

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 22

Виконав студент _____Мешков_Андрій_Ігорович_____

(шифр, прізвище, ім'я, по батькові)

Перевірив _____Вечерковська Анастасія Сергіївна_____

(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.

Варіант 22

Задача. Із заданою точністю обчислити значення математичної константи e :

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots$$

Постанова задачі. Формулою задано значення математичної константи як сума нескінченного ряду чисел. Результатом розв'язку є число Ейлера, обчислене за допомогою розкладу в ряд Маклорена. Де загальною формулою n -того члена є вираз $\frac{1}{n!}$

Змінна	Тип	Ім'я	Призначення
Задана точність	Дробовий	n	Початкові дані
Кількісний коефіцієнт	Натуральний	i	Проміжні дані
Значення факторіалу	Натуральний	fact	Проміжні дані
Попереднє значення математичної константи	Дійсний	prevres	Проміжні дані
Значення математичної константи	Дійсний	res	Результат

Згідно із заданою точністю n , не обмежуючи загальності, приходимо до висновку: коли різниця дійсного та попереднього значення числа стає меншим від n , він автоматично стає незначущим, тобто має виконуватися ітераційний цикл за умови $|res - prevres| \geq n$. Коли цикл відпрацює, то отримуємо значення числа e з потрібною кількістю цифр після коми.

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

Крок 1. Визначимо основні дії.

Крок 2. Деталізуємо дію обчислення початкових значень змінних.

Крок 3. Деталізуємо дію обчислення значення математичної константи e .

Псевдокод

Крок 1

Початок

Ввести n

Обчислення

початкових значень

змінних

Обчислення значення

математичної

константи e

Вивести res

Кінець

Крок 2

Початок

Ввести n

$res:=1$

$fact:=1$

$prevres:=0$

$i:=1$

Обчислення значення

математичної

константи e

Вивести res

Кінець

Крок 3

Початок

Ввести n

$res:=1$

$fact:=1$

$prevres:=0$

$i:=1$

поки $|res-prevres| \geq n$

повторити

$prevres:=res$

$fact:=fact*i;$

$res:=res+1/fact$

$i++$

все повторити

Вивести res

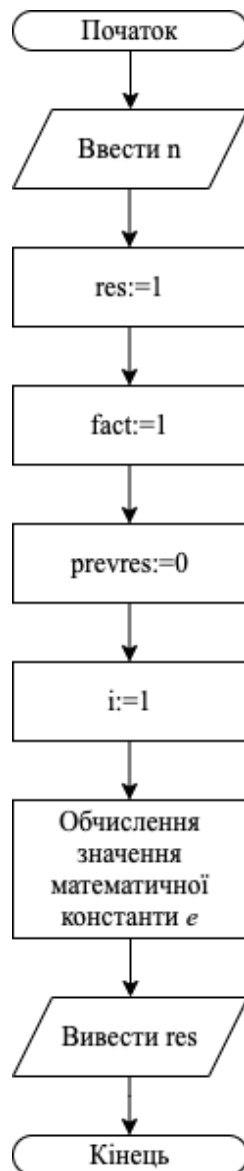
Кінець

Блок-схема

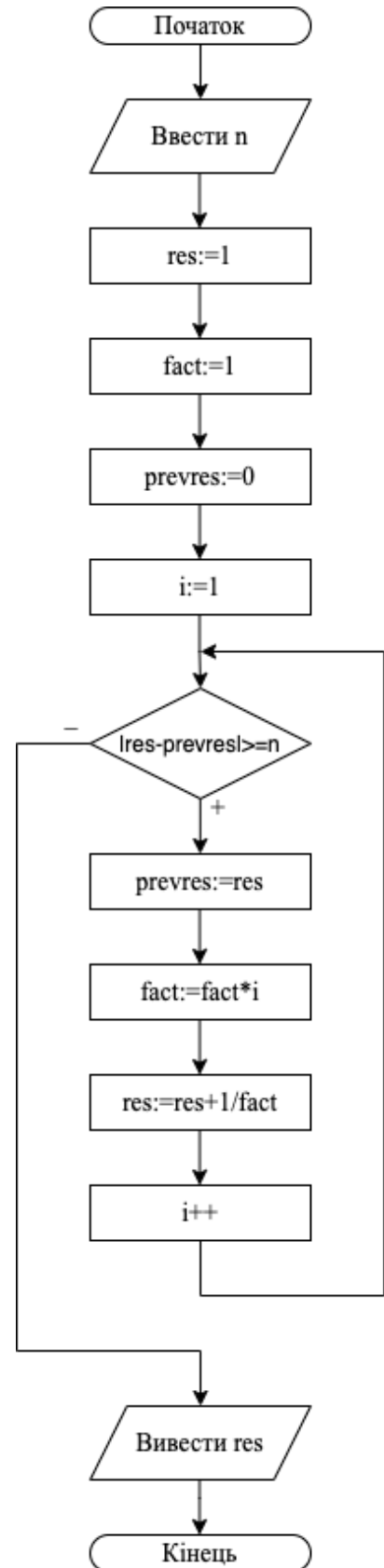
Крок 1



Крок 2



Крок 3



Випробування алгоритму: перевіримо правильність алгоритму на довільних конкретних значеннях початкових даних.

Тест №1

Блок	Дія
1	Початок
2	Введення: $n=0.00001$
3	$res:=1; fact:=1; prevres:=0; i:=1;$
4	Початок циклу. Перевірка умови: $ 1-0 = 1 \geq 0.00001$ – true
5	$prevres:=1; fact:=1; res:=2; i:=2$
6	Початок циклу. Перевірка умови: $ 2-1 = 1 \geq 0.00001$ – true
7	$prevres:=2; fact:=2; res:=2.5; i:=3$
8	Початок циклу. Перевірка умови: $ 2.5-2 = 0.5 \geq 0.00001$ – true
9	$prevres:=2.5; fact:=6; res:=2.66666675; i:=4$
10	Початок циклу. Перевірка умови: $0.16666675 \geq 0.00001$ – true
11	$prevres:=2.66666675; fact:=24; res:=2.70833349; i:=5$
12	Початок циклу. Перевірка умови: $0.04166674 \geq 0.00001$ – true
13	$prevres:=2.70833349; fact:=120; res:=2.71666694; i:=6$
14	Початок циклу. Перевірка умови: $0.00833345 \geq 0.00001$ – true
15	$prevres:=2.71666694; fact:=720; res:=2.71805573; i:=7$
16	Початок циклу. Перевірка умови: $0.00138879 \geq 0.00001$ – true
17	$prevres:=2.71805573; fact:=5040; res:=2.71825409; i:=8$
18	Початок циклу. Перевірка умови: $0.00019836 \geq 0.00001$ – true
19	$prevres:=2.71825409; fact:=40320; res:=2.71827888; i:=9$
20	Початок циклу. Перевірка умови: $0.00002479 \geq 0.00001$ – true
21	$prevres:=2.71827888; fact:=362880; res:=2.71828175; i:=10$
22	Початок циклу. Перевірка умови: $0.00000287 \geq 0.00001$ – false
23	Вивід: $res=2.718282$
24	Кінець

Висновок: було досліджено ітераційні циклічні алгоритми, проаналізовано подане завдання, декомпозовано та виконано. Також були розроблені псевдокод та блок-схема поставленого алгоритму.