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 № 5 з дисципліни «Алгоритми та структури даних-1. Основи алгоритмізації»

«Дослідження складних циклічних алгоритмів»

Варіант 16

Виконав студент ІП-11 Кузьменков Дмитро Олегович

(шифр, прізвище, ім'я, по батькові)

Перевірив Мартинова. О. П

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 5**

**Дослідження складних циклічних алгоритмів**

**Мета** - дослідити особливості роботи складних циклів та набути практичних навичок їх використання під час складання програмних специфікацій.

# Завдання:

З чисел Фібоначі в інтервалі від 1 до 100 визначити тільки прості числа, а також їх порядкові номери в ряду Фібоначі.

**Постановка задачі:**

Спочатку задаємо перші два числа Фібоначі та задаємо їхнє значення змінним, після чого знаходимо третє число ряду через суму перших двох, а також задаємо початкове значення показника індексу, що дорівнюватиме 2, тому що ми не виводимо 0 як перший член ряду. Далі просто виводимо перших три числа ряду (1, 1, 2), бо їх неможливо перевірити на простоту за нашим циклом. Використовуємо ітераційний цикл передумови, що знаходитиме числа Фібоначі, що менші ніж 100 та збільшуватиме показник індексу з кожним повторенням циклу. Ділі будуємо арифметичний цикл, що перевірятиме кожне число Фібоначі на простоту та виводитиме його разом з індексом тільки за умови, що воно просте. Для переривання циклу використовуємо команду break.

# Побудова математичної моделі. Складемо таблицю імен змінних:

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Перше число ряду Фібоначі | Натуральне | fib1 | Початкове дане, проміжне дане |
| Наступне число ряду Фібоначі | Натуральне | fib2 | Початкове дане, проміжне дане |
| Сума двох попередніх чисел Фібоначі | Натуральне | fib\_sum | Початкове дане,  проміжне дане |
| Показник порядкового номера | Натуральне | i | Початкове дане, проміжне дане |
| Число, на яке будемо ділити числа Фібоначі, щоб перевіряти їх на простоту | Натуральне | j | Початкове дане, проміжне дане |

Крок 1. Задаємо початкові значення fib1, fib2, fib\_sum, i.

Крок 2. Виводимо перші три числа Фібоначі.

Крок 3. Деталізуємо цикл, що знаходить числа Фібоначі.

Крок 4. Деталізуємо цикл перевірки числа на простоту.

1. **Псевдокод:**

# Початок

1) Задаємо початкові значення fib1, fib2, fib\_sum, i

2) Виводимо перші три числа Фібоначі

3) Деталізуємо цикл, що знаходить числа Фібоначі

4) Деталізуємо цикл перевірки числа на простоту

**Кінець**

*Крок1*

# Початок

1) fib1:= 1; fib2:= 2; fib\_sum:= fib1+fib2; i:= 2

2) Виводимо перші три числа Фібоначі

3) Деталізуємо цикл, що знаходить числа Фібоначі

4) Деталізуємо цикл перевірки числа на простоту

**Кінець**

*Крок2*

# Початок

1) fib1:= 1; fib2:= 2; fib\_sum:= fib1+fib2; i:= 2

2) Виводимо fib1, виводимо fib2, виводимо fib\_sum;

3) Деталізуємо цикл, що знаходить числа Фібоначі

4) Деталізуємо цикл перевірки числа на простоту

**Кінець**

*Крок3*

**Початок**

1) fib1:= 1; fib2:= 2; fib\_sum:= fib1+fib2; i:= 2

2) Виводимо fib1, виводимо fib2, виводимо fib\_sum;

3) **Поки** fib\_sum <= 100

**Повторити**

fib\_sum:= fib1+fib2

fib1:= fib2

fib2:= fib\_sum

i = i + 1

4) Деталізуємо цикл перевірки числа на простоту

**Все повторити**

**Кінець**

*Крок4*

**Початок**

1) fib1:= 1; fib2:= 2; fib\_sum:= fib1+fib2; i:= 2

2) Виводимо fib1, виводимо fib2, виводимо fib\_sum;

3) **Поки** fib\_sum <= 100

**Повторити**

fib\_sum:= fib1+fib2

fib1:= fib2

fib2:= fib\_sum

i = i + 1

4) **Поки** j:= 2; j < fib1; j = j + 1

**Повторити**

**Якщо** fib1 % j == 0;

**то** break;

**інакше**

**Якщо** fib1 == j + 1;

**то** виводимо n;

**Все повторити**

**Все повторити**

**Кінець**

1. **Блок-схема**

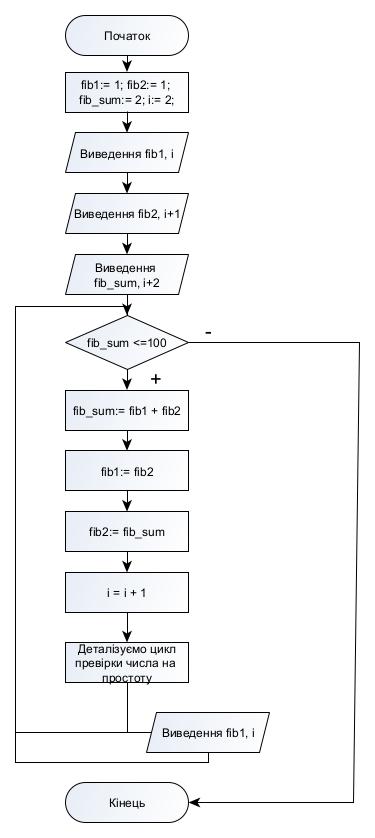
*Крок1*

|  |  |
| --- | --- |
|  |  |

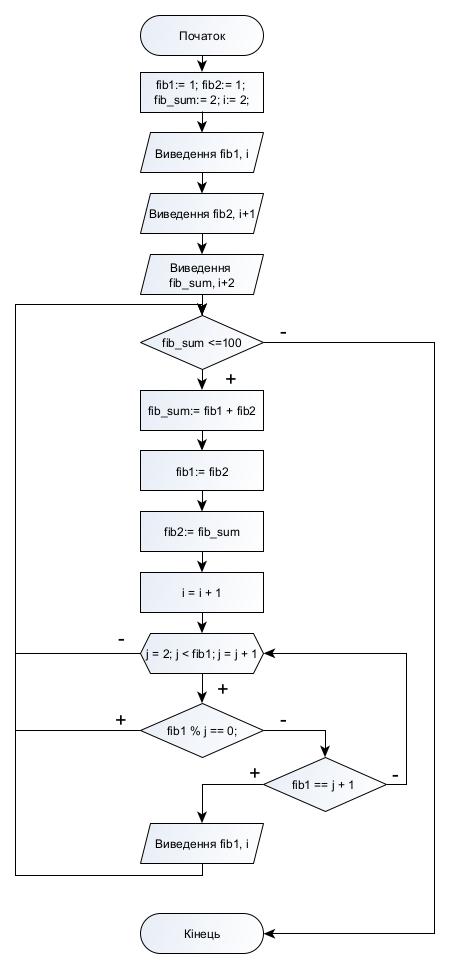
*Крок2*



*Крок3*



*Крок4*



**5. Випробування алгоритму:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Блок** | **Дія(цикл 1)** | **Дія(цикл 2)** | **Дія(цикл 3)** | **Дія(цикл 4)** |
|  | Початок |  |  |  |
| **1** | fib1:=1; fib2:=1; fib\_sum:=1+1=2; |  |  |  |
| **2** | Виведення 1, [2] |  |  |  |
| **3** | Виведення 1, [3] |  |  |  |
| **4** | Виведення 2, [4] |  |  |  |
| **5** | 3 == 2+1 | 5 == 4 + 1 | 8 % 2 == 0 | 13 == 12+1 |
| **6** | Виведення 3, [5] | Виведення 5, [6] | break | Виведення 13, [8] |
| **Блок** | **Дія(цикл 5)** | **Дія(цикл 6)** | **Дія(цикл 7)** | **Дія(цикл 8)** |
| **1** |  |  |  |  |
| **2** |  |  |  |  |
| **3** |  |  |  |  |
| **4** |  |  |  |  |
| **5** | 21 % 3 == 0 | 34 % 2 == 0 | 55 % 5 == 0 | 89 == 88 + 1 |
| **6** | break | break | break | Виведення 89, [12] |
|  |  |  |  | Кінець |

**Висновок:** Виконуючи лабораторну роботу я дослідив особливості роботи складних циклів та набув практичних навичок їх використання під час складання програмних специфікацій. Побудував алгоритм знаходження простих чисел Фібоначі від 1 до 100 включно з використанням складного циклу. В процесі випробування алгоритму було виведено такі числа та їх порядкові номери:

1[2]

1[3]

2[4]

3[5]

5[6]

13[8]

89[12]