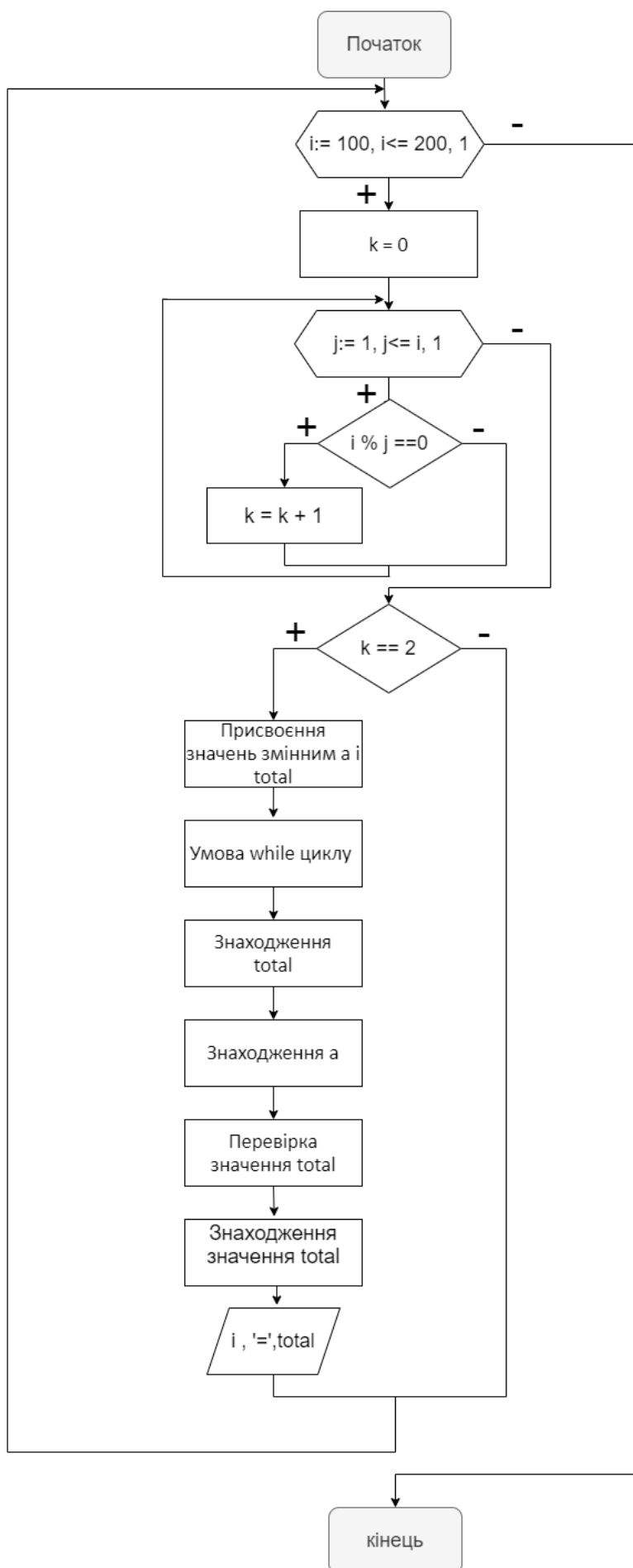


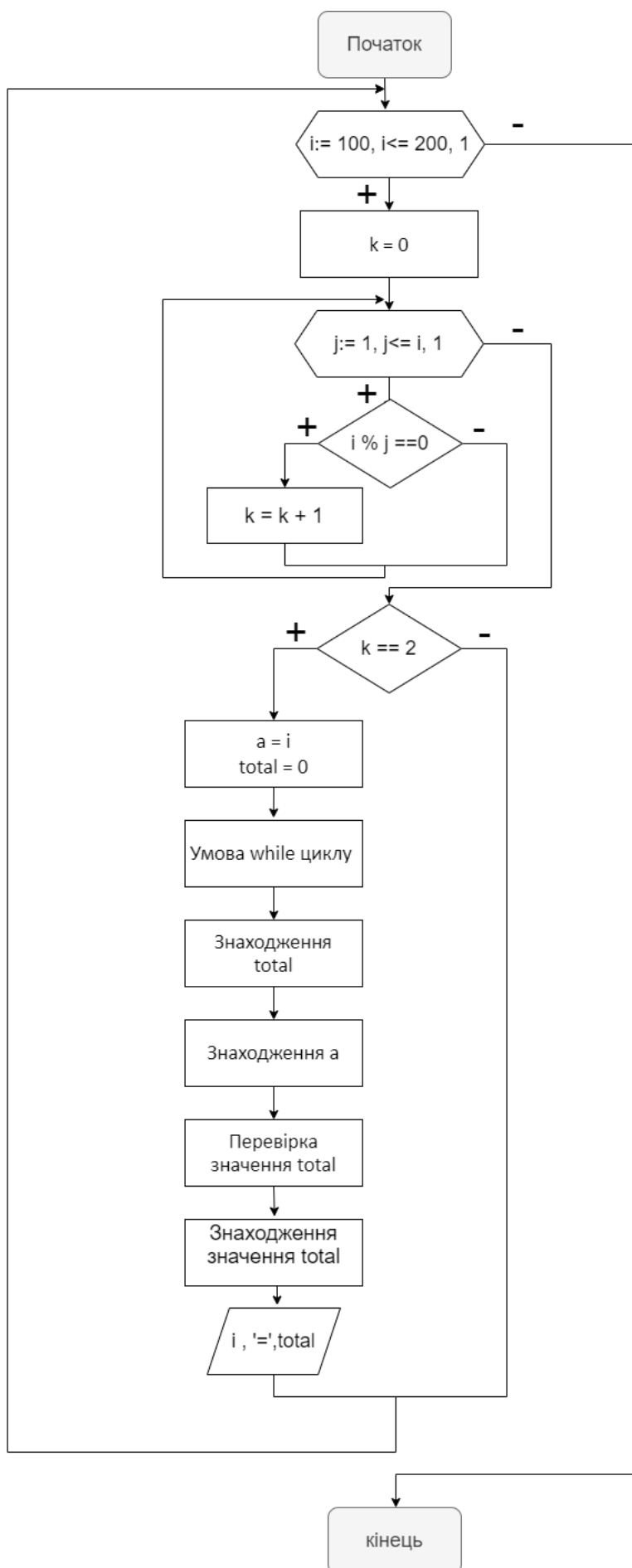


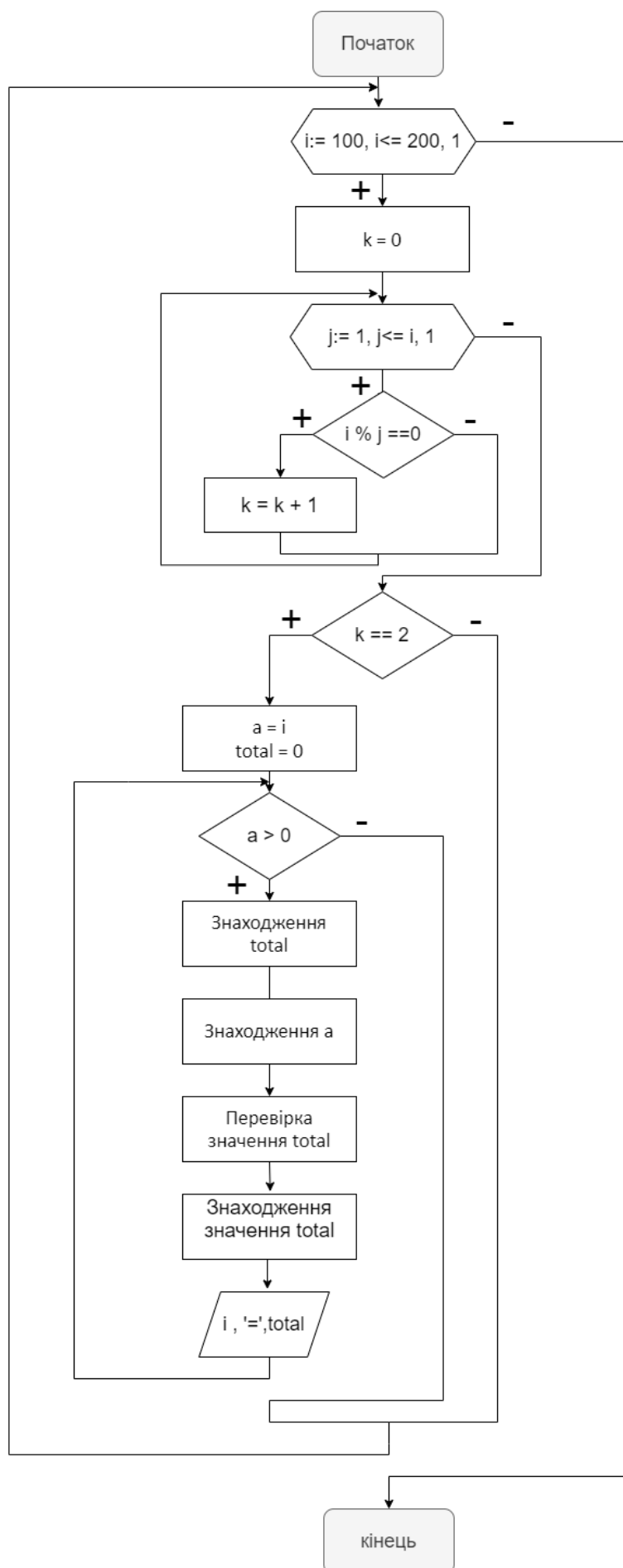


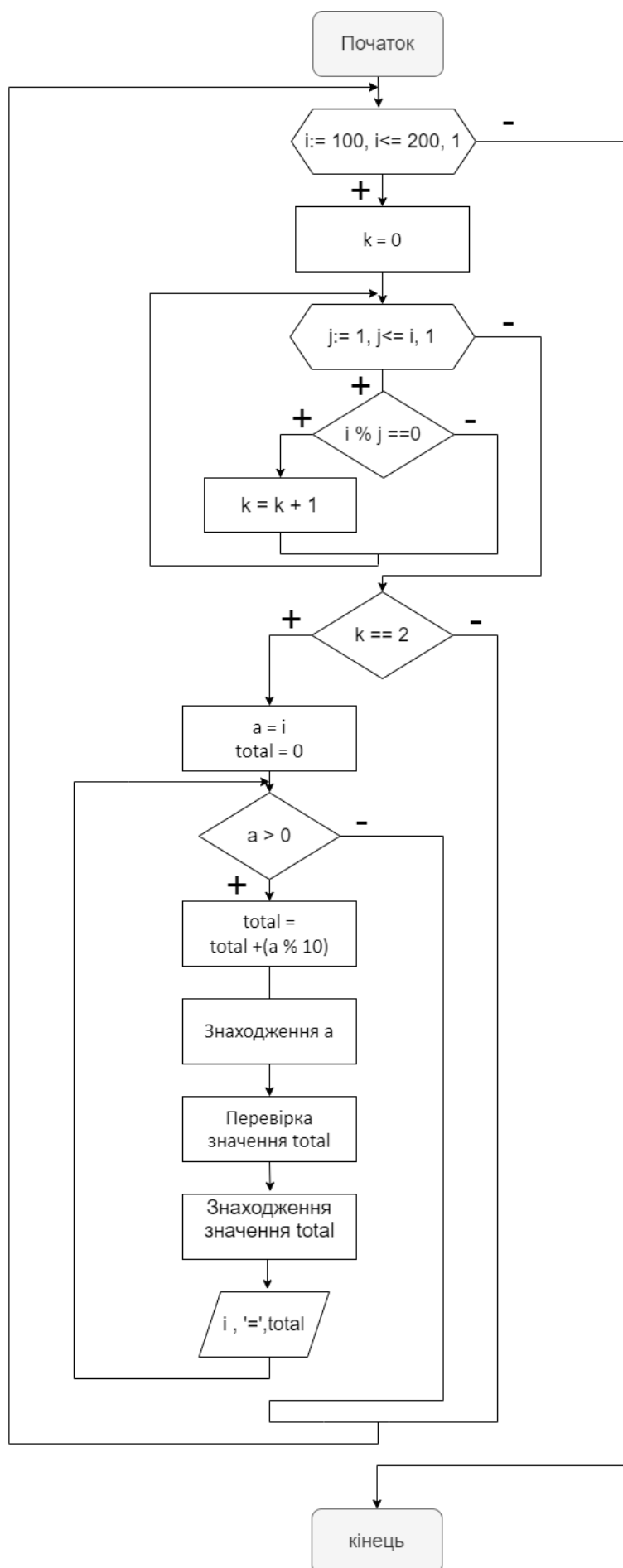




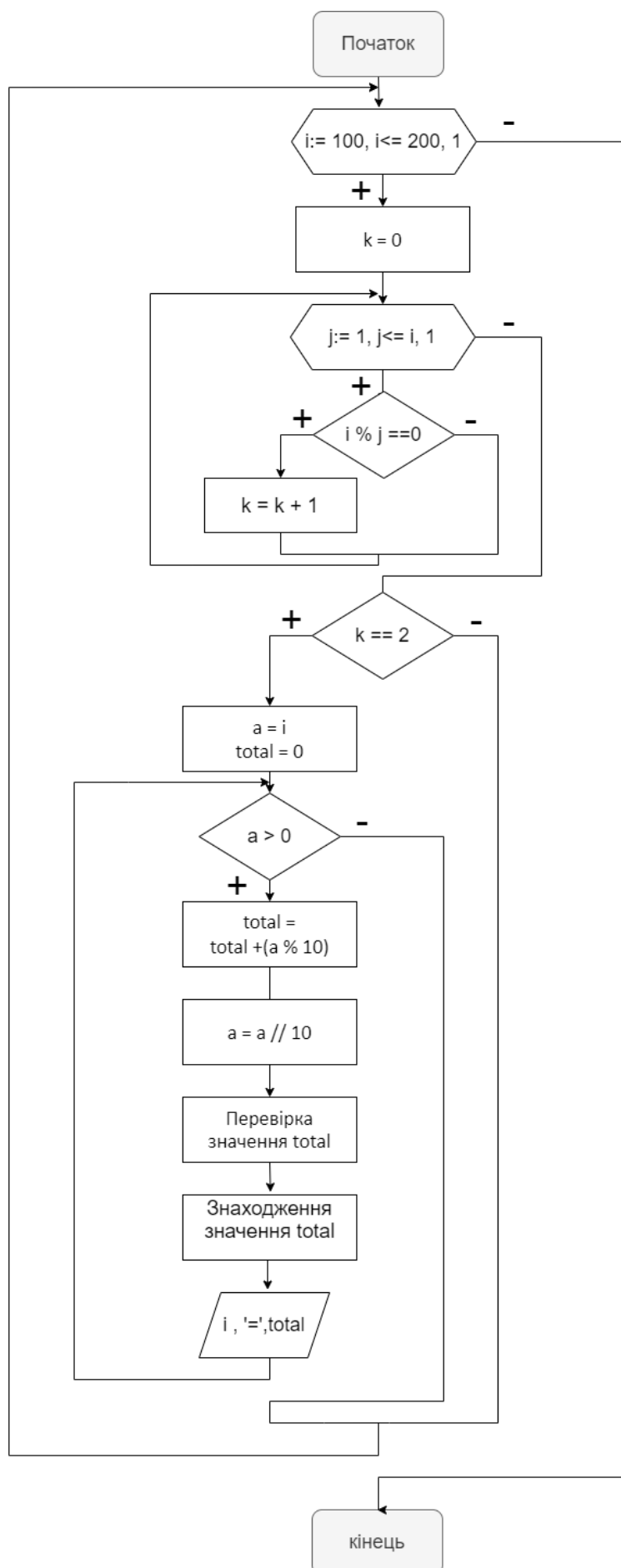


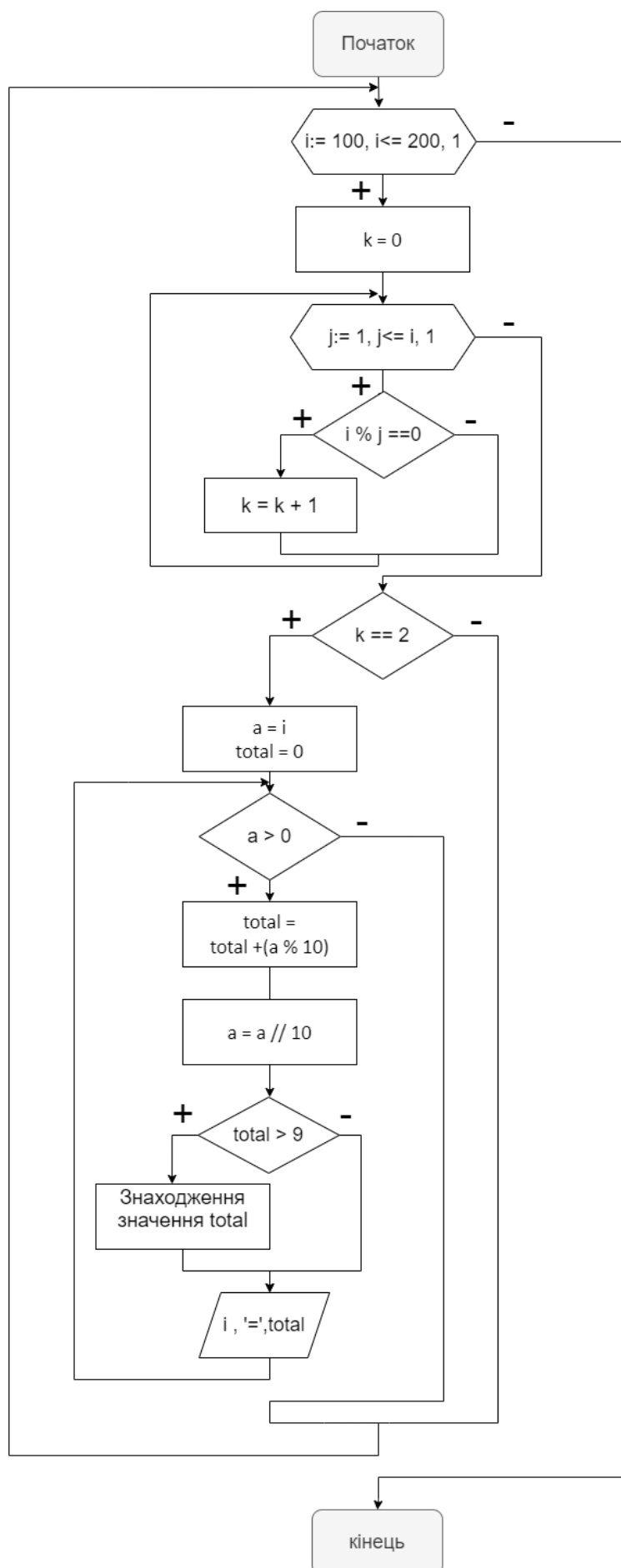


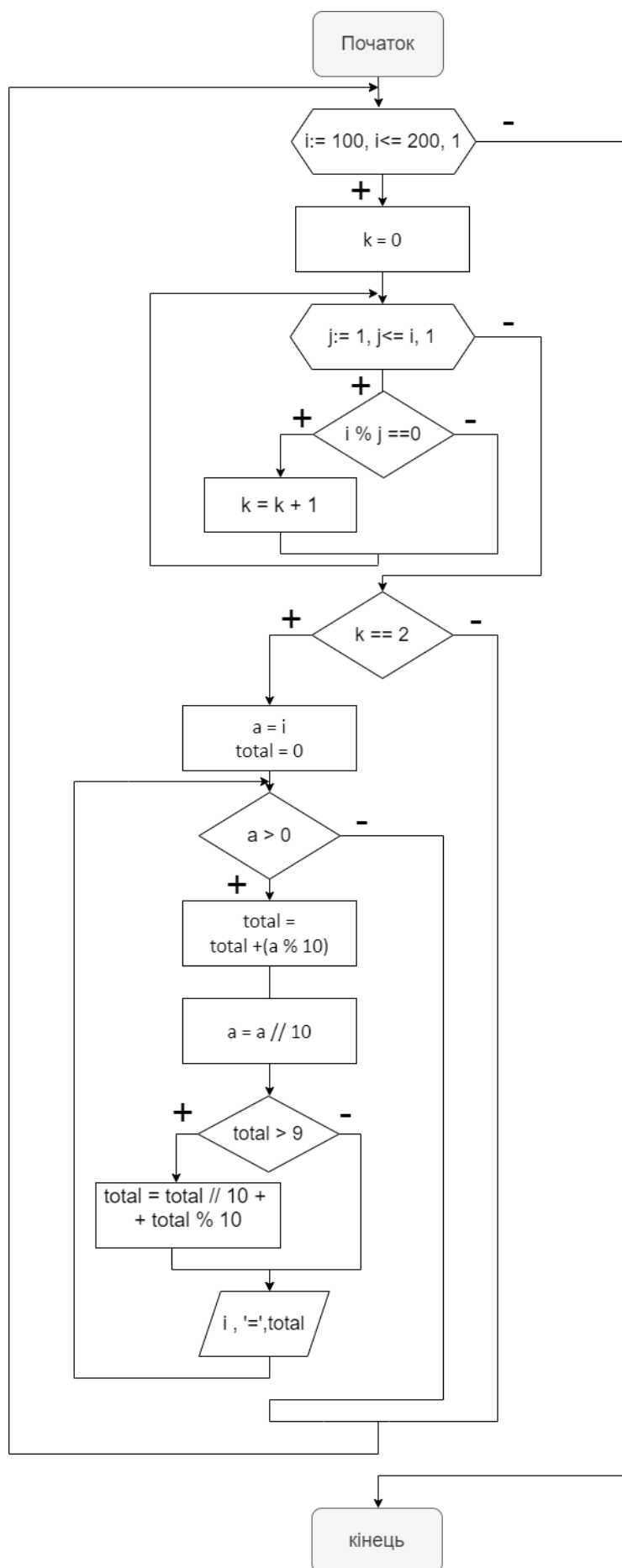












## 6 . Випробування алгоритму

Блок	Дія
	Початок
1	i = 100
2	k = 0
3	j = 1
4	k = 1
5	j = 2
6	k = 2
7	j = 3
8	k = 2
9	j = 4
10	k = 3
11	j = 5
12	k = 4
13	j = 6
14	k = 4
15	j = 7
16	k = 4
.....	
202	j = 100
203	k = 9
204	false
203	i = 2
204	k = 0
205	j = 1
206	k = 1
207	j = 2
208	k = 1
209	j = 3
210	k = 1
.....	

304	j = 101
305	k = 2
306	true
307	total = 2
308	Виведення: 101 =2
.....	
7515	i = 137
7516	j = 1
7517	k = 1
7518	j = 2
7519	k = 1
7520	j = 3
7521	k = 1
.....	
7791	j = 137
7792	k = 2
7793	true
7794	total = 11
7795	true
7796	total = 2
7797	Виведення: 137 =2
.....	

## 7.Висновок

Було досліджено особливості роботи складних циклів та набуто практичних навичок їх використання під час складання програмних специфікацій. В результаті виконання лабораторної роботи ми отримали алгоритм який знаходить цифровий корінь простих чисел з проміжку [ 100, 200 ] розділивши задачу на 13 кроків .