

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 6

Виконав студент ІП-13 Вдовиченко Станіслав Юрійович
(шифр, прізвище, ім'я, по батькові)

Перевірів _____
(прізвище, ім'я, по батькові)

Київ 20211

Лабораторна робота 4

Дослідження арифметичних циклічних алгоритмів

- **Мета** – дослідити особливості роботи арифметичних циклів та набутти практичних навичок їх використання під час складання програмних специфікацій.
- **Постановка задачі (Варіант 6)**

6. Визначити n -не число Фібоначі. Числа Фібоначі визначаються за рекурентною формулою:

$$f_0 = 0, f_1 = 1, f_i = f_{i-1} + f_{i-2}, i > 1.$$

Задані перші два числа послідовності Фібоначі. За рекурентною формулою знаходимо n -ий член послідовності. Щоб знайти будь-який член такої послідовності треба додати два попередніх члена цього ряду.

- **Математична модель**

Змінна	Тип	Ім'я	Призначення
Номер числа	Натуральний і 0	n	Вхідні дані
Лічильник циклу	Натуральний	i	Параметр циклу
Перший компонент послідовності ($n-2$)	Натуральний	previousComponent1	Вхідні дані
Другий компонент послідовності ($n-1$)	Натуральний	previousComponent2	Вхідні дані
Число	Натуральний	component	Вихідні дані

Розв'язання.

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

Крок 1. Визначимо основні дії.

Крок 2. Деталізуємо дію перевірки номера числа(вхідні дані)

Крок 3. Деталізуємо дію обчислення n -го члена послідовності Фібоначі.

Застосуємо цикл з параметром, де

- $i := 1$ – лічильник циклу(початкове значення 1)
- $i < n$ - умова повторення (кінцеве значення). Якщо лічильник дорівнює або більше за номер числа, то закінчуємо цикл.

- $i++$ - збільшуємо лічильник на 1.

В тіло циклу помістимо формулу для знаходження члена послідовності:

$$C_n = C_{n-1} + C_{n-2}$$

- **Псевдокод**

Крок 1.

Початок

Введення даних

Декларування змінних

Перевірка вхідних даних

Обчислення числа

Виведення даних

Кінець

Крок 2.

Початок

Введення n

previousComponent1 := 0, previousComponent2 := 1;

Перевірка вхідних даних

Обчислення числа

Виведення даних

Кінець

Крок 2.

Початок

Введення n

previousComponent1 := 0, previousComponent2 := 1;

якщо (n == 0 || n == 1)

 component := n

все якщо

Перевірка даних

Обчислення числа

Виведення даних

Кінець

Крок 3.

Початок

Введення n

$previousComponent1 := 0, previousComponent2 := 1;$

якщо $(n == 0 \parallel n == 1)$

$component := n$

інакше якщо $(n < 0)$

Виведення “Змінна задана невірно”

все інакше

все якщо

Перевірка даних

Обчислення числа

Виведення даних

Кінець

Крок 4.

Початок

Введення n

$previousComponent1 := 0, previousComponent2 := 1;$

якщо $(n == 0 \parallel n == 1)$

$component := n$

інакше якщо $(n < 0)$

Виведення “Змінна задана невірно”

інакше

для i від 1 до n повторити

Обчислення числа

все повторити

все інакше

все інакше

все якщо

Виведення даних

Кінець

Крок 5.

Початок

Введення n

$\text{previousComponent1} := 0, \text{previousComponent2} := 1;$

якщо $(n == 0 \parallel n == 1)$

$\text{component} := n$

інакше якщо $(n < 0)$

Виведення "Змінна задана невірно"

інакше

для i від 1 до n повторити

$\text{component} := \text{previousComponent1} +$

$\text{previousComponent2};$

$\text{previousComponent1} :=$

$\text{previousComponent2};$

$\text{previousComponent2} :=$

$\text{component}; i++;$

все повторити

все інакше

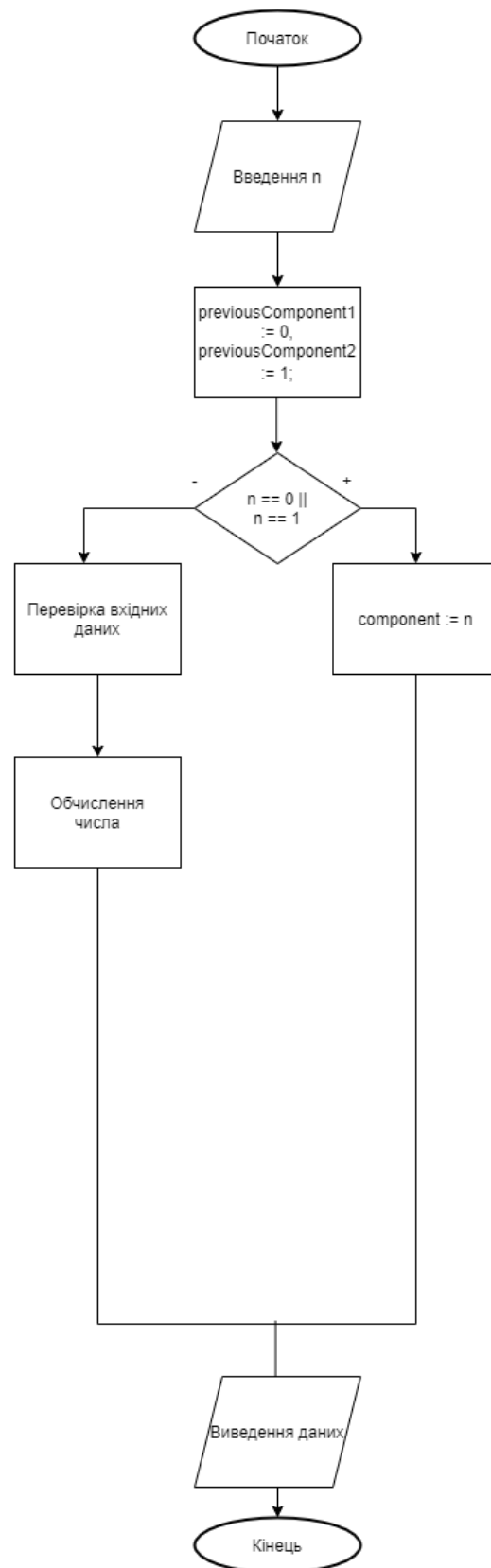
все інакше

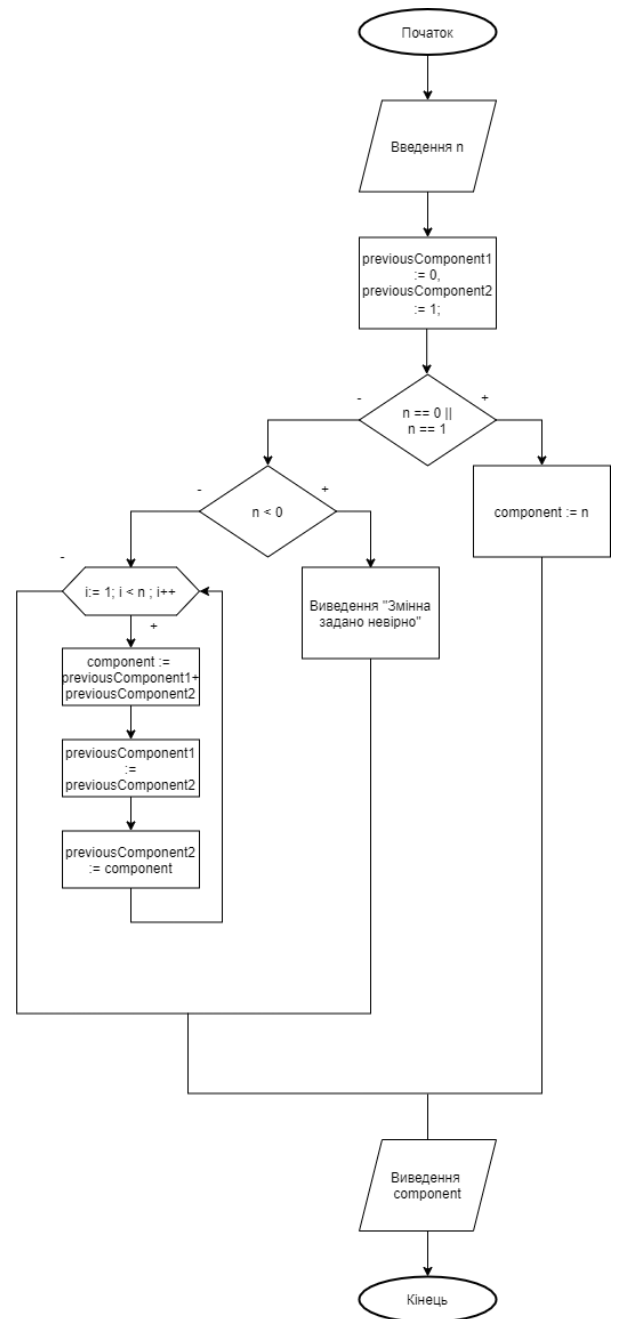
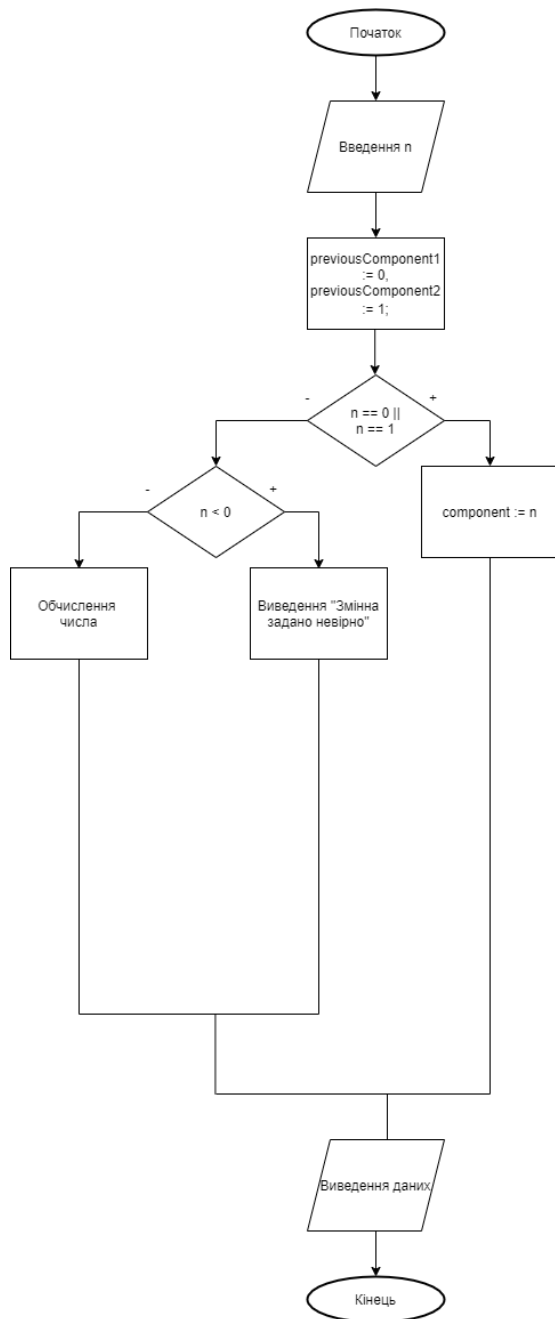
все якщо

Виведення component

Кінець

- Блок-схема





- **Випробування алгоритму**

Блок	Дія
	Початок
1	Введення $n = 5$;
2	$previousComponent1 := 0$, $previousComponent2 := 1$.
3	Перевірка ($n \neq 0$, $n \neq 1$, $n > 0$)
4	($i := 1$, $1 < 5$), $component := 0 + 1 = 1$; $previousComponent1 := 1$, $previousComponent2 := 1; i++$;
5	($i := 2$, $2 < 5$), $component :=$ $1 + 1 = 2$; $previousComponent1 := 1$, $previousComponent2 := 2; i++$;
6	($i := 3$, $3 < 5$), $component :=$ $1 + 2 = 3$; $previousComponent1 := 2$, $previousComponent2 := 3; i++$;
7	($i := 4$, $4 < 5$), $component :=$ $2 + 3 = 5$; $previousComponent1 := 3$, $previousComponent2 := 5; i++$;
8	($i := 5$, $5 = 5$)
9	Виведення $component := 5$
	Кінець

- **Висновок**

В цій лабораторній роботі я дослідив особливості роботи арифметичних циклів та набув практичних навичок їх використання під час складання програмних специфікацій. Розробив алгоритм знаходження числа з послідовності Фібоначчі за його номером за допомогою арифметичного циклу. Створив псевдокод, блок-схему, випробував алгоритм, дослідив значення після кожної ітерації.