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 № 6 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження рекурсивних алгоритмів»

Варіант 24

Виконав студент ІП-14 Прокопенко Олексій Анатолійович

(шифр, прізвище, ім'я, по батькові)

Перевірив доц. Мартинова О. П.

(прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 6**

**Дослідження рекурсивних алгоритмів**

**Мета** – дослідити особливості роботи рекурсивних алгоритмів та набути практичних навичок їх використання під час складання програмних специфікацій підпрограм.

**Варіант 24**

**Завдання:**

Визначити перші 8 п’ятизначних паліндромів, що не є простими числами.

1. **Побудова математичної моделі:**

Необхідно поступово перебирати п’ятизначні паліндроми та шукати серед них не прості числа. Алгоритм перебору паліндромів можна реалізувати за допомогою трьох циклів з лічильниками. Число-паліндром буде обчислюватись за формулою number := i \* 10000 + j \* 1000 + y \* 100 + j \* 10 + i, де i, j та y ˗ лічильники, а простоту числа перевіряємо рекурсивною підпрограмою, яка буде шукати остачу від ділення паліндрома на числа від 2 до ceil(sqrt(number)) та прирівнюванням неї до 0, викликаючи саму себе повторно.

**Розв’язання:**

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

Крок 1. Визначення основних дій;

Крок 2. Деталізація дії перебору паліндромів;

Крок 3. Деталізація дії визначення простоти числа за допомогою рекурсії;

**Математична модель:**

|  |  |  |  |
| --- | --- | --- | --- |
| Опис | Тип | Ім’я | Призначення |
| Лічильник 1 | int | i | Проміжні дані |
| Лічильник 2 | int | j | Проміжні дані |
| Лічильник 3 | int | y | Проміжні дані |
| Лічильник 4 | int | u | Проміжні дані |
| Кількість виведених чисел | int | out | Проміжні дані |
| Простота числа | bool | isprime | Проміжні дані |
| П'ятизначний паліндром | int | number | Вихідні дані |

**Псевдокод:**

*Крок 1.*

**Підпрограма**

**prime (number, u):**

Визначення простоти числа за допомогою рекурсії

**все підпрограма початок**

Перебір паліндромів

**кінець**

*Крок 2.*

**функція prime (number, u):**

Визначення простоти числа за допомогою рекурсії

**все функція початок**

out := 0

**повторити для** i від 1 до 9

**повторити для** j від 0 до 9

**повторити для** y від 0 до 9

number := i \* 10000 + j \* 1000 + y \* 100 + j \* 10 + i isprime := True

**якщо** out < 8

**то**

isprime := prime(number, 2)

**якщо** isprime == False

**то**

out := out + 1

**виведення** number

**все якщо все якщо**

**все повторити все повторити**

**все повторити кінець**

*Крок 3.*

**Підпрограма**

**prime (number, u):**

**якщо** u < ceil(sqrt(number) + 1)

**то**

**якщо** number % u == 0

**то**

**повернути** False

**інакше**

u = u + 1

**повернути** prime(number, u)

**все якщо інакше**

**повернути** True

**все якщо все підпрограма**

**початок**

out := 0

**повторити для** i від 1 до 9

**повторити для** j від 0 до 9

**повторити для** y від 0 до 9

number := i \* 10000 + j \* 1000 + y \* 100 + j \* 10 + i isprime := True

**якщо** out < 8

**то**

isprime := prime(number, 2)

**все якщо**

**якщо** isprime == False

**то**

out := out + 1

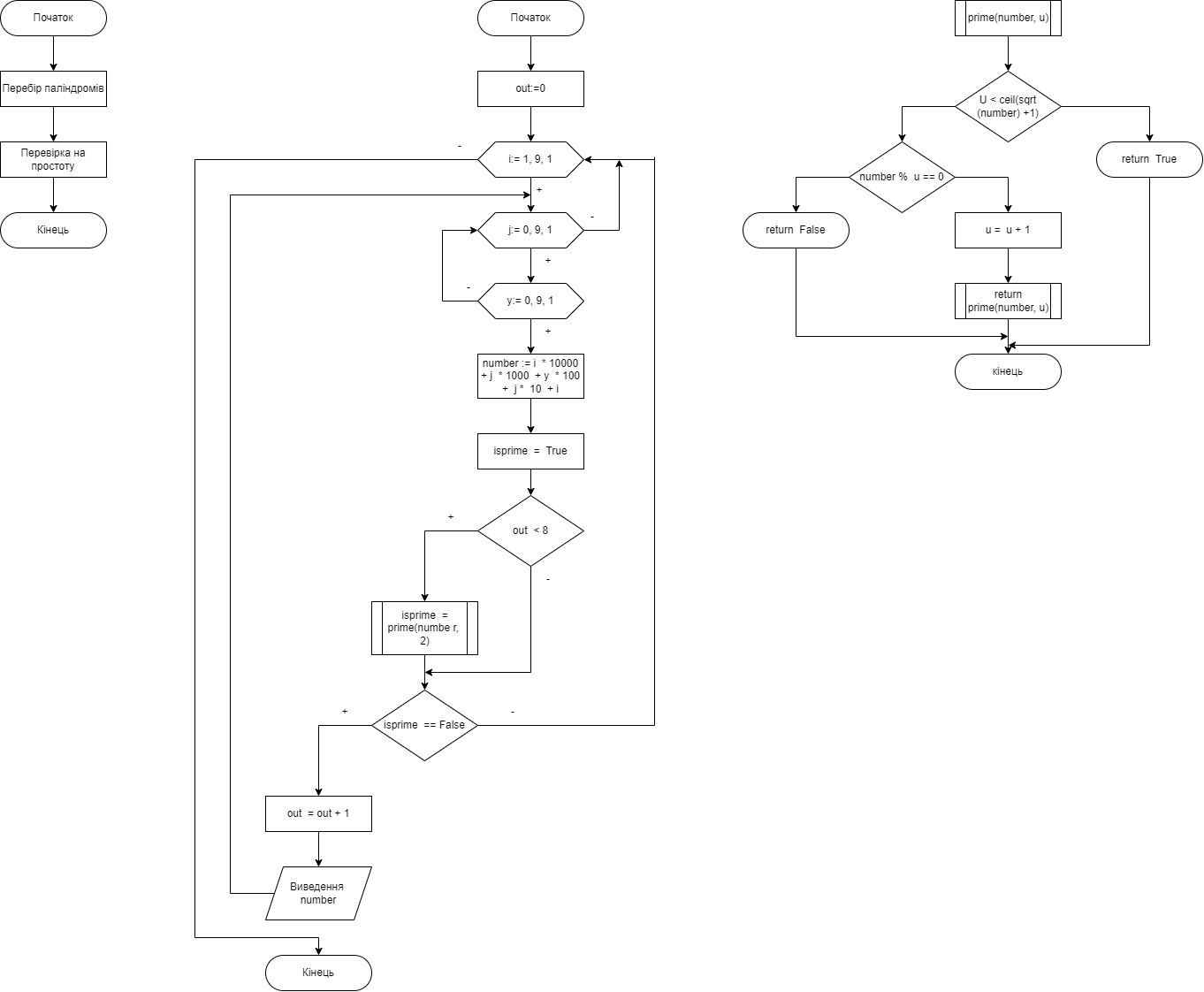
**виведення** number

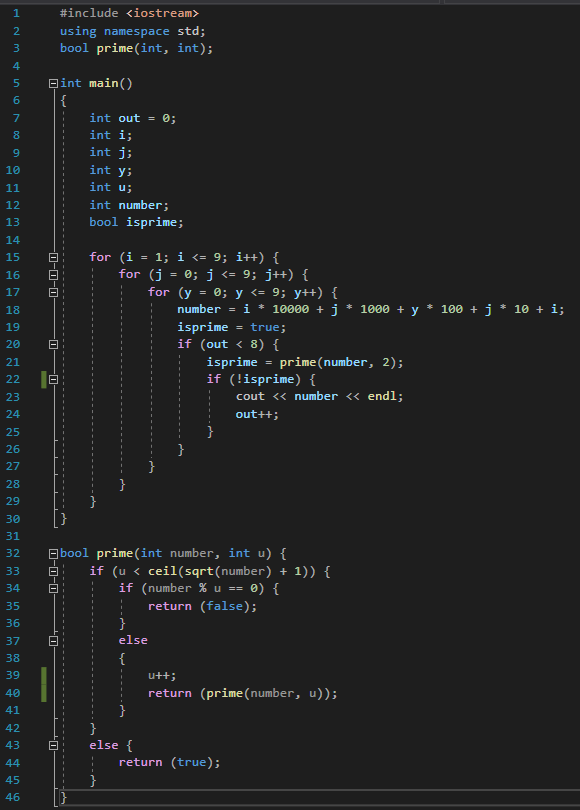
**все якщо все повторити**

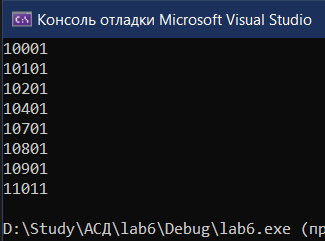
**все повторити все повторити**

**кінець**

**Блок – схема**



**Код програми на C++**



**Тестування**

|  |  |
| --- | --- |
| Блок | Дія |
|  | Початок |
| 1 | i = 1  j = 0  y = 0  number = 10001  …  u = 11  isprime = False Out = 0 + 1 = 1  Виведення number |
| 2 | … |
| 3 | i = 1  j = 0  y = 3  number = 10301  …  isprime == True |
| 4 | … |
|  | Кінець |

**Висновки:**

Я набув практичних навичок використання рекурсивних алгоритмів під час складання програмних специфікацій. В процесі виконання лаболаторної роботи було використано 3 ітераційні цикли за допомогою яких я перебирав п’ятизначні паліндроми, а також рекурсивну функцію яка повертала значення чипу “boolean”, за допомогою якої ці числа перевірялися на простоту.