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 10

Виконав студент ІП-14 Качмар Андрій Дмитрович

(шифр, прізвище, ім'я, по батькові)

Перевірив Мартинова Оксана Петрівна

( прізвище, ім'я, по батькові)

Київ 2021

**Лабараторна робота 5**

**Тема** – Організація циклічних процесів складні цикли

**Мета** – вивчити особливості організації складних циклів

**Завдання**: Дано натуральне число n. Знайти всі число Мерсена, що не перевищують число n. Просте число називається числом Мерсена, якщо його можна представити у вигляді де p – теж просте число.

**Розв’язання**:

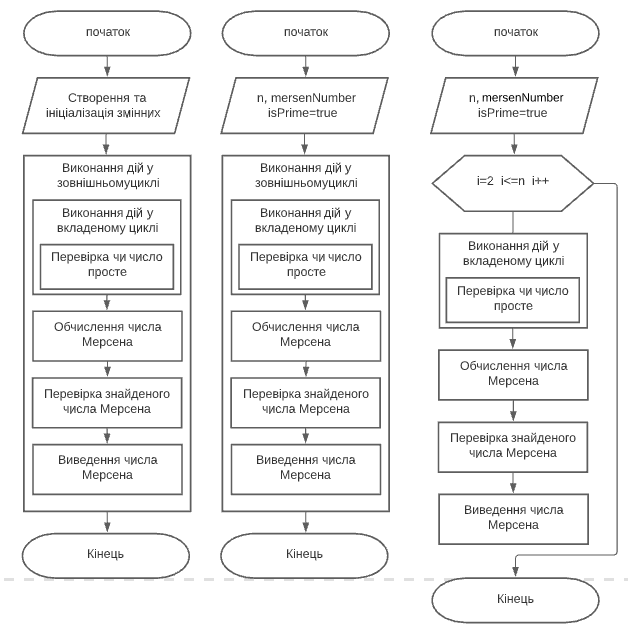
Постановка задачі: задано натуральне число n. Потрібно знайти усі числа Мерсена по формулі і вивести користувачу ті числа що не перевищують введене натуральне число n де число p теж натуральне. Для цього після введення числа n потрібно створити арифметичний цикл з i=2 по i<=n і у цьому циклі створити ще один арифметичний цикл починаючи з j=2 по j<=i. У тілі 2 циклу створимо умовний оператор умови де будемо перевіряти число просте чи ні та встановлювати відповідне значення true або false. Якщо число просте після виходу з 2 циклу створюємо оператор альтернативи де перевіряємо результат першого умовного оператора якщо істина обраховуємо число Мерсена за формулою Далі перевіряємо чи визначене число не більше за введенне число n якщо умова істина виводимо число Мерсена. Якщо умова хибна повертаємося до 1 циклу і повторюємо дії проходимо ще одну ітерацію.

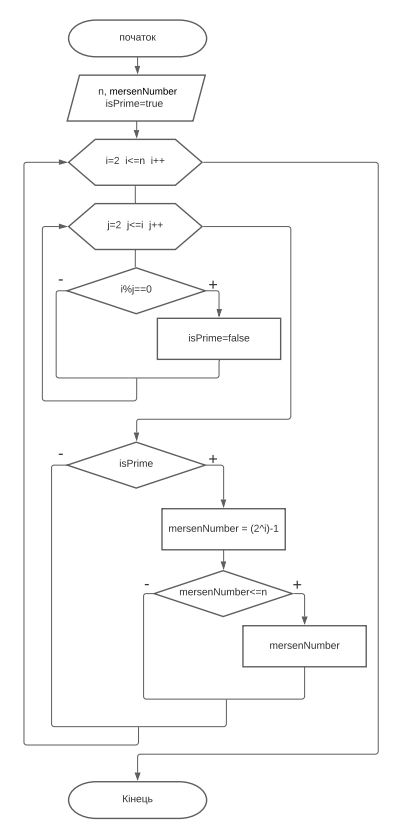
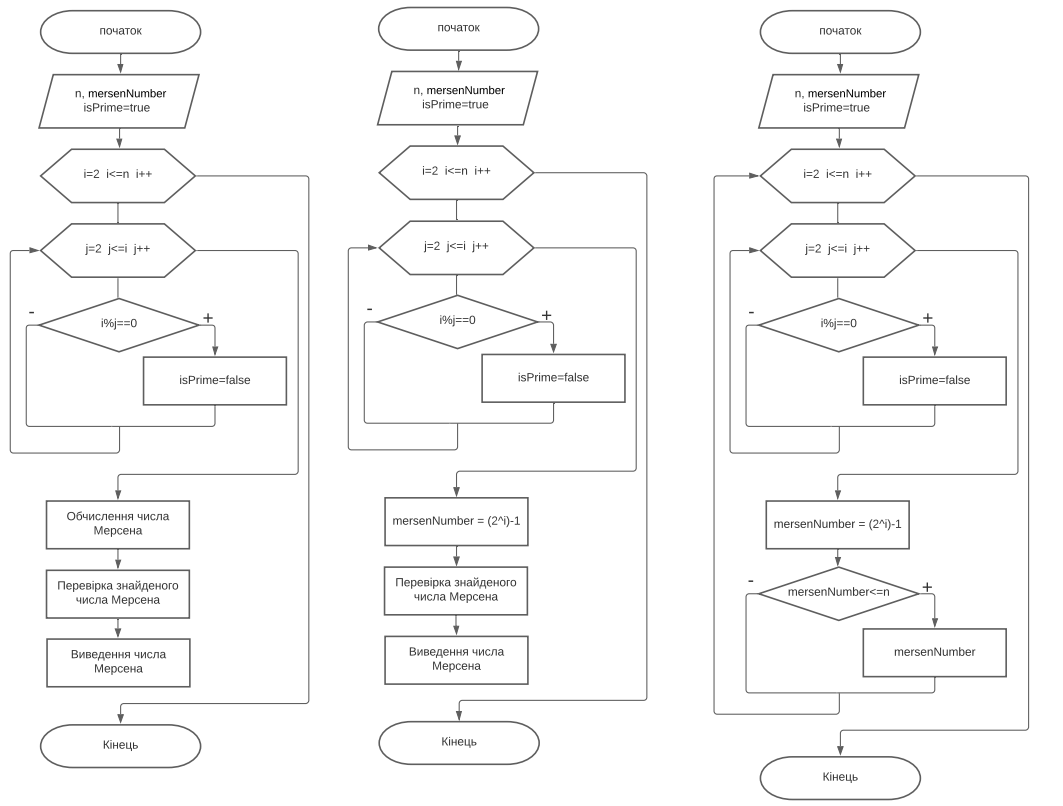
Побудова математичної моделі

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Перший параметр | int | n | Вхідне значення границя пошуку |
| Число Мерсена | int | mersenNumber | Збереження значення числа Мерсена |
| Перевірка простого числа | boolean | isPrime | Збереження результату перевірки простого числа |
| Параметр 1 циклу | int | i | Змінний параметр циклу |
| Параметр 2 циклу | int | j | Змінний параметр циклу |

|  |
| --- |
| Крок 1  **Початок**  створення та ініціалізація змінних  виконання дій у 1 циклі   1. Виконання дій у 2 циклі    * 1. Перевірка чи число просте 2. Обчислення числа Мерсена 3. Перевірка знайденого числа Мерсена 4. Виведення числа Мерсена   **кінець**  Крок 2  **Початок**  n, mersenNumber isPrime= true  виконання дій у 1 циклі   1. Виконання дій у 2 циклі    * 1. Перевірка чи число просте 2. Обчислення числа Мерсена 3. Перевірка знайденого числа Мерсена 4. Виведення числа Мерсена   **кінець**  Крок 3  **Початок**  n, mersenNumber isPrime=true  **почати** з i = 2 i<=n i++   1. Виконання дій у 2 циклі    * 1. Перевірка чи число просте 2. Обчислення числа Мерсена 3. Перевірка знайденого числа Мерсена 4. Виведення числа Мерсена   **кінець**  Крок 4  **Початок**  n, mersenNumber isPrime= true  **почати** з i = 2 i<=n i++  **почати** з j = 2 j<=i j++  **якщо** i%j==0  **то** isPrime= false   1. Обчислення числа Мерсена 2. Перевірка знайденого числа Мерсена 3. Виведення числа Мерсена   **кінець**  Крок 5  **Початок**  n, mersenNumber isPrime= true  **почати** з i = 2 i<=n i++  **почати** з j = 2 j<=i j++  **якщо** i%j==0  **то** isPrime= false  mersenNumber = (2^i)-1   1. Перевірка знайденого числа Мерсена 2. Виведення числа Мерсена   **кінець**  Крок 6  **Початок**  n, mersenNumber isPrime= true  **почати** з i = 2 i<=n i++  **почати** з j = 2 j<=i j++  **якщо** i%j==0  **то** isPrime= false  mersenNumber = (2^i)-1  **якщо** mersenNumber<=n  **то** mersenNumber  **кінець** |

**Блок-схема алгоритму**

**** Крок 1 Крок 2 Крок 3

****Крок 4 Крок 5 Крок 6

**Випробування:** Слідуючи псевдокоду тестуємо наш алгоритм

При заданому параметру n

|  |  |
| --- | --- |
| Етап | Дія |
| 1 | Введення n = 1000000 |
| 2 | Цикл з i=2 i<=n i++ |
| 3 | Цикл з j=2 j<=n j++ |
| 4 | Після виконання циклів отримуємо числа Мерсена які не перевищують число n |
| 5 | n=2  isPrime = true  А отже виконується 2 умовний оператор |
| 6 | mersenNumber = (2^I)-1 = 3  Далі виконується умова по неперевищенню введеного числа |
| 7 | Число менше введеного а отже виводимо  Знайдене число 3  Але якщо число перевищує задане то число **не** виведеться |
| 8 | Виконуємо дані дії на інших ітераціях і отримуємо послідовність |
| 9 | 3, 7, 31, 127, 2047, 8191, 131071, 524287 |

**Висновок:** Виконавши дану лабораторну роботу було написано алгоритм для знаходження усіх чисел Мерсена , ,……,,, що неперевищують введеного користувачем натурального числа n = 3,4…… та обраховуються за формулою де p просте число. Також було вивчено та застосовано арифметичні вкладені цикли з параметром. Було досліджено побудову арифметичних складних циклічних алгоритмів. У ході виконання побудована блок-схему та написано псевдокод. Також даний алгоритм було протестовано арифметично шляхом підстановки значень. Під час виконання даної лабораторної роботи було досліджено особливості роботи арифметичних складних циклів та набуто практичних навичок їх використання під час складання програмних специфікацій.