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-15, Куманецька І. В.\_\_\_  
Перевірив \_Вечерковська А. С.\_\_\_\_\_\_

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.

**Індивідуальне завдання**

**Варіант 16**



**Постановка задачі**

Знайти всі прості дільники заданого натурального числа. Результатом виведення мають бути власне дільники.

**Побудова математичної моделі**

|  |  |  |  |
| --- | --- | --- | --- |
| **Змінна** | **Тип** | **Ім'я** | **Призначення** |
| Задане натуральне число | Ціле натуральне | chislo | Вхідні дані |
| Число, для якого виводимо прості дільники | Ціле натуральне | number | Проміжні дані, аргумент функції delim |
| Можливий простий дільник | Ціле натуральне | divider | Проміжні дані, аргумент функції delim |

Розрахування та вивід простих дільників виконується за допомогою функції delim(number, divider). Термінальна гілка виконується, якщо саме число дорівнює 1, тобто простих дільників немає. У рекурсивній гілці знаходиться можливий простий дільник, після чого число ділиться на максимальний степінь отриманого дільника. Потім рекурсивно запускається ця ж функція від вже зменшеного числа та наступного можливого значення дільника.

Через sqrt() позначається знаходження квадратного кореня від числа.  
Через a /= b позначається операція a = а / b. Через % позначається операція  
знаходження остачі від ділення. Через != позначається «не дорівнює.»

**Розв’язання**

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

Крок 1. Визначення основних дій.

Крок 2. Деталізація розрахування та виведення простих дільників числа.

**Псевдокод**

*Крок 1*

**початок**

введення chislo

розрахування та виведення простих дільників chislo

**кінець**

*Крок 2*

**початок**

введення chislo

delim(chislo, 2)

**кінець**

*Підпрограма*

**delim(number, divider)**

**якщо** number = 1

**то** виведення «Більше простих дільників немає»

**інакше**

**поки** number % divider != 0

**повторити**

divider += 1

**все повторити**

виведення divider

**повторити**

number /= divider

**поки** number % divider = 0

**все повторити**

delim(number, divider + 1)

**кінець**

**Блок-схема**

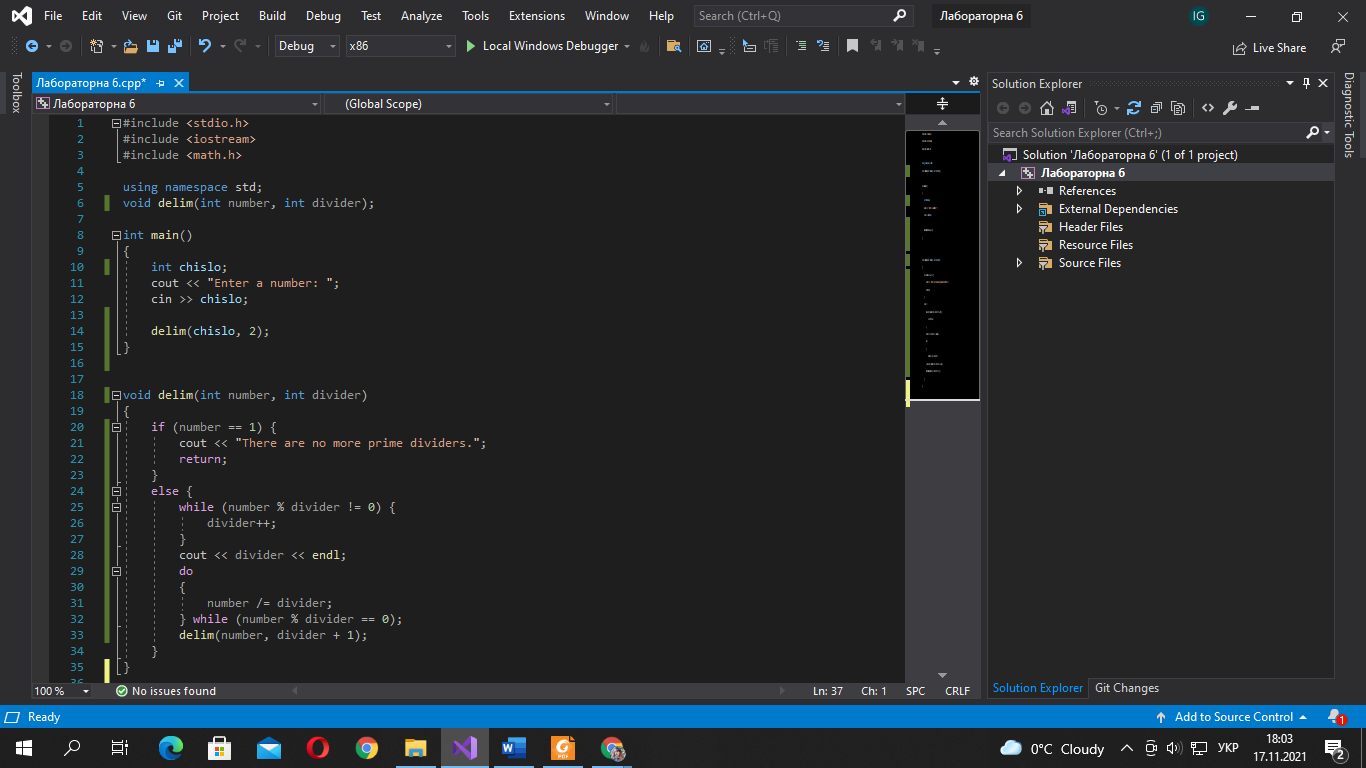
*Крок 1 Крок 2*

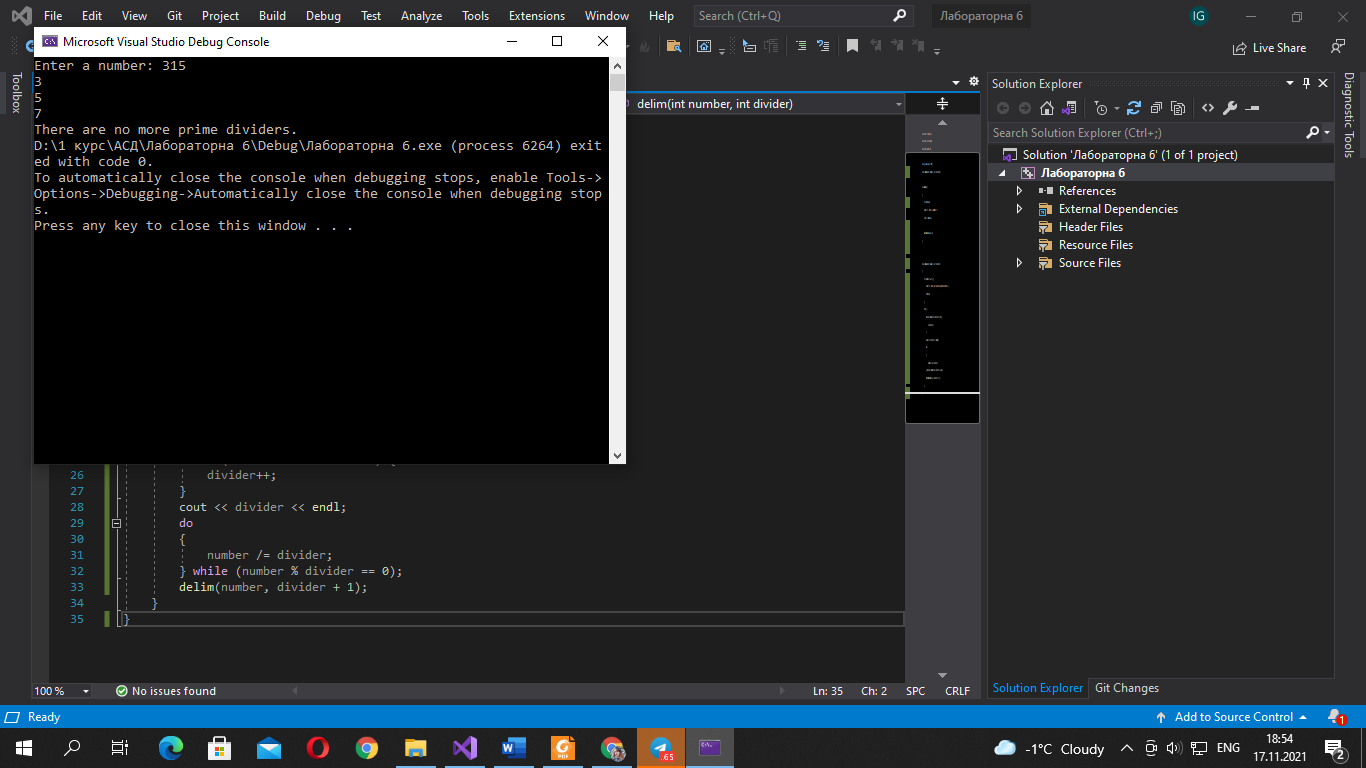


*Підпрограма*



**Код**





**Випробування**

|  |  |
| --- | --- |
| **Блок** | **Дія** |
| Початок |  |
| 1 | введення chislo := 315 |
|  | delim(315, 2) |
| 2 | 315 = 1 False |
| 3 (перший цикл 1) | 315 % 2 = !0 True  divider := 3 |
| 4 (перший цикл 2) | 315 % 3 != 0 False |
| 5 | виведення 3 |
| 6 (другий цикл 1) | chislo := 105  105 % 3 = 0 True |
| 7 (другий цикл 2) | chislo := 35  35 % 3 = 0 False |
|  | delim(35, 4) |
| 8 | 35 = 1 False |
| 9 (перший цикл 1) | 35 % 4 = !0 True  divider := 5 |
| 10 (перший цикл 2) | 315 % 5 != 0 False |
| 11 | виведення 5 |
| 12 (другий цикл 1) | chislo := 7  7 % 5 = 0 False |
|  | delim(7, 6) |
| 13 | 7 = 1 False |
| 14 (перший цикл 1) | 7 % 6 = !0 True  divider := 7 |
| 15 (перший цикл 2) | 7 % 7 != 0 False |
| 16 | виведення 7 |
| 17 (другий цикл 1) | chislo := 1  1 % 7 = 0 False |
|  | delim(1, 8) |
| 18 | 1 = 1 True |
| 19 | виведення «Більше простих дільників немає» |

**Висновок**

Було досліджено особливості роботи рекурсивних алгоритмів та набути практичних навичок їх використання під час складання програмних специфікацій підпрограм. В результаті виконання лабораторної роботи було знайдено прості  
дільники числа, розділивши задачу на 2 кроки: визначення основних дій, деталізація розрахування та виведення простих дільників числа. В процесі  
випробування було розраховано прості дільники числа 315: 3, 5, 7.