

# **Алгоритми та структури даних. Основи алгоритмізації**

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. Основи алгоритмізації»

«Дослідження складних циклічних алгоритмів»

Варіант 28

Виконав студент: ІП-15 Рибаків Дмитро Вадимович

Перевірів: Вечерковська Анастасія Сергіївна

Київ 2021

# Алгоритми та структури даних. Основи алгоритмізації

## Лабораторна робота 4

### Дослідження складних циклічних алгоритмів

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

#### Індивідуальне завдання

#### Варіант 28

#### Постановка задачі

Розробити алгоритм, псевдокод та блок-схему, щоб отримати всі піфагорові трійки натуральних чисел, кожне з яких не перевищує  $n$ , тобто всі такі трійки натуральних чисел  $a, b, c$ , що  $a^2 + b^2 = c^2$  ( $a \leq b \leq c \leq n$ ) використовуючи складний циклічний алгоритм.

#### Побудова математичної моделі

Змінна	Тип	Ім'я	Призначення
Задане число	цілий	$n$	Вхідні дані
Натуральне число, частина піфагорової трійки	цілий	$a$	Вихідні дані
Натуральне число, частина піфагорової трійки	цілий	$b$	Вихідні дані
Натуральне число, частина піфагорової трійки	цілий	$c$	Вихідні дані

У роботі використовується зовнішній арифметичний цикл зі змінною-лічильником  $c = 1$ , умовою  $c \leq 0$ , кроком  $c++$ , внутрішні арифметичними цикли зі змінною-лічильником  $a = 1$ , умовою  $a \leq 0$ , кроком  $a++$  і  $b = a$ , умовою  $b \leq 0$ , кроком  $b++$ . У циклах  $c$  та  $a$  мають початкове значення 1, бо це натуральні числа, а  $b$  має початкове значення  $a$ , щоб уникнути повторення піфагорових трійок. Також у роботі використовуються наступні дії:

« $\leq$ » - менше або рівно;

« $++$ » - додавання одиниці до числа;

«==» - дорівнює (умова). Також у псевдокодi використовується «,» - що означає виведення коми для відокремлювання виведених даних і «; » для перелічення результатів та пробілу між ними.

## **Розв'язання**

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

Крок 1. Визначимо основні дії.

Крок 2. Введення натурального числа та деталізація дії отримання піфагорових трійок менших за введене натуральне число.

## **Псевдокод**

крок 1

**початок**

введення змінної n

отримання піфагорових трійок менших за введене натуральне  
число

**кінець**

крок 2

**початок**

ввід n

**для c від 1 до n включно повторити**

**якщо  $c*c == a*a + b*b$**

**то вивід a «,» b «,» c «; »**

**для a від 1 до n включно повторити**

**якщо  $c*c == a*a + b*b$**

**то вивід a «,» b «,» c «; »**

**для b від a до n включно повторити**

**якщо  $c*c == a*a + b*b$**

**то вивід a «,» b «,» c «; »**

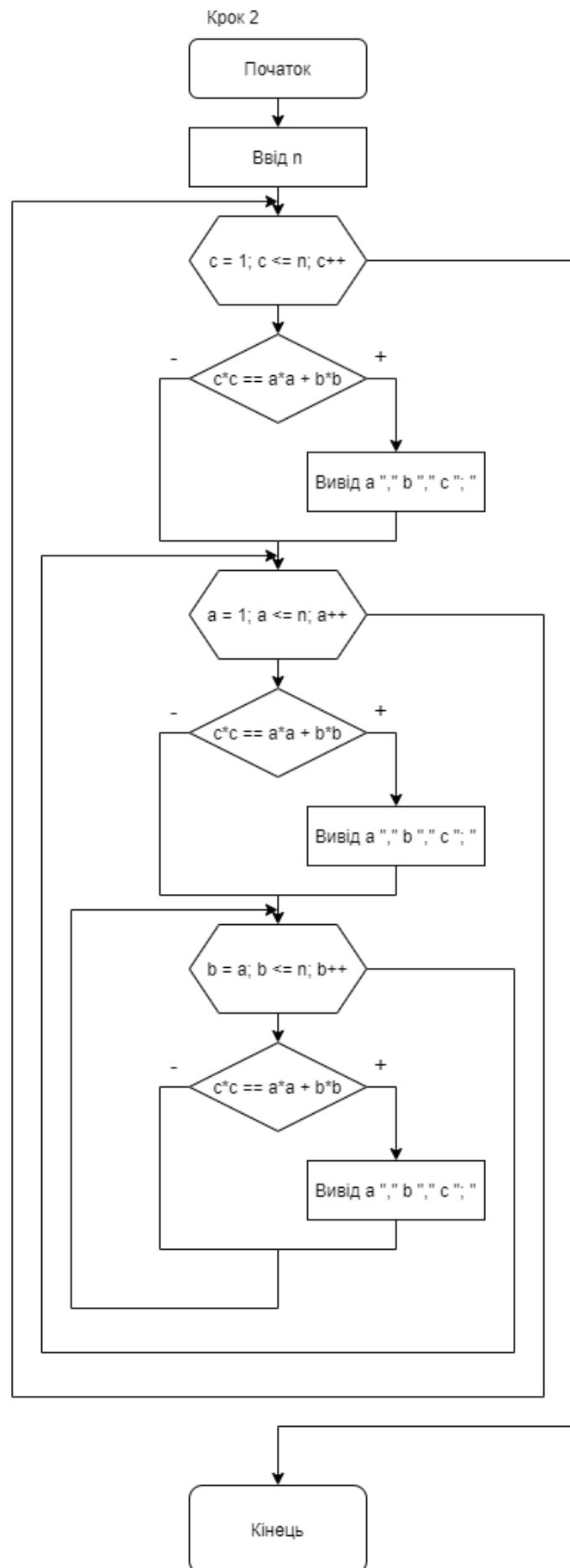
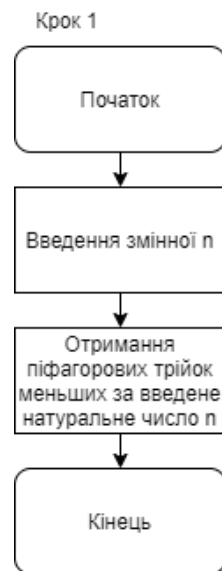
**все повторити**

**все повторити**

**все повторити**

**кінець**

## Блок-схема



## Випробування

Блок	Дія
	Початок
1	$n = 10$
2	$a = 3$
3	$b = 4$
4	$c = 5$
5	Вивід 3,4,5;
6	$a = 6$
7	$b = 8$
8	$c = 10$
9	Вивід 6,8,10;
	Кінець

Блок	Дія
	Початок
1	$n = 16$
2	$a = 3$
3	$b = 4$
4	$c = 5$
5	Вивід 3,4,5;
6	$a = 6$
7	$b = 8$
8	$c = 10$
9	Вивід 6,8,10;
10	$a = 5$
11	$b = 12$
12	$c = 13$
13	Вивід 5,12,13;
14	$a = 9$
15	$b = 12$
16	$c = 15$
17	Вивід 9,12,15;
	Кінець

## Висновки

На цій лабораторній роботі ми дослідили особливості роботи складних циклічних алгоритмів та набули практичних навичок їх використання під час складання програмних специфікацій. В результаті виконання лабораторної роботи ми отримали алгоритм виведення піфагорових трійок значення яких не перевищують значення заданого натурального числа, при цьому розділили

виконання задачі на 2 кроки: визначення основних дій, введення натурального числа і деталізація дії отримання піфагорових трійок менших за введене натуральне число. Розробили псевдокод та блок-схему. В процесі випробування ми розглянули два випадки: введення натурального числа 10 і виведення результату - 3,4,5; 6,8,10; та введення натурального числа 16 і виведення результату - 3,4,5; 6,8,10; 5,12,13; 9,12,15;. Алгоритм ефективен та результативен, він виводить усі піфагорові трійки які менші за задане натуральне число без повторень.