# 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 лінійних алгоритмів»

Варіант 25

Виконав студент ІП-11 Прищепа Владислав Станіславович

(шифр, прізвище, ім'я, по батькові)

Перевірив Мартинова Оксана Петрівна

( прізвище, ім'я, по батькові)

Київ 2021

**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.

Індивідуальне завдання: Задане дійсне число х. Послідовність a1, a2, ..., an утворена за законом:

an=((x^2n)\*sin(x^n))/n^2, n = 1, 2, … . Отримати суму a1 + a2 + ... + ak , де х ∈ (-2, 2), k - найменше ціле число, що задовольняє двом умовам: k > 10, | ak | < 10^(-4) .

Варіант 25

**Постановка задачі**

Задати х і розрахувати суму елементів послідовності, яка складається щонайменше з 11 членів і її останній член по модулю менший за 10^-4 за веденим х (х є (-2;2)).

Математична модель:

**Складемо таблицю імен змінних**

|  |  |  |  |
| --- | --- | --- | --- |
| Зміна | Тип | Ім’я | Призначення |
| x | double | x | Вхідне значення |
| Сума членів послідовності | long double | sum | Результат |
| Останній член (виз. к-сті членів) | long long | a | Проміжне значення |
| Останній член (для підрахунку суми) | double | b | Проміжне значення |
| Аргумент синуса у формулі члена | long double | pow1 | Проміжне значення |
| Значення синуса у формулі члена | long double | sin1 | Проміжне значення |
| К-сть членів послідовності | int | k | Проміжне значення |
| К-сть доданків | int | i | Проміжне значення |

Складемо ітераційний цикл, котрий вирахує к-сть членів послідовності (мінімум їх може бути 11), щоб останній член по модулю був менший за 10^-4. Потім складемо другий ітераційний цикл, котрий підрахує суму членів заданої послідовності. **Pow – функція піднесення степеню.**

**Розв’язання**

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

Крок 1. Визначимо основні дії;

Крок 2. Задаємо початкові умови для входження в цикл;

Крок 3. Деталізуємо дію перевірки -2<x та x<2 з використанням альтернативної форми;

Крок 4. Деталізуємо дію знаходження кількості членів послідовності;

Крок 5. Деталізуємо дію знаходження суми членів послідовності.

**Псевдокод**

Крок 1

**Початок**

Задаємо початкові умови для входження в цикл

Перевіряємо чи -2<x та x<2

Знаходимо кількість k членів послідовності

Знаходимо суму членів послідовності

**Кінець**

Крок 2

**Початок**

sum:=0

a:=1

Перевіряємо чи -2<x та x<2

Знаходимо кількість k членів послідовності

Знаходимо суму членів послідовності

**Кінець**

Крок 3

**Початок**

sum:=0

a:=1

**Якщо** -2<x та x<2

**то**

Знаходимо кількість k членів послідовності

Знаходимо суму членів послідовності

**інакше**

Помилка

**Все якщо**

**Кінець**

Крок 4

**Початок**

sum:=0

a:=1

**Якщо** -2<x та x<2

**то**

**повторити поки** abs(a) >= 0.0001

k:= k+1

a := (pow(x, 2 \* k) \* sin(pow(x, k))) / (pow(k, 2))

**все повторити**

Знаходимо суму членів послідовності

**інакше**

Помилка

**Все якщо**

**Кінець**

Крок 5

**Початок**

sum:=0

a:=1

**Якщо** -2<x та x<2

**то**

**повторити поки** abs(a) >= 0.0001

k:= k+1

a := (pow(x, 2 \* k) \* sin(pow(x, k))) / (pow(k, 2))

**все повторити**

**повторити поки** i < k

і:= і+1

pow1 := pow(x, i)

sin1 := sin(pow1)

b := (pow(x, 2 \* i) \* sin1) / (pow(i, 2))

sum := sum + b

**інакше**

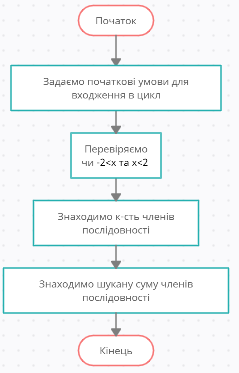
Помилка

**Все якщо**

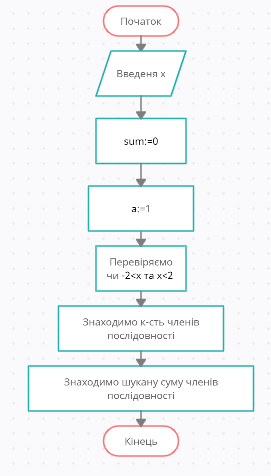
**Кінець**

**Блок-схема**

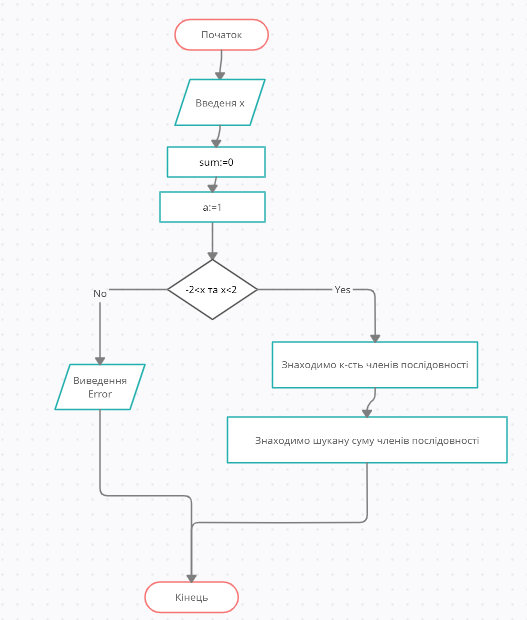
**Крок 1**

****

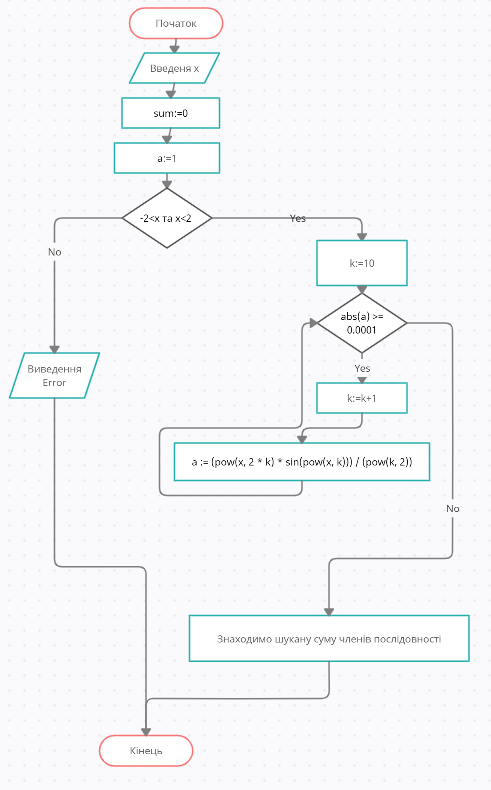
**Крок 2**

****

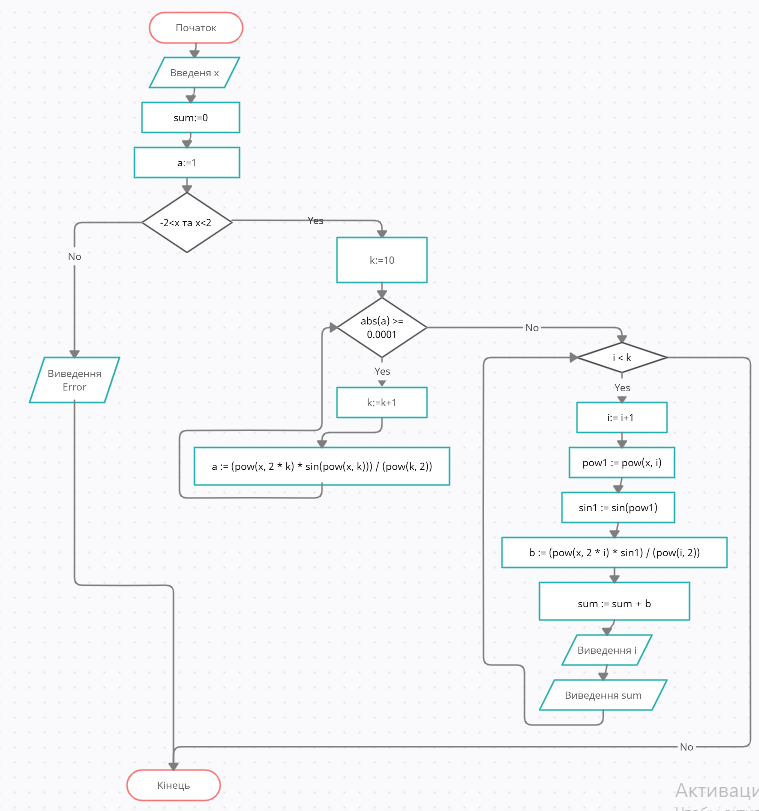
**Крок 3**

****

**Крок 4**

****

**Крок 5**

**Перевірка**

|  |  |  |
| --- | --- | --- |
| **Блок** | **Дія** |  |
|  | **Початок** |  |
| **1** | Введення: х:=0.2 |  |
| **2** | k:=11 |  |
| **3** | і:=1 | sum := 0.00794677 |
| **4** | і:=2 | sum := 0.00796277 |
| **5** | і:=3 | sum := 0.00796283 |
| **6** | і:=4 | sum := 0.00796283 |
| **7** | і:=5 | sum := 0.00796283 |
| **8** | і:=6 | sum := 0.00796283 |
| **9** | і:=7 | sum := 0.00796283 |
| **10** | і:=8 | sum := 0.00796283 |
| **11** | і:=9 | sum := 0.00796283 |
| **12** | і:=10 | sum := 0.00796283 |
| **13** | і:=11 | sum := 0.00796283 |
| **14** | Виведення: sum := 0.00796283 |  |
|  | **Кінець** |  |

**Висновок**

Отже, у результаті лабораторної роботи ми дослідили ітераційні цикли та набули практичних навичок їх створення; навчилися будувати математичну модель, що відповідає постановці задачі, псевдокод та блок-схеми, які пояснюють логіку алгоритму. В результаті виконання лабораторної роботи ми отримали алгоритм для знаходження суми елементів послідовності, яка складається щонайменше з 11 членів і її останній член по модулю менший за 10^-4 за веденим х (х є (-2;2)), декомпозували задачу на 5 кроків: визначили основні дії, задали початкові умови для входження в цикл, потім перевірили, чи справджуються тотожність х є (-2;2), а далі по черзі вирахували к-сть елементів послідовності та розрахували їх суму. У алгоритмі використовується два ітераційних цикли з передумовою поспіль. Оскільки формула перевірені вручну результати розрахунку членів послідовності відповідають дійсності, то алгоритм правильно подає результат.