**Лабораторна робота 5**

**Дослідження складних циклічних алгоритмів**

**Мета** – дослідити особливості роботи складних циклів та набути практичних навичок їх

використання під час складання програмних специфікацій.

**Варіант №06**

**1.Постановка задачі**

Дано натуральне число n. Перевірити, чи можна подати n! у вигляді добутку трьох

послідовних цілих чисел.

**2.Математична модель:**

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Назва | Тип | Призначення |
| Користувацький ввід | n | Integer | Користувацький ввід |
| Проміжний результат | Factorial | Integer | Значення обчисленного факторіалу |
| Проміжний результат | а | Integer | Перше число із трійки для перевірки |
| Проміжний результат | b | Integer | Друге число із трійки для перевірки |
| Проміжний результат | c | Integer | Третє число із трійки для перевірки |
| Проміжний результат | isFound | Integer | Значення факторіалу |

**Псевдокод**:

**Крок1**

**Початок**

—Отримуємо користувацький ввід

—Ініціалізуємо змінні

—Обчислюємо факторіал

—Пошук послідовних чисел

—Виводимо результат

**Кінець**

**Крок2**

**Початок**

—Отримуємо користувацький ввід

—а = 1, b = 2, c = 3, i = 1

isFound = False, Factorial = 1

—Обчислюємо факторіал

—Пошук послідовних чисел

—Виводимо результат

**Кінець**

**Крок3**

**Початок**

—Отримуємо користувацький ввід

—a = 1, b = 2, c = 3, i = 1

isFound = False, Factorial = 1

—**повторити** від i до n включно

Factorial \*= i

**все повторити**

—Пошук послідовних чисел

—Виводимо результат

**Кінець**

**Крок4**

**Початок**

—Отримуємо користувацький ввід

—a = 1, b = 2, c = 3, i = 1

isFound = False, Factorial = 1

—**повторити** від i до n включно

Factorial \*= i

**все повторити**

**—поки** !isFound and

a \* b \* c < Factorial

a += 1

b += 1

c += 1

isFound = a \* b \* c == Factorial

**все поки**

—Виводимо результат

**Кінець**

**Крок5**

**Початок**

—Отримуємо користувацький ввід

—a = 1, b = 2, c = 3, i = 1

isFound = False, Factorial = 1

—**повторити** від i до n включно

Factorial \*= i

**все повторити**

**—поки** !isFound and

a \* b \* c < Factorial

a += 1

b += 1

c += 1

isFound = a \* b \* c == Factorial

**все поки**

—**якщо** isFound

**то**

Виводимо «a, b, c»

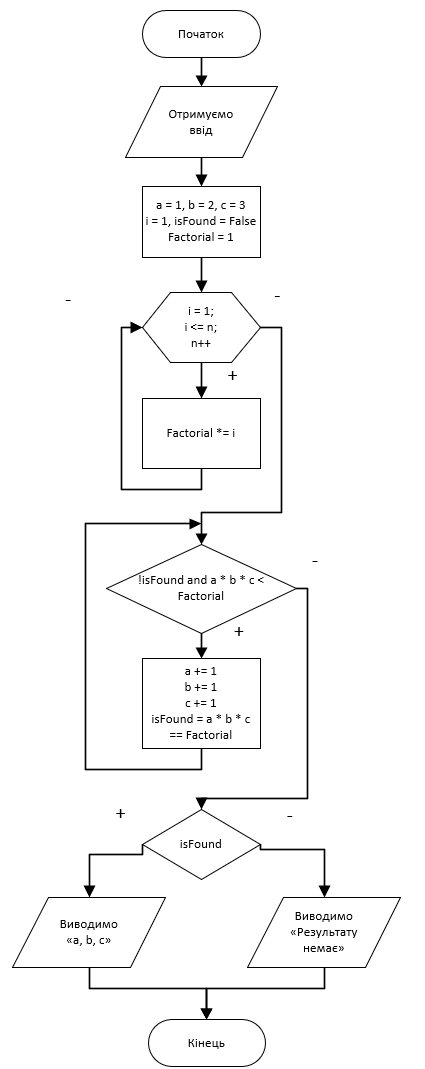
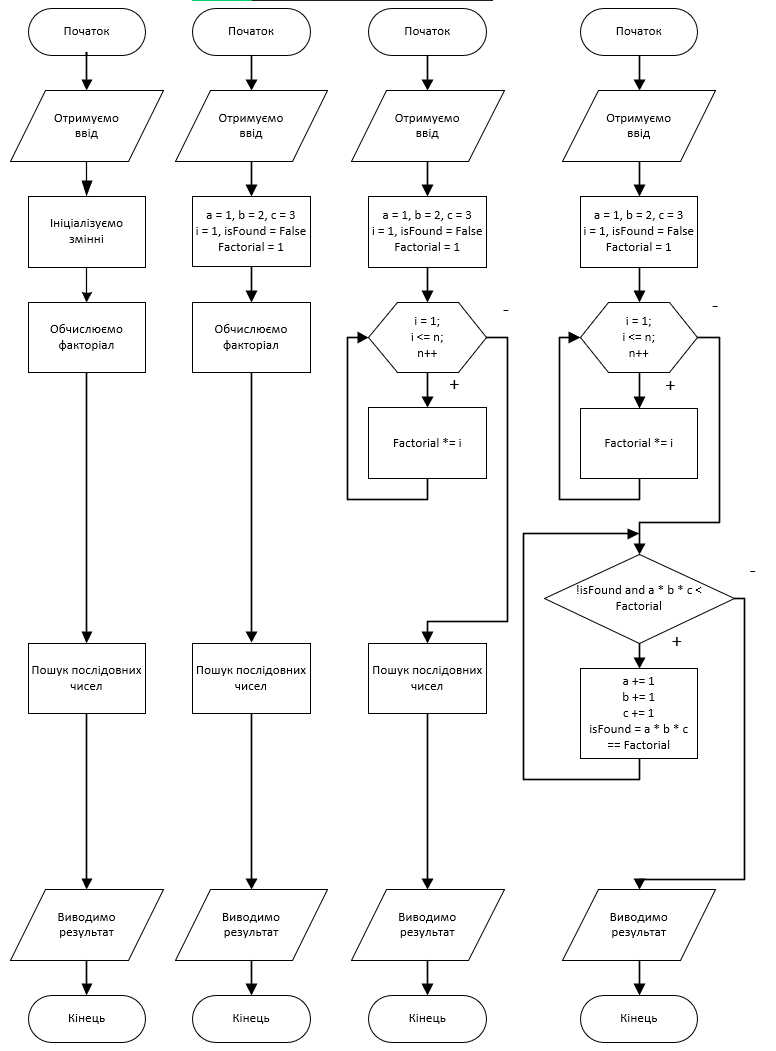
**інакше**

Виводимо «Не знайдено»

**все якщо**

**Кінець**

**Блок-схема:**



Випробування алгоритму:

|  |  |
| --- | --- |
| Блок | Дія |
|  | Початок |
| 1 | n = 5 |
| 2 | a = 1, b = 2, c = 3, i = 1,  isFound = False, Factorial = 1 |
| 3 | Factorial \*= 1 = [1 \* 1] = 1 |
| 7 | Factorial \*= 5 = [24 \* 5] = 120 |
| 8 | a[+=1] = 2  b[+=1] = 3  c[+=1] = 4  isFound = [2 \* 3 \* 4 == 120] = False |
| 10 | a[+=1] = 4  b[+=1] = 5  c[+=1] = 6  isFound = [4 \* 5 \* 6 == 120] = True |
| 11 | Виводимо 4 5 6 |
|  | Кінець |

**Висновок:**

Дослідив особливості роботи складних циклів та набув практичних навичок їх використання під час складання програмних специфікацій.