

Додаток 1

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

«Дослідження складних циклічних алгоритмів »

Варіант 4

Виконав студент ІП-15, Бутов Даниїл Романович
(шифр, прізвище, ім'я, по батькові)

Перевірів Всечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Лабораторна робота 5 Дослідження складних циклічних алгоритмів

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

Варіант 4

Завдання. Визначити усі «щасливі» шестизначні числа. Число вважається «щасливим», якщо сума перших його трьох цифр дорівнює сумі останніх трьох цифр.

Постановка задачі:

Нам задані числа на проміжку $[-999\ 999 ; -100\ 000]$ и $[100\ 000 ; 999\ 999]$. Результатом завдання буде виведення чисел, де сума перших трьох цифр дорівнює сумі останніх.

Побудова математичної моделі:

Змінна	Тип	Ім'я	Призначення
Перша сума трьох цифр	Цілий	firstSum	Проміжкове
Друга сума трьох цифр	Цілий	secondSum	Проміжкове

Щоб знайти “щасливі” числа використані вкладені цикли та оператор розгалуження. В основному циклі лічильником буде значення i , яке приймає число з нашого діапазону. У другому циклі ми знаходимо значення суми перших трьох цифр та останніх. В операторі розгалуження перевіряємо їх на однаковість з виводом числа. Суму будемо знаходити за допомогою степеня, цілочисельного ділення - “/”, та для отримання остачі від ділення використовуємо - “%”. У виведенні ми також будемо виводити i з мінусом, бо на мінусовому діапазоні числа не відрізняються від додатнього. У другому циклі лічильник - j , який має діапазон : $[0;3)$. Піднесення до степеня - **pow()**.

Розв’язання:

Програмні специфікації запишемо у псевдокодi та графічній формi у вигляді блок-схеми

Крок 1. Визначимо основні дії.

Крок 2. Деталізуємо основний цикл перебору чисел.

Крок 3. Оновлення змінних.

Крок 4. Деталізуємо цикл знаходження суми. (firstSum, secondSum)

Крок 5. Деталізуємо перевірку сум на рівність.

Псевдокод:

Крок 1.

Початок

Деталізуємо основний цикл перебору чисел

Оновлення змінних

Деталізуємо цикл знаходження суми

Деталізуємо перевірку сум на рівність

Кінець

Крок 2.

Початок

для i від 100 000 до 999 999

Оновлення змінних

Деталізуємо цикл знаходження суми

Деталізуємо перевірку сум на рівність

все повторити

Кінець

Крок 3.

Початок

для i від 100 000 до 999 999

firstSum = 0

secondSum = 0

Деталізуємо цикл знаходження суми

Деталізуємо перевірку сум на рівність

все повторити

Виведення i , -i

Кінець

Крок 4.

Початок

для i від 100 000 до 999 999

$firstSum = 0$

$secondSum = 0$

для j від 0 до 3

$firstSum += i / \text{pow}(10, 6-j-1) \% 10$

$secondSum += i / \text{pow}(10, j) \% 10$

все повторити

Деталізуємо перевірку сум на рівність

все повторити

Кінець

Крок 5.

Початок

для i від 100 000 до 999 999

$firstSum = 0$

$secondSum = 0$

для j від 0 до 3

$firstSum += i / \text{pow}(10, 6-j-1) \% 10$

$secondSum += i / \text{pow}(10, j) \% 10$

все повторити

якщо $firstSum == secondSum$

то

Виведення i , $-i$

все якщо

все повторити

Кінець

Блок схема:

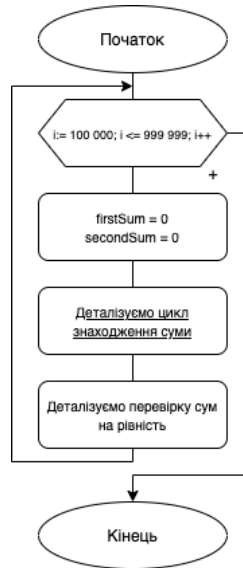
Крок 1



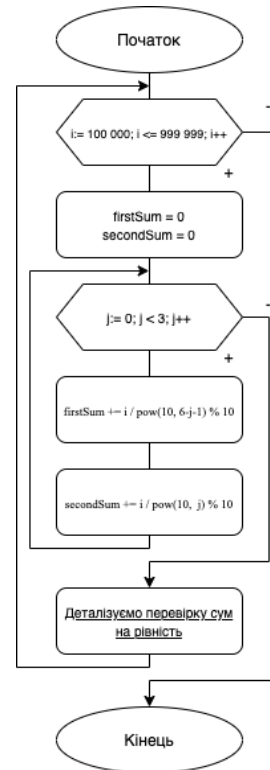
Крок 2



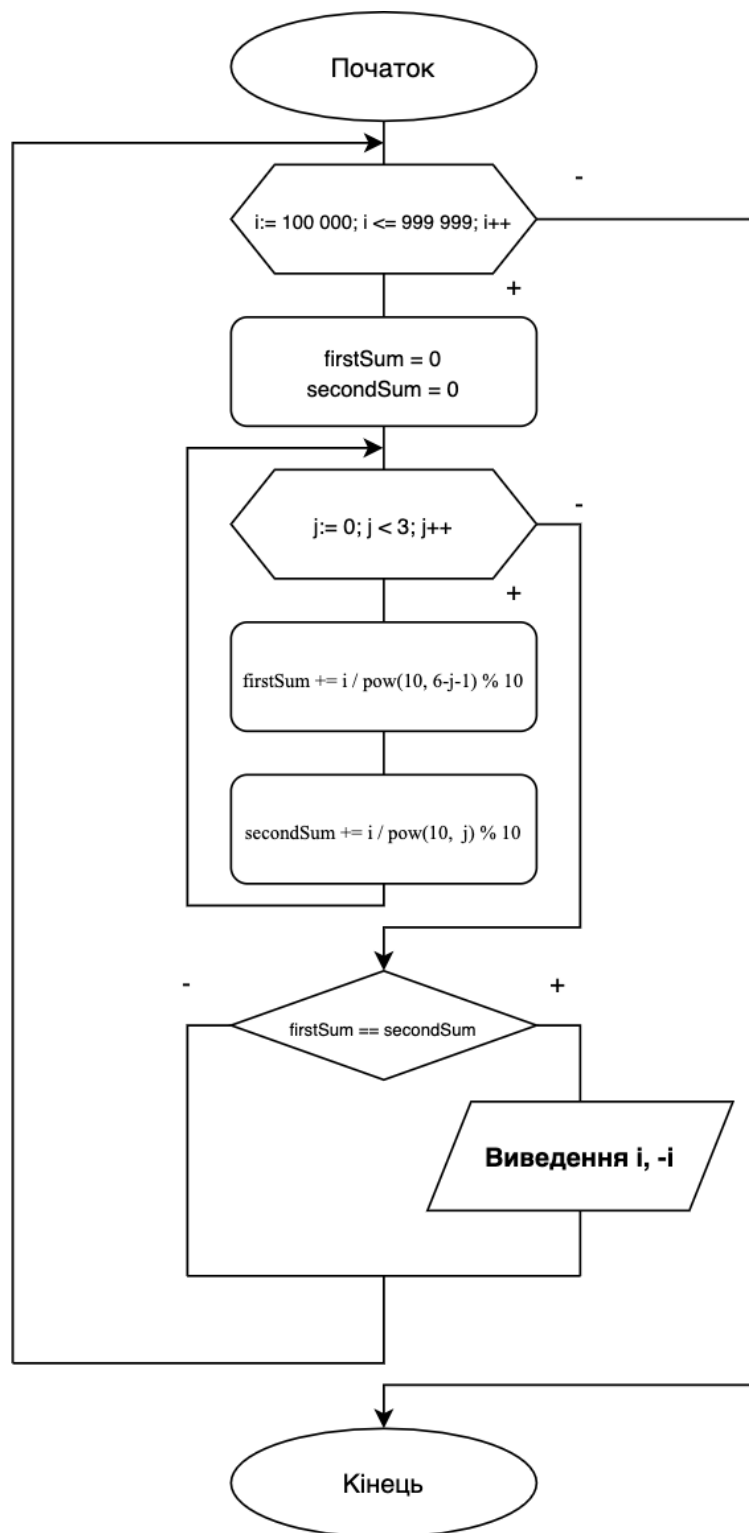
Крок 3



Крок 4



Крок 5



Випробування:

Блок	Дія
	Початок
	$i = 100\,000$
1	$firstSum = 0, secondSum = 0$
2	$firstsSum = 100\,000 / 10^6 \% 10; = 1; firstsSum = 100\,000 / 10^5 \% 10; = 0; firstsSum = 100\,000 / 10^4 \% 10; = 0;$
3	$secondSum = 100\,000 / 10^0 \% 10 = 0; secondSum = 100\,000 / 10^1 \% 10 = 0; secondSum = 100\,000 / 10^2 \% 10 = 0;$
4	$firstSum = 1; secondSum = 0; firstSum \neq secondSum$
	$i = 100\,001$
1	$firstSum = 0, secondSum = 0$
2	$firstsSum = 100\,001 / 10^6 \% 10; = 1; firstsSum = 100\,001 / 10^5 \% 10; = 0; firstsSum = 100\,001 / 10^4 \% 10; = 0;$
3	$secondSum = 100\,001 / 10^0 \% 10 = 1; secondSum = 100\,001 / 10^1 \% 10 = 0; secondSum = 100\,001 / 10^2 \% 10 = 0;$
4	$firstSum = 1; secondSum = 1; firstSum = secondSum$
	Виведення $i, -i$
	Кінець

Висновок – ми дослідили особливості роботи складних циклів та набули практичних навичок їх використання під час складання програмних специфікацій. Склали алгоритм знаходження чисел за специфічною ознакою в великому діапазоні, що надало нам змогу зрозуміти та навчитися складати алгоритми з різною вкладеністю.