

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 № 5 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження складних циклічних алгоритмів»

Варіант 7

Виконав студент ІП-13 Гогіберідзе Торніке Лашаевич

(шифр, прізвище, ім'я, по батькові)

Перевірила

( 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.

#### Варіант 7

7. Натуральне число називається паліндромом, якщо його запис читається однаково з початку та з кінця (наприклад, 4884, 575, 9). Знайти всі числа-паліндроми, що не перевищують 1000.

#### *Постановка задачі*

Використовуємо складний циклічний алгоритм: зовнішній арифметичний цикл, внутрішній цикл з передумою і внутрішній арифметичний цикл (між собою останні два цикла – послідовні.) Використовуючи зовнішній арифметичний цикл, перебираємо всі натуральні числа, що не перевищують 1000. Використовуючи внутрішній цикл з передумою, знаходимо кількість цифр в числі, яке ми розглядаємо. Використовуючи внутрішній арифметичний цикл, «перевертаємо» число та записуємо його в змінній `reNum`. Вкінці кожного повторення зовнішнього циклу порівнюємо початкове число з його «перевернутим» числом (використовуємо умовну форму оператора вибору). Якщо обидва числа одинакові – виводимо наше початкове число.

Результатом розв'язку є знаходження та виведення всіх чисел-паліндромів, що не перевищують 1000.

#### *Побудова математичної моделі*

Таблиця імен змінних

<b>Змінна</b>	<b>Тип</b>	<b>Ім'я</b>	<b>Призначення</b>
Верхня границя паліндромів, які потрібно знайти	Ціле	limit	Початкові дані
Лічильник в зовнішньому арифметичному циклі	Ціле	i	Проміжні дані
Лічильник в внутрішньому арифметичному циклі	Ціле	j	Проміжні дані
Число, яке ми наразі розглядаємо	Натуральне	curNum	Проміжні дані/ Результат
«Перевернуте» число	Ціле	reNum	Проміжні дані
Кількість цифр в числі	Ціле	quantity	Проміжні дані

Перед початком циклу ми встановлюємо значення ліміту змінної на 1000

(верхня межа паліндрому, яку потрібно знайти). Усередині зовнішнього циклу встановіть значення змінної curNum = i (число, яке ми зараз перевіряємо).

Далі ініціалізуємо змінні reNum = 0 (змінна «взаємна») і quantity

= 0 (кількість цифр у числі). Використовуйте внутрішній цикл з передумовою, щоб знайти кількість цифр у числі: розділіть наше число на 10 цілих чисел і додайте 1 до змінної числа. Для цілого поділу ми використовуємо div (curNum = curNum div 10). В кінці ітераційного циклу ми повертаємо значення curNum = i і використовуємо лічильник j, щоб увійти у внутрішній арифметичний цикл, який побудує число, «протилежне» нашому. Від j = число-1 до 0 програма помножить кожну цифру нашого числа на 10 в степені j і додасть її до змінної reNum; отже, кожне число буде записане з кінця. Додайте перше число в кінці наступним чином: reNum = reNum + (curNum mod 10) \* (10 pow j)), де mod — це функція для знаходження залишку від ділення, а pow — функція для обчислення ступеня (у нашому випадку ступінь 10). Додайте ціле число і поділіть на curNum = curNum div 10 і повторіть до кінця внутрішнього арифметичного циклу. Потім ми порівнюємо reNum з curNum, і якщо вони однакові, виводимо curNum (знаходимо паліндром).

### *Розв'язання*

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

*Крок 1.* Визначимо основні дії.

*Крок 2.* Ініціалізація змінної limit

*Крок 3.* Визначення зовнішнього арифметичного циклу

*Крок 4.* Ініціалізація змінних curNum, reNum і quantity

*Крок 5.* Визначення внутрішнього ітераційного циклу

*Крок 6.* Задання значення змінної curNum

*Крок 7.* Визначення внутрішнього арифметичного циклу

*Крок 8.* Визначення оператору розгалуження та виведення паліндрому

*Псевдокод*

**Крок 1 початок**

Ініціалізація змінної limit

Визначення зовнішнього  
арифметичного циклу **кінець**

**Крок 2**

**початок**

limit : = 1000

Визначення зовнішнього  
арифметичного циклу

**кінець**

**Крок 3**

**початок**

limit : = 1000

**повторити**

для i від 1 до limit

Ініціалізація змінних curNum, reNum і quantity

Визначення внутрішнього ітераційного циклу

Задання значення змінної curNum

Визначення внутрішнього арифметичного циклу

Визначення оператору розгалуження та виведення паліндрому

**все повторити**

**кінець**

*Крок 4 початок limit := 1000*

**повторити** для i від 1  
до limit

curNum := i  
reNum := 0  
quantity := 0

Визначення внутрішнього ітераційного циклу

Задання значення змінної curNum

Визначення внутрішнього арифметичного циклу

Визначення оператору розгалуження та виведення паліндрому **все**  
**повторити**

**кінець**

*Крок 5 початок limit := 1000*

**повторити** для i від 1  
до limit

curNum := i  
reNum := 0  
quantity := 0

**повторити**

**поки** (curNum != 0)

quantity := quantity + 1

curNum := curNum div 10

**все повторити**

Задання значення змінної curNum

Визначення внутрішнього арифметичного циклу

Визначення оператору розгалуження та виведення паліндрому **все**  
**повторити**

**кінець**

*Krok 6 початок limit := 1000*

**повторити для i від 1**

**до limit curNum := i**

**reNum := 0**

**quantity := 0**

**повторити**

**поки (curNum != 0)**

**quantity := quantity + 1**

**curNum := curNum div 10**

**все повторити**

**curNum := i**

**Визначення внутрішнього арифметичного циклу**

Визначення оператору розгалуження та виведення паліндрому **все**

**повторити**

**кінець**

*Krok 7 початок limit := 1000*

**повторити для i від 1**

**до limit**

**curNum := i**

**reNum := 0**

**quantity := 0**

**повторити**

**поки (curNum != 0)**

**quantity := quantity + 1**

**curNum := curNum div 10**

**все повторити**

**curNum := i**

**повторити** для  $j$  від (quantity -1) до 0, з

кроком -1

    reNum = reNum + (curNum mod 10) \* (10 pow j)

    )                  curNum := curNum div 10                  **все повторити**

Визначення оператору розгалуження та виведення паліндрому **все**

**повторити**

**кінець**

*Krok 8 початок limit := 1000*

**повторити** для  $i$  від 1

до limit

    curNum :=

    i    reNum := 0

    quantity := 0

**повторити**

**поки** (curNum != 0)

    quantity := quantity + 1

    curNum := curNum div 10

**все повторити**

    curNum := i    **повторити** для  $j$  від (quantity -1) до 0, з

кроком -1

    reNum = reNum + (curNum mod 10) \* (10

    pow j )

    curNum := curNum div 10                  **все повторити**

**якщо** (i == reNum)

**то**

**виведення** i

**все якщо**

**все повторити**

**кінець**

## Блок-схема

Крок 1.

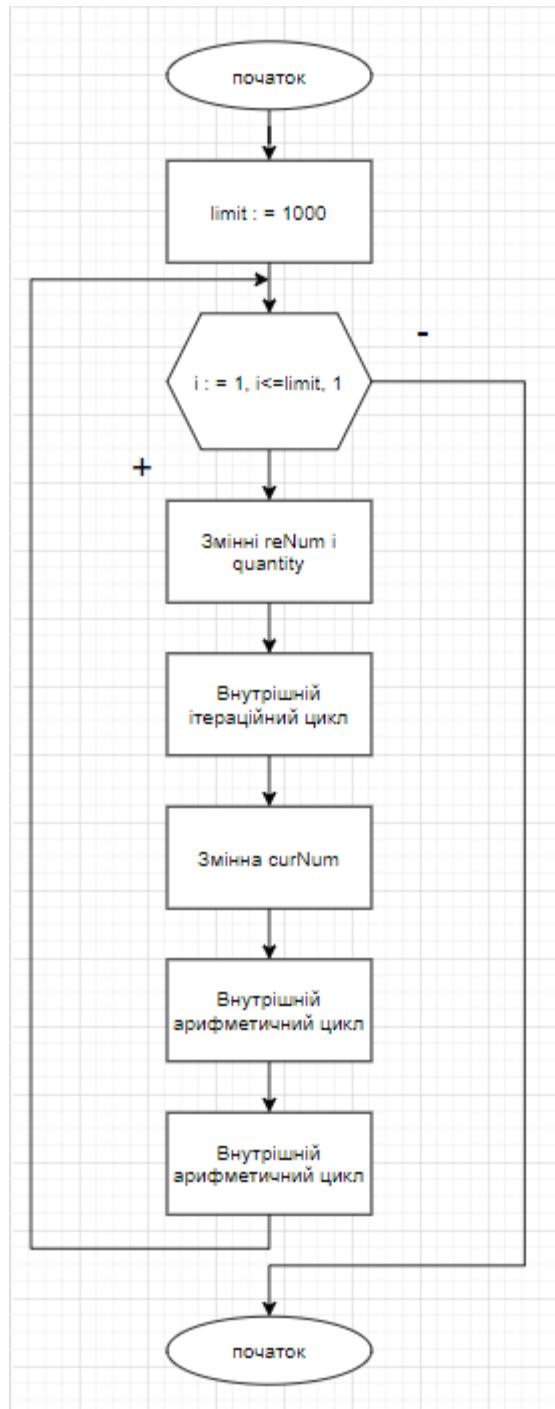


Крок 2.

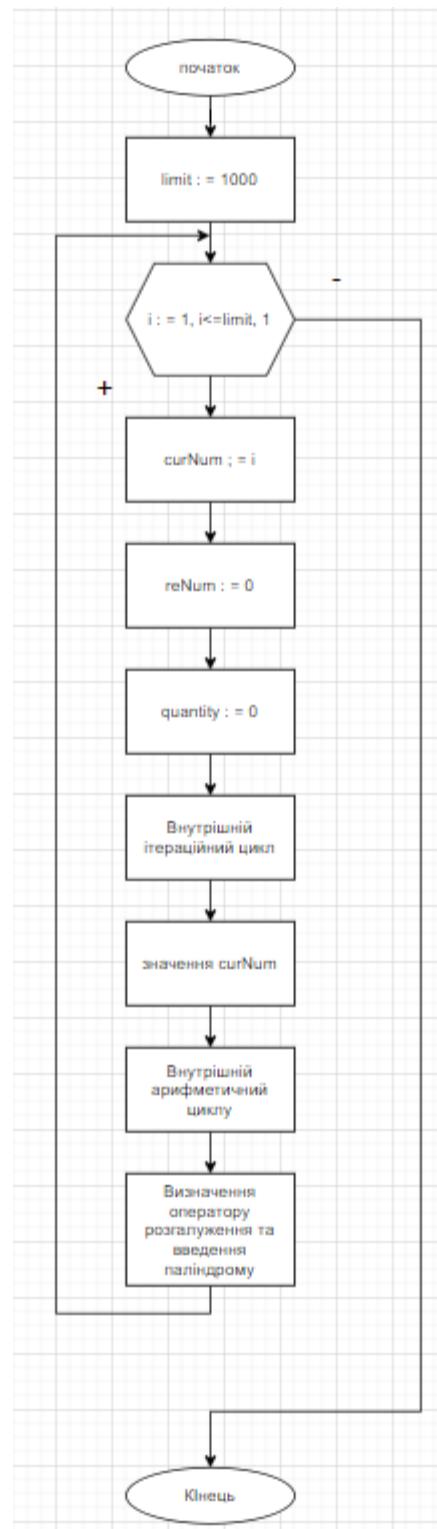


Крок 3.

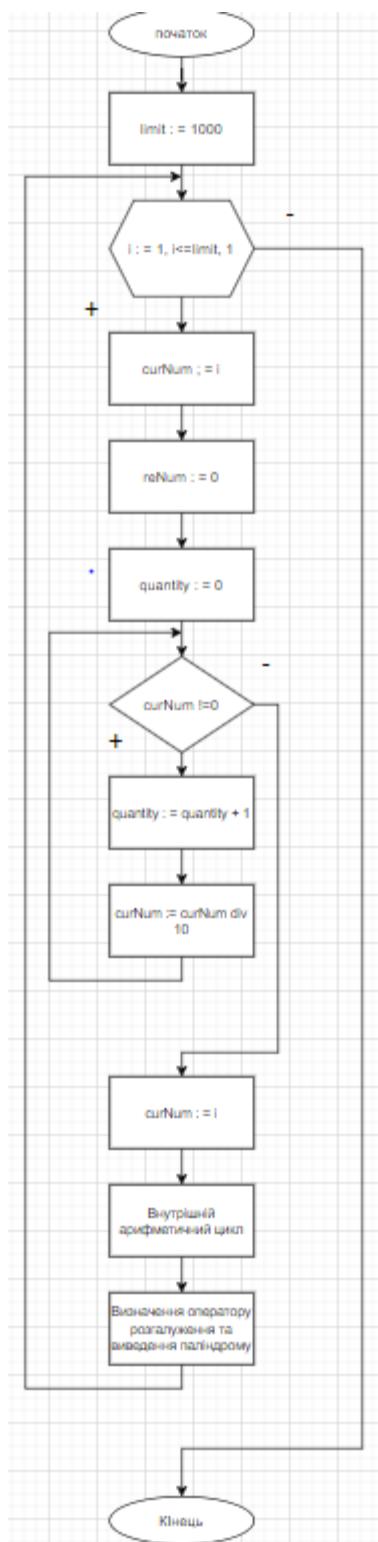
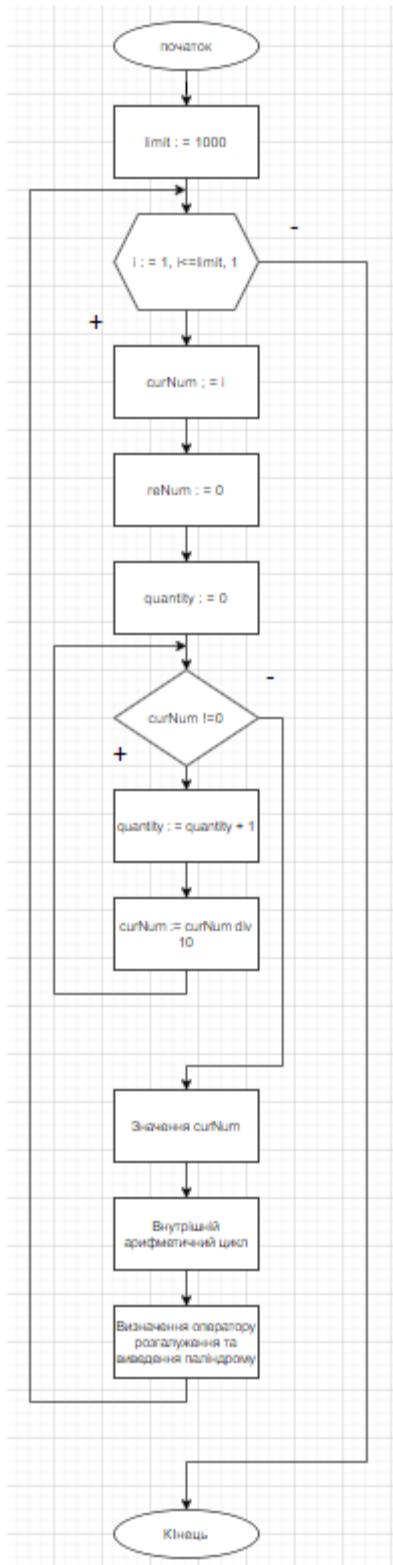
Крок 4.



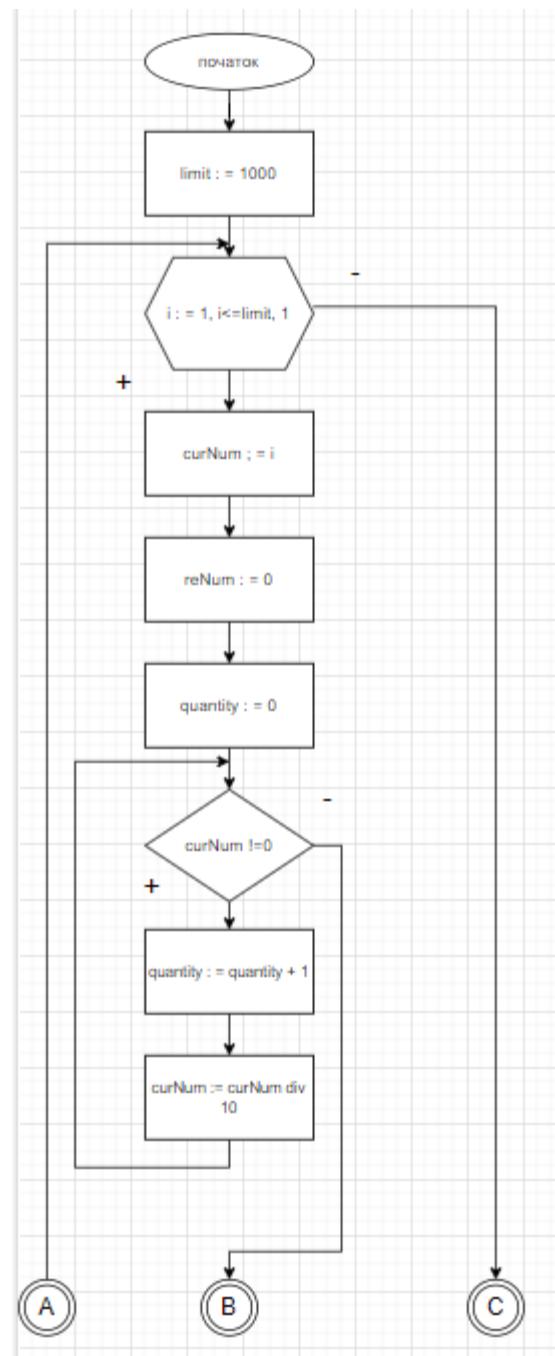
*Крок 5.*

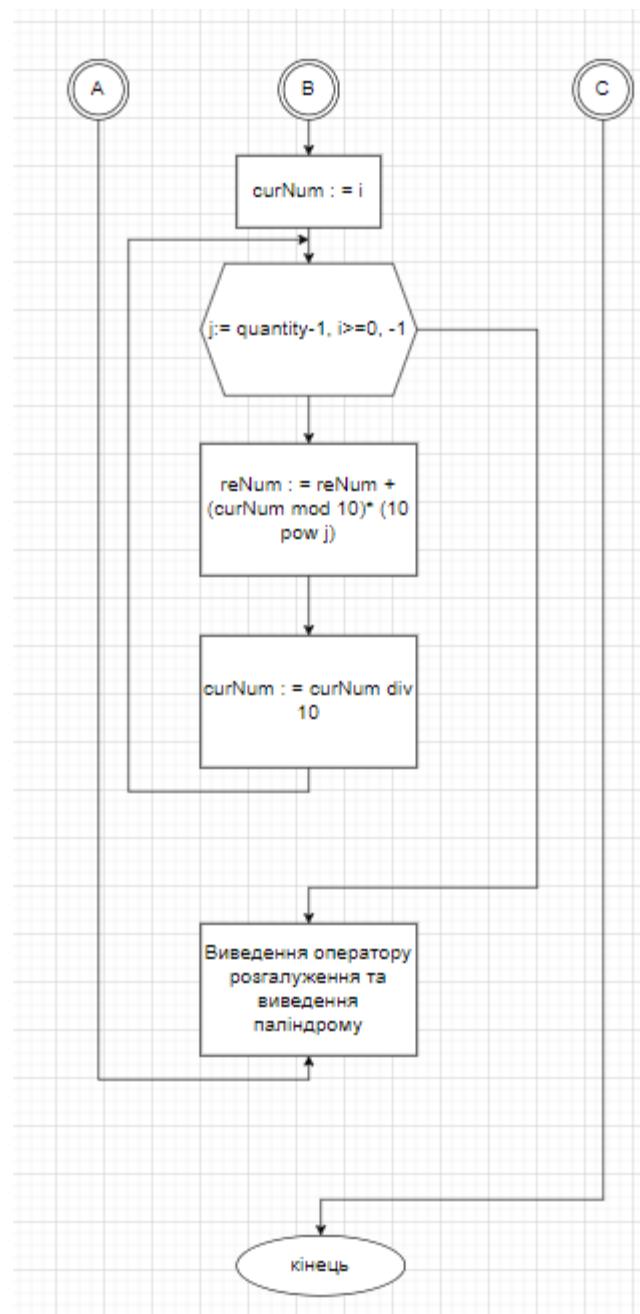


*Крок 6.*

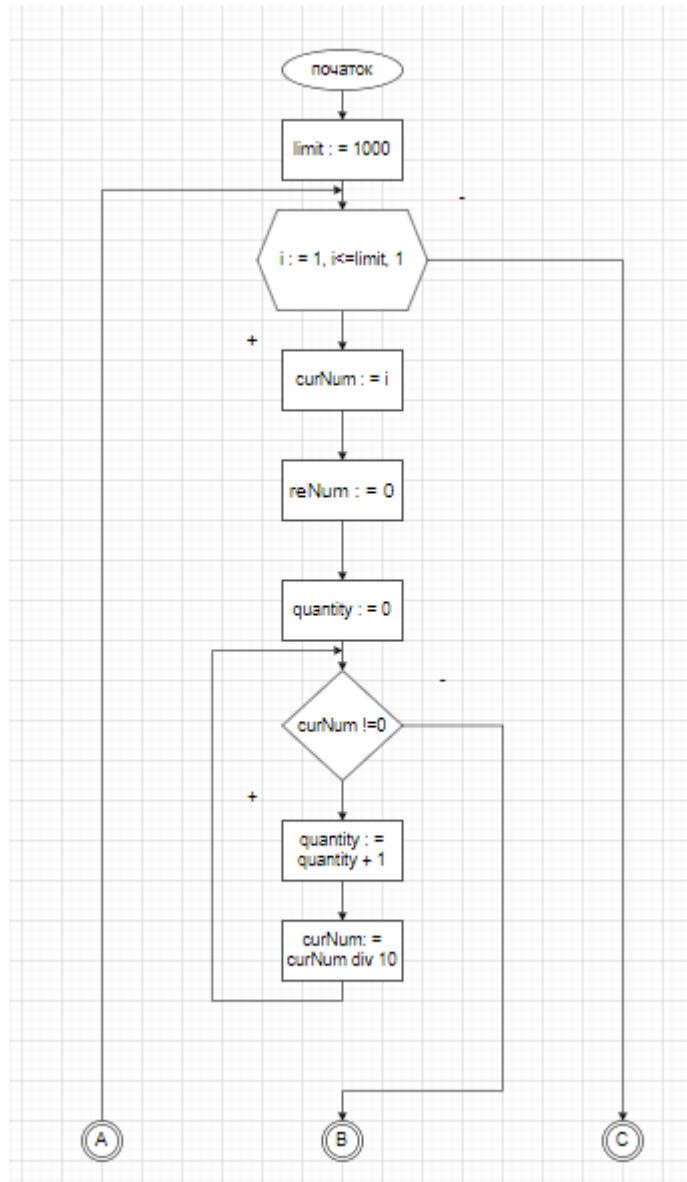


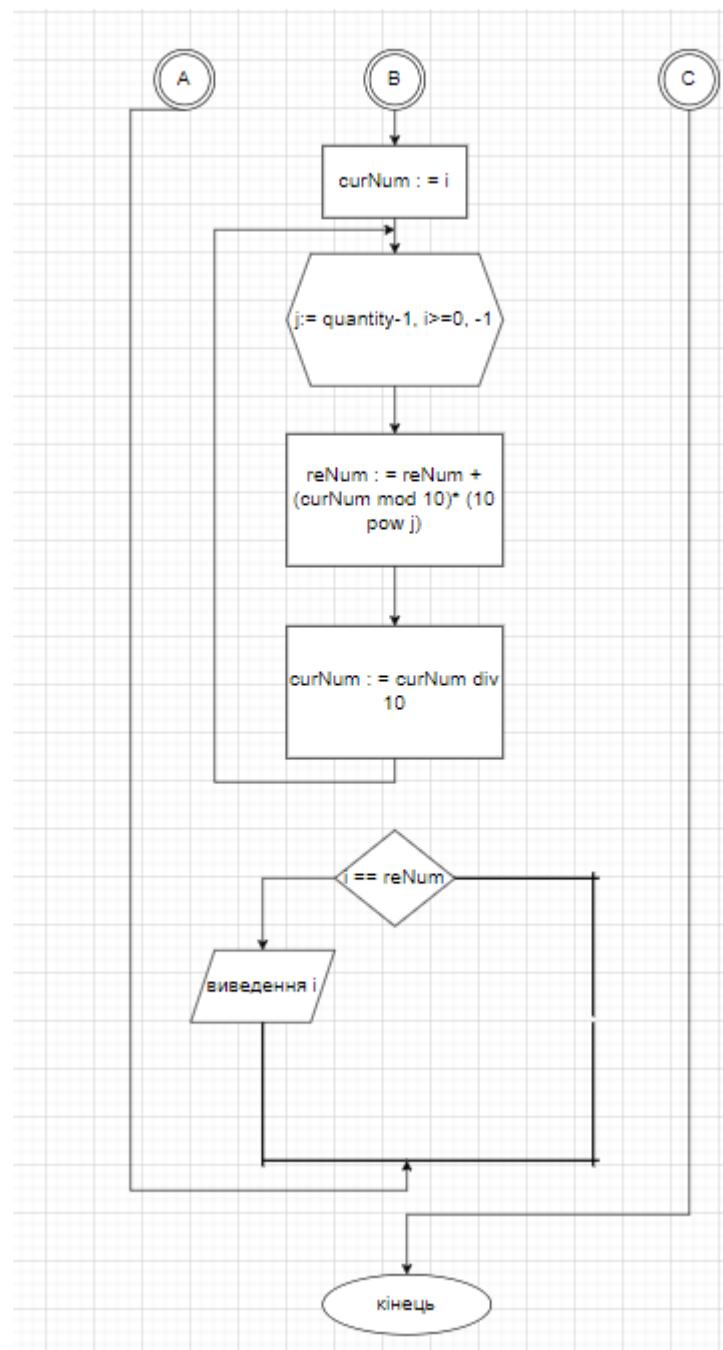
*Крок 7.*





## *Крок 8.*





### Тестування

Блок	Дія (зовнішній цикл)	Дія (внутрішні цикли)
	Початок	

1	limit = 1000	
2	i = 1, 1 <= 1000 → true	
3	curNum = 1, reNum = 0, quantity = 0	
4		1 != 0 → true
5		quantity = 0 + 1 = 1
6		curNum = 1 div 10 = 0
7		0 != 0 → false
8	curNum = 1	
9		j = 0, 0 >= 0 → true
10		reNum = 0 + 1*(10 pow 0) = 1
11		curNum = 1 div 10 = 0
12		j = -1, -1 >= 0 → false
13	1 == 1 → true	
14	Виведення 1	
15	i = 2, 1 <= 1000 → true	
	...	
16	i = 1000, 1000 <= 1000 → true	
17	curNum = 1000, reNum = 0, quantity = 0	
18		1000 != 0 → true
19		quantity = 0 + 1 = 1

21		curNum = 1000 div 10 = 100
----	--	----------------------------

22		$100 \neq 0 \rightarrow \text{true}$
23		$\text{quantity} = 1 + 1 = 2$
24		$\text{curNum} = 100 \text{ div } 10 = 10$
25		$10 \neq 0 \rightarrow \text{true}$
26		$\text{quantity} = 2 + 1 = 3$
27		$\text{curNum} = 10 \text{ div } 10 = 1$
28		$1 \neq 0 \rightarrow \text{true}$
29		$\text{quantity} = 3 + 1 = 4$
30		$\text{curNum} = 1 \text{ div } 10 = 0$
31		$0 \neq 0 \rightarrow \text{false}$
32	$\text{curNum} = 1000$	
33		$j = 3, 3 \geq 0 \rightarrow \text{true}$
34		$\text{reNum} = 0 + 0 * (10 \text{ pow } 3) = 0$
35		$\text{curNum} = 1000 \text{ div } 10 = 100$
36		$j = 2, 2 \geq 0 \rightarrow \text{true}$
37		$\text{reNum} = 0 + 0 * (10 \text{ pow } 2) = 0$
38		$\text{curNum} = 100 \text{ div } 10 = 10$
39		$j = 1, 1 \geq 0 \rightarrow \text{true}$
40		$\text{reNum} = 0 + 0 * (10 \text{ pow } 1) = 0$
41		$\text{curNum} = 10 \text{ div } 10 = 1$
42		$j = 0, 0 \geq 0 \rightarrow \text{true}$

43		$\text{reNum} = 0 + 1 * (10 \text{ pow } 0) = 1$
44		$\text{curNum} = 1 \text{ div } 10 = 0$
45		$j = -1, -1 \geq 0 \rightarrow \text{false}$
46	$1000 == 1 \rightarrow \text{false}$	
	Кінець	

### *Висновки*

Протягом п'ятогї лабораторної роботи я дослідив особливості роботи складних циклів та набув практичних навичок їх використання під час складання програмних специфікацій. В результат ми отримали алгоритм, що використовує складені циклі та знаходить числа-паліндроми, які не більше 1000.