

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

«Дослідження арифметичних циклічних  
алгоритмів»

Варіант\_\_30\_\_

Виконав студент \_\_\_\_\_ ІП-15 Розін Олексій Іванович \_\_\_\_\_  
(шифр, прізвище, ім'я, по батькові)

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

## Лабораторна робота 4

### Дослідження арифметичних циклічних алгоритмів

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

### Індивідуальне завдання

### Варіант 30

#### Постановка задачі

30. Дано дійсне число  $x$ . Обчислити

$$\frac{(x-2)(x-4)(x-8)\dots(x-64)}{(x-1)(x-3)(x-7)\dots(x-63)}$$

#### Математична модель

Змінна	Тип	Ім'я	Призначення
Задане число	Дійсний	$x$	Початкові дані
Чисельник	Дійсний	numerator	Проміжні дані
Знаменник	Дійсний	denominator	Проміжні дані
Лічильник	Цілий	$i$	Проміжні дані

Ввід змінної  $x$ . Ініціалізуємо змінні numerator та denominator (numerator = 1; denominator = 1). Задаємо арифметичний цикл зі змінною-лічильником  $i = 1$ , умовою  $i \leq 6$  та кроком  $i++$ . Для піднесення числа до степеня використовуємо функцію pow(). В тілі циклу обчислюємо значення чисельника та знаменника за формулами (numerator \*= ( $x - \text{pow}(2, i)$ )); denominator \*= ( $x - (\text{pow}(2, i) - 1)$ )). Далі за допомогою умовного оператора перевіряємо чи дорівнює знаменник нулю: якщо denominator == 0, то виводимо повідомлення про помилку, інакше – виводимо numerator / denominator.

#### Розв'язання

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

Крок 1. Визначимо основні дії.

Крок 2. Введення  $x$ .

Крок 3. Ініціалізація numerator, denominator.

Крок 4. Обчислення numerator, denominator.

Крок 5. Виведення результату.

#### Псевдокод

Крок 1

**початок**

Введення  $x$

Ініціалізація numerator, denominator.

Обчислення numerator, denominator.

Виведення результату

**кінець**

*Крок 2*

**початок**

Введення  $x$

numerator = 1; denominator = 1

Обчислення numerator, denominator.

Виведення результату

**кінець**

*Крок 3*

**початок**

Введення  $x$

numerator = 1; denominator = 1

**повторити**

для  $i$  від 1 до 6

numerator \*= ( $x - \text{pow}(2, i)$ )

denominator \*= ( $x - (\text{pow}(2, i) - 1)$ )

**все повторити**

Виведення результату

**кінець**

*Крок 4*

**початок**

Введення  $x$

$\text{numerator} = 1; \text{denominator} = 1$

**повторити**

для  $i$  від 1 до 6

$\text{numerator} *= (x - \text{pow}(2, i))$

$\text{denominator} *= (x - (\text{pow}(2, i) - 1))$

**все повторити**

**якщо**  $\text{denominator} == 0$

**то**

Вивід “Помилка – ділення на нуль”

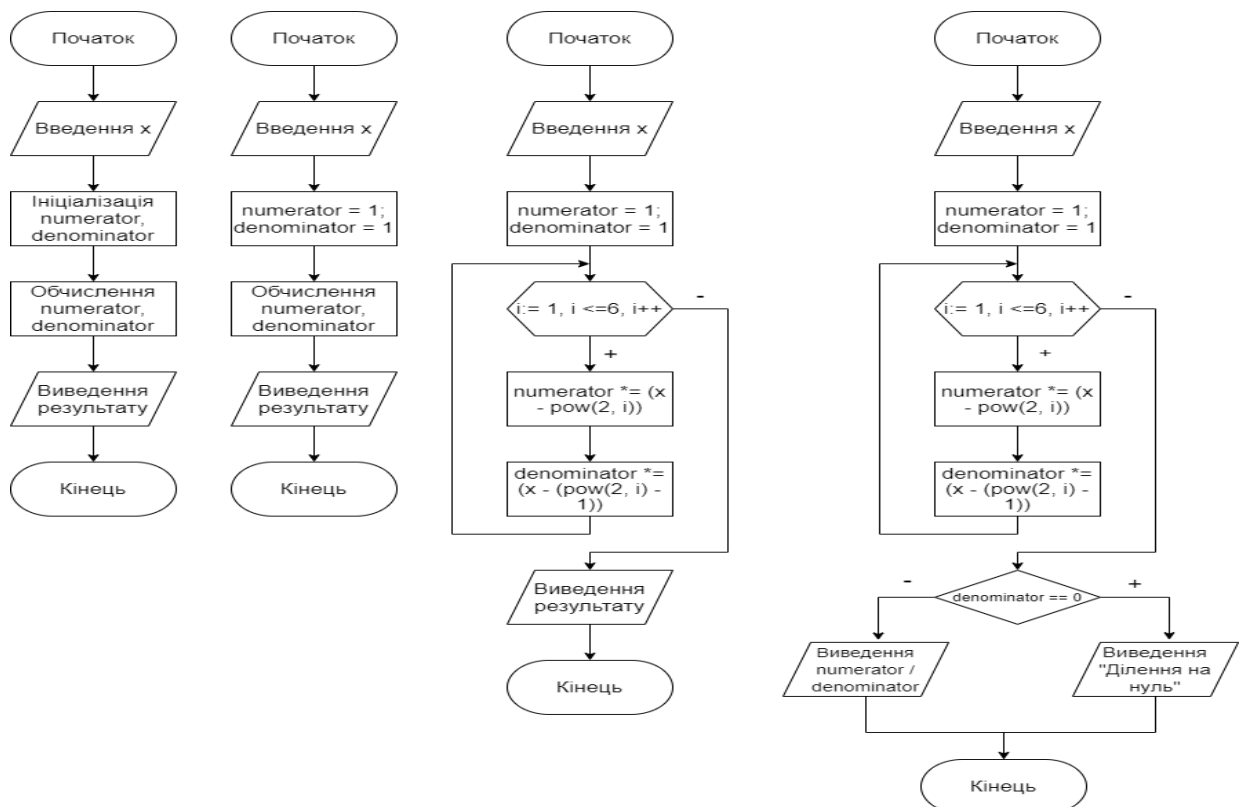
**інакше**

Вивід  $\text{numerator} / \text{denominator}$

**все якщо**

**кінець**

**Блоксхема**



## Випробування

Блок	Дія
Початок	
1	$x = 7$
2	numerator = 1 denominator = 1
3.1 Початок циклу	Numerator = 5 Denominator = 6
3.2	numerator = 15 denominator = 24
3.3	numerator = -15 denominator = 0
3.4	numerator = 135 denominator = 0
3.5	numerator = -3375 denominator = 0
3.6 Кінець циклу	numerator = 192375 denominator = 0
4	Вивід “Помилка – ділення на нуль”
Кінець	

## Висновки

Ми дослідили особливості роботи арифметичних циклів та набули практичних навичок їх використання під час складання програмних специфікацій. В даній лабораторній роботі ми склали програму, яка обчислює конкретний математичний приклад за допомогою використання арифметичного циклу та умовного оператора.