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 16

Виконав студент ІП-12 Ковинєв Владислав Андрійович

Перевірив \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(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.

Задача. Дано натуральне число n. Треба знайти всі прості дільники

**I.** Постановка задачі. Розв`язати задачу поможе арифметичний цикл, котрий буде підбирати найменший дільник для числа та ділити на нього, а потім таку ж дію повторити з остачею, поки остача не буде простим числом.

**II.** Побудова математичної моделі. Складемо таблицю імен даних



Таким чином, математичне формулювання завдання зводиться до рекурсивного ділення числа на прості дільники до простої остачі .

*III. Розв’язання*

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

*Крок 1. Визначимо основні дії*

*Крок 2. Деталізуємо дію вводу значень*

*Крок 3. Деталізуємо дію знаходження дільника*

*Крок 4. Деталізуємо дію перевірки дільника*

*Крок 5. Деталізуємо дію виводу цього дільника*

*Крок 6. Деталізуємо дію ділення числа на цей дільника*

*Крок 5. Деталізуємо дію збільшення дільника*

*Крок 7. Деталізуємо дію перевірки остачі числа*

*Крок 8. Деталізуємо дію виводу остачі числа*

***Псевдокод***

***крок 1***

**початок**

**Введення значень**

Знаходження дільника

Перевірка дільника

**Виведення цього дільника**

Ділення числа на цей дільник

Збільшення дільника

Перевірка остачі числа

**Виведення остачі числа**

**кінець**

***крок 2***

**початок**

**Введення n**

Знаходження дільника

Перевірка дільника

**Виведення цього дільника**

Ділення числа на цей дільник

Збільшення дільника

Перевірка остачі числа

**Виведення остачі числа**

**кінець**

***крок 3***

**початок**

**Введення n**

Для i від 2 до sqrt(n)

Перевірка дільника

**Виведення цього дільника**

Ділення числа на цей дільник

Збільшення дільника

Кінець циклу

Перевірка остачі числа

**Виведення остачі числа**

**кінець**

***крок 4***

**початок**

**Введення n**

Для i від 2 до sqrt(n)

Якщо n%і==0

**Виведення цього дільника**

Ділення числа на цей дільник

Інакше

Збільшення дільника

Все якщо

Кінець циклу

Перевірка остачі числа

**Виведення остачі числа**

**кінець**

***крок 5***

**початок**

**Введення n**

Для i від 2 до sqrt(n)

Якщо n%і==0

**Виведення i**

Ділення числа на цей дільник

Інакше

i+=1

Все якщо

Кінець циклу

Перевірка остачі числа

**Виведення остачі числа**

**кінець**

***крок 6***

**початок**

**Введення n**

Для i від 2 до sqrt(n)

Якщо n%і==0

**Виведення i**

n/=i

Інакше

i+=1

Все якщо

Кінець циклу

Перевірка остачі числа

**Виведення остачі числа**

**кінець**

***крок 7***

**початок**

**Введення n**

Для i від 2 до sqrt(n)

Якщо n%і==0

**Виведення i**

n/=i

Інакше

i+=1

Все якщо

Кінець циклу

Якщо n>1

**Виведення остачі числа**

**кінець**

***крок 7***

**початок**

**Введення n**

Для i від 2 до sqrt(n)

Якщо n%і==0

**Виведення i**

n/=i

Інакше

i+=1

Все якщо

Кінець циклу

Якщо n>1

**Виведення n**

**кінець**



















**IV.** Випробування алгоритму



**V.** Висновок

Було досліджено особливості роботи арифметичних циклів за допомогою математичних моделей, псевдокоду та графічного подання у вигляді блок-схеми, завдяки чому були набуто практичних навичок їх використання під час складання програмних специфікацій.

**VI.** Код

