Алгоритми та структури данних. Основи алгоритмів розгалуження.

*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 алгоритмів розгалуженяя»

Варіант 15

Виконав студент ІП-12, Кириченко Владислав Сергійович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 202 1

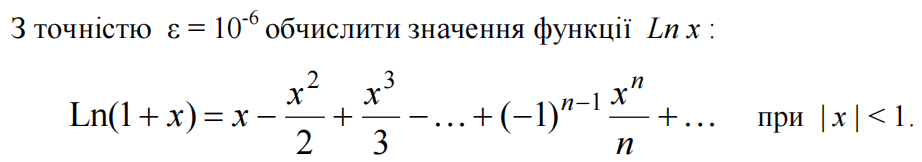
**Лабораторна робота № 3**

**Назва роботи**: Дослідження ітераційних циклічних алгоритмів

**Мета**:Дослідити подання операторів повторення дій та набути практичних навичок їх використання під час складання циклічних програмних специфікацій.

**Варіант 15**

**Умова задачі***:*



**Постановка задачі**: Задана змінна “**х**”, обчислити значення функції Ln(1+**x**) із заданою точністю **e**. Результатом розв’язку задачі є змінна **ln.**

**Побудова математичної моделі:** Маємо змінну “**х**”, в залежності від якої ми повинні знайти значення натурального логарифму числа (**x**+1) (змінна **ln**) із заданою точністю. Для цього скористаємося циклом передумови while і введемо змінну n у якості лічильника. Точність обчислення знаходться за формулою abs(Xn - Xn+1). Також нам буде потрібні функції pow(a,n) - піднесення числа а у степінь n, та abs(a) - модуль числа а.

Складемо таблицю змінних:

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Значення **e** | Дійсний | **e** | Початкові дані |
| Значення **n** | Натуральний | **n** | Початкові дані |
| Значення **x** | Дійсний | **х** | Початкові дані |
| Значення **lnPrevious** | Дійсний | **lnPrevious** | Проміжкове значення |
| Значення **ln** | Дійсний | **ln** | Результат |

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

**Крок 1.** Визначимо основні дії.

**Крок 2.** Перевірка чи задовольняє значення вхідного даного **х** умові задачі.

**Крок 3.** Ініціалізація змінної **lnPrevious.**

**Крок 4.** Деталізація обчислення початкового значення **ln**.

**Крок 5.** Деталізаці знаходження значення **ln** з точністю e.

*Псевдокод:*

*Крок 1.*

**початок**

введення **x,e,n**

перевірка чи abs(**x**) < 1

Ініціалізація змінної **lnPrevious**

обчислення значення **ln**

знаходження значення **ln** з точністю **e** за допомогою цикла

виведення **ln**

**кінець**

Крок 2.

**початок**

введення **x,e,n**

**якщо** abs(**x**) < 1

інізіалізація змінної **lnPrevious**

обчислення значення **ln**

знаходження значення **ln** з точністю **e** за допомогою цикла

виведення **ln**

**інакше**

виведення “дані виходять за межі передбачены умовою задачі”

**все якщо**

**кінець**

Крок 3.

**початок**

введення **x,e,n**

**якщо** abs(**x**) < 1

**lnPrevious** = 0

обчислення значення **ln**

знаходження значення **ln** з точністю **e** за допомогою цикла

виведення **ln**

**інакше**

виведення “дані виходять за межі передбачены умовою задачі”

**все якщо**

**кінець**

Крок 4.

**початок**

введення **x,e,n**

**якщо** abs(**x**) < 1

**lnPrevious** = 0

**ln** = **lnPrevious** + pow(-1,(**n**-1)) \* float(pow(**x**,**n**))/**n**

**n**++

знаходження значення **ln** з точністю **e**

виведення **ln**

**інакше**

виведення “дані виходять за межі передбачены умовою задачі”

**все якщо**

**кінець**

Крок 5.

**початок**

введення **x,e,n**

**якщо** abs(**x**) < 1

**lnPrevious** = 0

**ln** = **lnPrevious** + pow(-1,(**n**-1)) \* float(pow(**x**,**n**))/**n**

**n**++

**поки (**abs(**ln** - **lnPrevious**) > **e) повторити**

**lnPrevious** = **ln**

**ln** += pow(-1,(**n**-1)) \* float(pow(**x**,**n**))/**n**

**n**++

**все повторити**

виведення **ln**

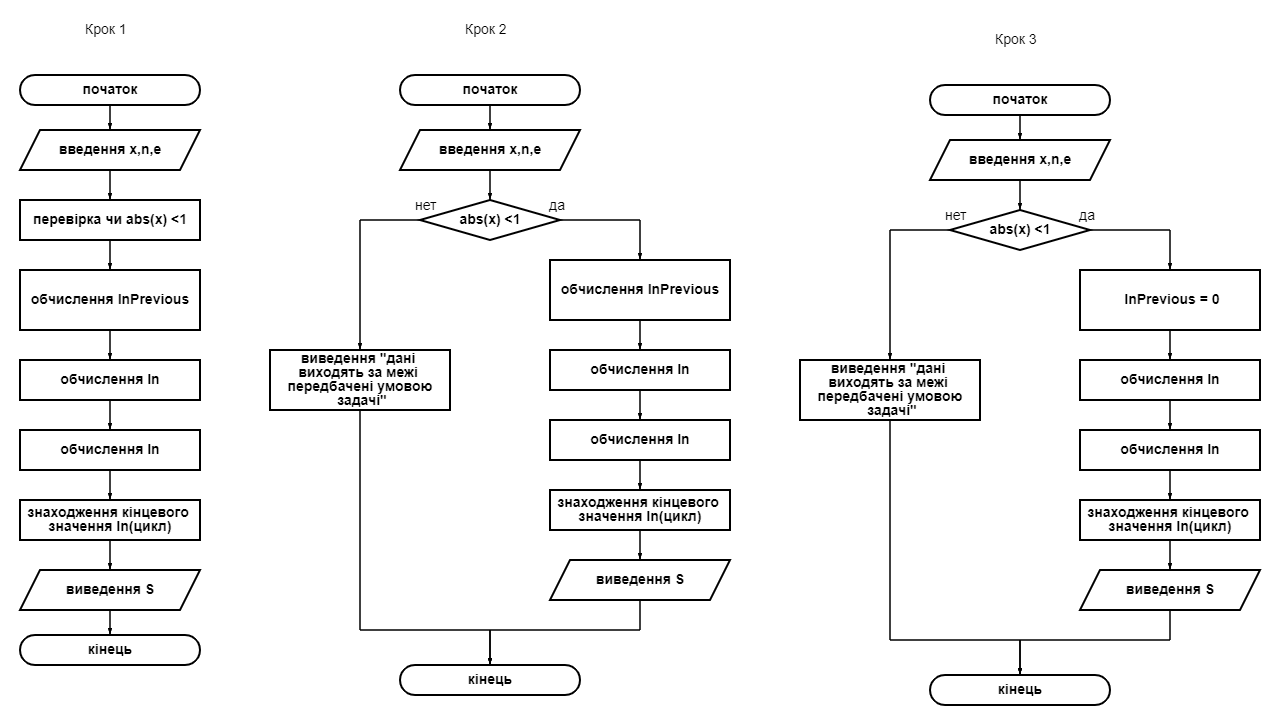
**інакше**

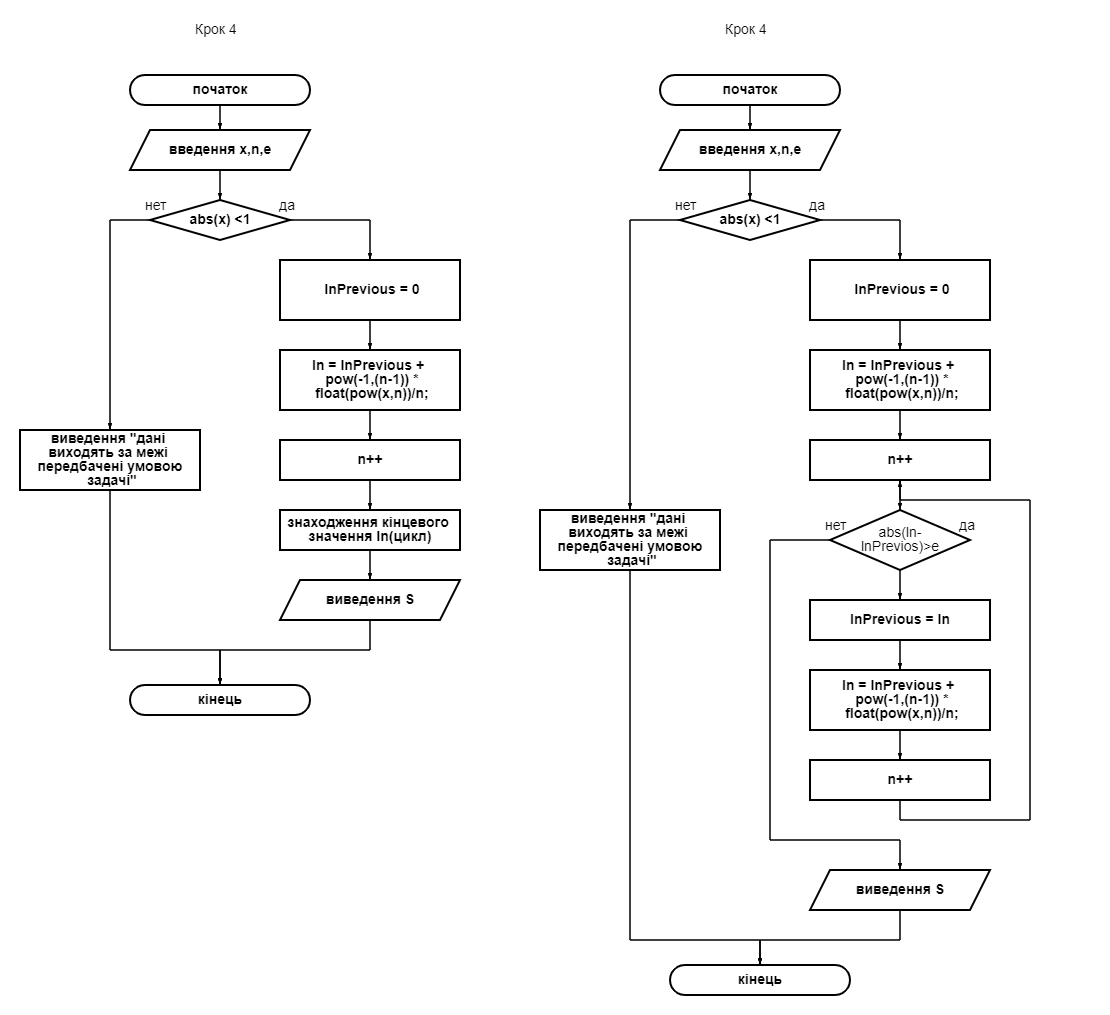
виведення “дані виходять за межі передбачены умовою задачі”

**все якщо**

**кінець**

Блок схема:





1. Перевірка алгоритму

|  |  |  |  |
| --- | --- | --- | --- |
| Блок | Дія | Дія | Дія |
|  | **Початок** | **Початок** | **Початок** |
| 1 | Введення  **x**=0.2, **n**=1,  **e** = 0.000001 | Введення  **x**=-0.3,**n=1**, **e**=0.000001 | Введення  **x**=2,**n=1**, **e**=0.000001 |
| 2 | abs(0.2)<1 - true | abs(-0.3)<1 - true | abs(2)<1 - false |
| 3 | **lnPrevious** = 0 | **lnPrevious** =0 | вивід  «дані виходять за межі передбачены умовою задачі» |
| 4 | **ln** = 0.2 | **ln** = -0.3 | **Кінець** |
| 5 | **виконання циклу**  (результат =>  **ln** =0.182322) | **виконання циклу**  (результат =>  **ln**= -0.356675 |  |
| 6 | Вивід: 0.182322 | Вивід: -0.356675 |  |
|  | **Кінець** | **Кінець** |  |

**Перевірка 1(всі значення ln)**

0.2

0.18

0.182667

0.182267

0.182331

0.18232

0.182322

**Перевірка 2(всі значення ln)**

-0.3

-0.345

-0.354

-0.356025

-0.356511

-0.356633

-0.356664

-0.356672

-0.356674

**Висновок -** Було досліджено подання операторів повторення дій та набуто практичних навичок їх використання під час складання циклічних програмних специфікацій.