

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

«Дослідження ітераційних циклічних  
алгоритмів»

Варіант 27

Виконав студент ІП-13 Паламарчук Олександр Олександрович  
(шифр, прізвище, ім'я, по батькові)

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

## 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 27

Обчислити значення квадратного кореня із числа  $a > 0$  із заданою точністю  $\epsilon$  на основі рекурентного співвідношення:

$$x_{n+1} = \frac{1}{2} \times \left[ x_n + \frac{a}{x_n} \right], x_0 = \frac{a}{2}$$

де  $x_n$  - попереднє,  $x_{n+1}$  - наступне наближення до кореня. Точність обчислення вважається досягнутою, коли  $|x_{n+1} - x_n| < 10^{-5}$ .

#### ◆ Постановка задачі

Вхідним даним є число  $a > 0$ . На основі рекурентного співвідношення обчислити корінь із цього числа до заданої точності  $\epsilon$ . Точність обчислення вважати досягнутою коли  $|x_{n+1} - x_n| < 10^{-5}$ . Вивести отриманий результат.

#### ◆ Побудова математичної моделі

Складемо таблицю змінних.

Змінна	Тип	Призначення
Задане число $a$	Дійсне	Початкове дане
Поточне значення виразу $x$	Дійсне	Проміжкове значення
Наступне значення виразу $x_{n\_1}$	Дійсне	Проміжкове значення

Складемо таблицю констант.

Константа	Тип	Призначення
$\epsilon$	Дійсне	Точність обчислення

Складемо таблицю функцій.

Оператор	Назва	Призначення
$\text{abs}(a)$	Абсолютне значення	Модуль числа $a$

1)  $\epsilon = 0.00001$ .

2) Нульовий член послідовності знаходимо за формулою:  $x = a/2$

### ◆ Розв'язання

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

*Крок 1.* Визначимо основні дії

*Крок 2.* Перевіримо  $a$  та обчислимо  $x_{n\_1}$  з вказаною точністю за допомогою циклічного оператора з передумовою.

### ◆ Псевдокод алгоритму

*Крок 1.*

**Початок**

Введення  $a$ .

Перевірка  $a$  та обчислення  $x_{n\_1}$  з вказаною точністю.

Виведення  $x_{n\_1}$

**Кінець**

*Крок 2.*

**Початок**

Введення  $a$ .

**якщо**  $a > 0$

**то**

$x = a/2$

$x_{n\_1} = 1/2 * (x + a/x)$ .

**повторити**

$x = x_{n\_1}$ .

$x_{n\_1} = 1/2 * (x + a/x)$ .

**поки**  $\text{abs}(x_{n\_1} - x) > \epsilon$

**все повторити**

**інакше**

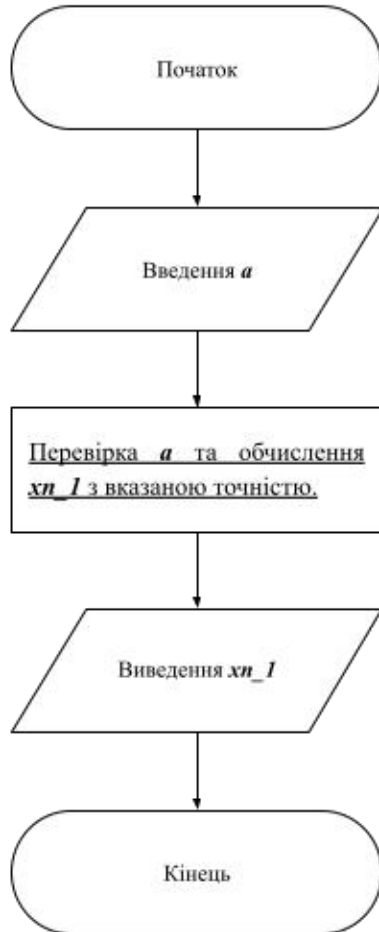
**все якщо**

Виведення  $x_{n\_1}$ .

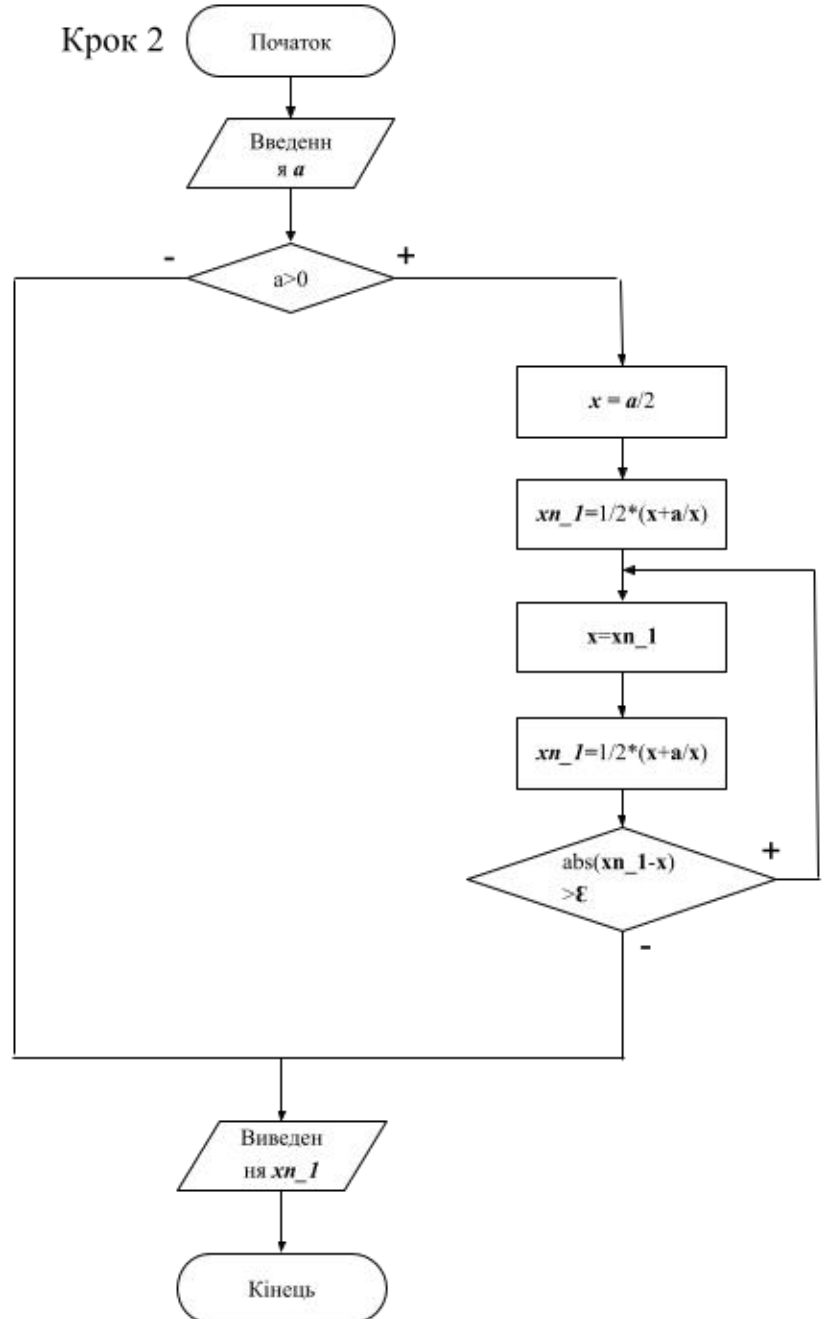
Кінець

◆ Блок схема алгоритму

Крок 1



Крок 2



## ◆ Випробовування алгоритму

### Випробовування №1

Блок	Дія
	<b>Початок</b>
1	Введення: $a=2$ .
2	$2>0$ (true)
3	$x = 2/2$
4	$xn\_1 = 1/2*(1+2/1)$
5	$x = 1.5$
6	$xn\_1 = 1/2*(1.5+2/1.5)$
7	$abs(1.4166667 - 1.5)>\epsilon$ (true)
8	$x = 1.4166667$
9	$xn\_1 = 1/2*(1.4166667+2/1.4166667)$
10	$abs(1.4142157-1.4166667)>\epsilon$ (true)
11	$x = 1.4142157$
12	$xn\_1 = 1/2*(1.4142157+2/1.4142157)$
13	$abs(1.4142135-1.4142157)>\epsilon$ (false)
14	Виведення: $xn\_1 = 1.4142135$
	<b>Кінець</b>

### Випробовування №2

Блок	Дія
	<b>Початок</b>
1	Введення: $a=5$ .
2	$5>0$ (true)
3	$x = 5/2$
4	$xn\_1 = 1/2*(2.5+2/2.5)$
5	$x = 2.25$
6	$xn\_1 = 1/2*(2.25+2/2.25)$

7	$\text{abs}(2.2361112 - 2.25) > \epsilon$ (true)
8	$x = 2.2361112$
9	$xn\_1 = 1/2 * (2.2361112 + 2/2.2361112)$
10	$\text{abs}(2.236068 - 2.2361112) > \epsilon$ (true)
11	$x = 2.236068$
12	$xn\_1 = 1/2 * (2.236068 + 2/2.236068)$
13	$\text{abs}(2.236068 - 2.236068) > \epsilon$ (false)
14	Виведення: $xn\_1 = 2.236068$
	<b>Кінець</b>

## ◆ Висновок

На цій лабораторній роботі було декомпозовано задачу на такі етапи: визначення основних кроків, перевірка  $a$ , обчислення  $xn\_1$  з вказаною точністю, і виведення результату. Було досліджено подання операторів повторення дій та набуто практичних навичок їх використання під час складання циклічних програмних специфікацій.