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 22

Виконав студент \_\_\_\_\_\_\_\_Мєшков\_Андрій\_Ігорович\_\_\_\_\_\_

(шифр, прізвище, ім'я, по батькові)

Перевірив \_\_\_\_\_\_\_\_\_Вєчерковська Анастасія Сергіївна\_\_\_\_\_

( 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.

**Варіант 22**

*Задача.* Натуральне число називається паліндромом, якщо його запис читається однаково з початку та з кінця (наприклад, 575, 9). Знайти всі паліндроми з інтервалу [1000000,1000000000].

**Постанова задачі.** Для знаходження паліндрому ми будемо використовувати два цикли: зовнішній арифметичний цикл для перерахування кожного числа проміжку та цикл з передумовою для перетворення числа на обернене для подальшого порівняння. Для отримання оберненого числа ми будемо використовувати прості арифметичні дії та остачу від ділення числа.

|  |  |  |  |
| --- | --- | --- | --- |
| Змінна | Тип | Ім’я | Призначення |
| Натуральне число в проміжку, паліндром, елемент зовнішнього циклу | Цілий/Натуральний | a | Початкові дані, результат |
| Елемент внутрішнього циклу | Цілий/Натуральний | b | Проміжні дані |
| Обернене число a | Цілий/Натуральний | z | Проміжні дані |

Кожне значення a ми будемо привласнювати b. Поки b>0 ми будемо підсумовувати остачу від ділення(%) на 10 у z та ділити b на 10 націло. Після виходу з внутрішнього циклу ми будемо порівнювати z та a. Якщо число дорівнює оберненому, то число є паліндромом та виводиться на екран. Після перевіряється наступне число з проміжку.

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

*Крок 1.* Визначимо основні дії.

*Крок 2.* Деталізуємо дію зовнішнього циклу: знаходження та виведення паліндромів.

*Крок 3.* Деталізуємо дію внутрішнього циклу: знаходження оберненого числа.

*Крок 4.* Деталізуємо дію порівняння натурального числа з оберненим та виведення паліндромів.

**Псевдокод**

*Крок 2*

**Початок**

**повторити**

**для a від 1000000 до 1000000000**

b:=a

z:=0

Внутрішній цикл, знаходження оберненого числа

Порівняння натурального числа з оберненим та виведення паліндромів

**все повторити**

**Кінець**

*Крок 3*

**Початок**

**повторити**

**для a від 1000000 до 1000000000**

b:=a

z:=0

**поки b>0**

**повторити**

z:=z\*10

z:=z+b%10

b:=b/10

**все повторити**

Порівняння натурального числа з оберненим та виведення паліндромів

**все повторити**

**Кінець**

*Крок 4*

**Початок**

**повторити**

**для a від 1000000 до 1000000000**

b:=a

z:=0

**поки b>0**

**повторити**

z:=z\*10

z:=z+b%10

b:=b/10

**все повторити**

**якщо** a==z

**то**

Вивести a

**все якщо**

**все повторити**

**Кінець**

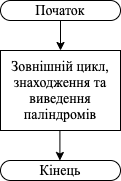
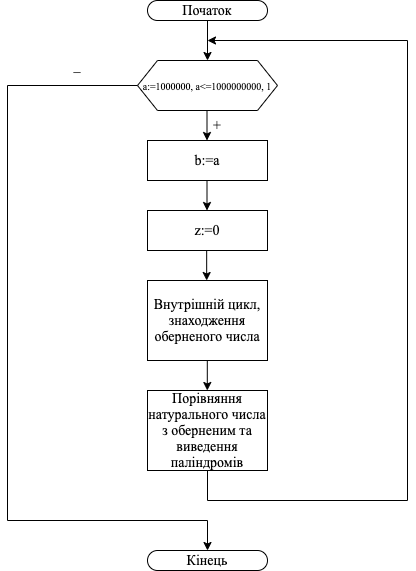
*Крок 1*

**Початок**

Зовнішній цикл, знаходження та виведення паліндромів

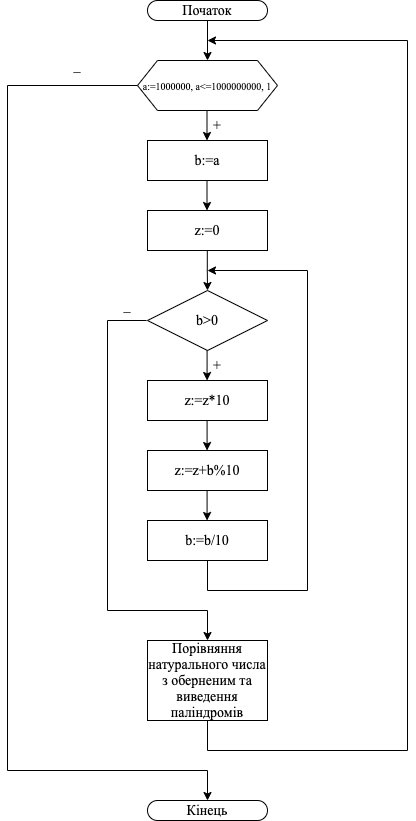
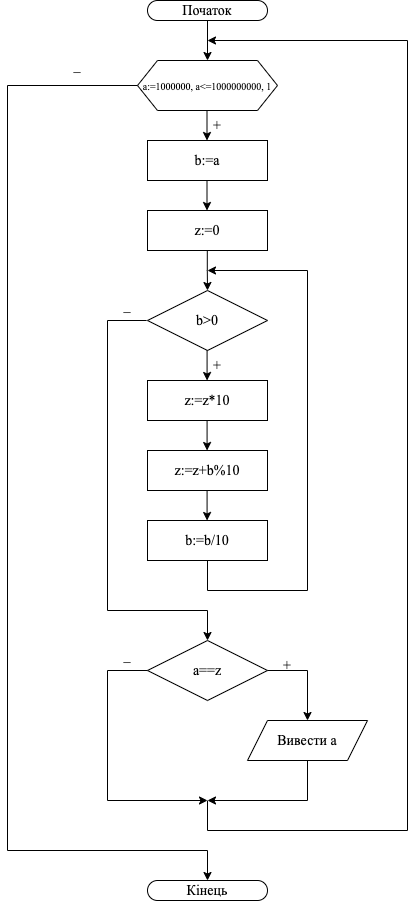
**Кінець**

**Блок-схема**

**

*Крок 2*

*Крок 1*

**

*Крок 4*

*Крок 3*

**Випробування алгоритму:** перевіримо правильність алгоритму на довільних конкретних значеннях початкових даних.

**Тест№1**

|  |  |
| --- | --- |
| Блок | Дія |
| 1 | **Початок** |
| 2 | **Початок арифм. циклу. a=1000000; a<=1000000000; a++** |
| 3 | b=1000000, z=0 |
| 4 | Початок циклу. Перевірка умови: 1000000>0 – **true** |
| 5 | z=0, z=0, b=100000 |
| 6 | Початок циклу. Перевірка умови: 100000>0 – **true** |
| 7 | z=0, z=0, b=10000 |
| 8 | Початок циклу. Перевірка умови: 10000>0 – **true** |
| 9 | z=0, z=0, b=1000 |
| 10 | Початок циклу. Перевірка умови: 1000>0 – **true** |
| 11 | z=0, z=0, b=100 |
| 12 | Початок циклу. Перевірка умови: 100>0 – **true** |
| 13 | z=0, z=0, b=10 |
| 14 | Початок циклу. Перевірка умови: 10>0 – **true** |
| 15 | z=0, z=0, b=1 |
| 16 | Початок циклу. Перевірка умови: 1>0 – **true** |
| 17 | z=0, z=1, b=0 |
| 18 | Початок циклу. Перевірка умови: 0>0 – **false** |
| 19 | 1000000==1 – **false** |
| 20 | b=1000001, z=0 |
| 21 | Початок циклу. Перевірка умови: 1000001>0 – **true** |
| 22 | z=0, z=1, b=100000 |
| 23 | Початок циклу. Перевірка умови: 100000>0 – **true** |
| 24 | z=10, z=10, b=10000 |
| 25 | Початок циклу. Перевірка умови: 10000>0 – **true** |
| 26 | z=100, z=100, b=1000 |
| 27 | Початок циклу. Перевірка умови: 1000>0 – **true** |
| 28 | z=1000, z=1000, b=100 |
| 29 | Початок циклу. Перевірка умови: 100>0 – **true** |
| 30 | z=10000, z=10000, b=10 |
| 31 | Початок циклу. Перевірка умови: 10>0 – **true** |
| 32 | z=100000, z=100000, b=1 |
| 33 | Початок циклу. Перевірка умови: 1>0 – **true** |
| 34 | z=1000000, z=1000001, b=0 |
| 35 | Початок циклу. Перевірка умови: 0>0 – **false** |
| 36 | 1000001==1000001 – **true** |
| 37 | Вивід: a=1000001 |
|  |  |
| … | Вивід: a=1001001 |
|  |  |
| … | Вивід: а=354757453 |
|  |  |
| … | Вивід: а=999999999 |
| … | b=1000000000, z=0 |
| … | Початок циклу. Перевірка умови: 1000000000>0 – **true** |
| … | z=0, z=0, b=100000000 |
| … | Початок циклу. Перевірка умови: 100000000>0 – **true** |
| … | z=0, z=0, b=10000000 |
| … | Початок циклу. Перевірка умови: 10000000>0 – **true** |
| … | z=0, z=0, b=1000000 |
| … | Початок циклу. Перевірка умови: 1000000>0 – **true** |
| … | z=0, z=0, b=100000 |
| … | Початок циклу. Перевірка умови: 100000>0 – **true** |
| … | z=0, z=0, b=10000 |
| … | Початок циклу. Перевірка умови: 10000>0 – **true** |
| … | z=0, z=0, b=1000 |
| … | Початок циклу. Перевірка умови: 1000>0 – **true** |
| … | z=0, z=0, b=100 |
| … | Початок циклу. Перевірка умови: 100>0 – **false** |
| … | z=0, z=0, b=10 |
| … | Початок циклу. Перевірка умови: 10>0 – **true** |
| … | z=0, z=0, b=1 |
| … | Початок циклу. Перевірка умови: 1>0 – **true** |
| … | z=0, z=1, b=0 |
| … | Початок циклу. Перевірка умови: 0>0 – **false** |
| … | 1000000000==1 – **false** |
| … | **Вихід з арифм. циклу** |
| … | **Кінець** |

**Висновок:** булодосліджено складні циклічні алгоритми, проаналізовано подане завдання, декомпозовано та виконано. Також були розроблені псевдокод та блок-схема поставленого алгоритму.