# 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 № 8 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження алгоритмів перетворення матриць

та отримання агрегатних значень»

Варіант 19

Виконав студент ІП-12 Лисенко Олександр Олександрович

(шифр, прізвище, ім'я, по батькові)

Перевірив

( прізвище, ім'я, по батькові)

Київ 2021

**Лабораторна робота 8**

**Дослідження алгоритмів перетворення матриць та отримання агрегатних значень**

**Мета** – дослідити підходи до пошуку та перетворення на матрицях та

набути практичних навичок використання укладених керувальних дій

повторення і їх з’єднання під час складання програмних специфікацій.

**Завдання:**

****

**Постановка задачі**

Створюються два масиви: один пустий, інший – заповнюється випадковими числами. Далі йде визначення двух функцій: знаходження добутку елементів стовпця і методу Шела. Потім виводиться перший масив, елементи другого масиву і результат сортування методом Шела.

**Побудова математичної моделі**

|  |  |  |  |
| --- | --- | --- | --- |
| Ім’я | Тип | Змінна | Призначення |
| a | Масив | Двовимірний масив | Початкове дане |
| b | Масив | Двовимірний масив | Результат |
| column | Цілий | Номер стовпця | Проміжне дане |
| i | Цілий | Лічильник циклу | Лічильник |
| j | Цілий | Лічильник циклу | Лічильник |

|  |  |  |  |
| --- | --- | --- | --- |
| Ім’я | Тип | Змінна | Призначення |
| total | Дійсний | Добуток | Результат |
| m | Масив | Масив | Початкове дане |
| column | Цілий | Лічильник циклу | Лічильник |
| row | Цілий | Лічильник циклу | Лічильник |

|  |  |  |  |
| --- | --- | --- | --- |
| Ім’я | Тип | Змінна | Призначення |
| data | Масив | Масив | Результат |
| gap | Цілий |  | Проміжне дане |
| i | Цілий | Лічильник циклу | Проміжне дане |
| current | Дійсний |  | Проміжне дане |
| pos | Цілий |  | Проміжне дане |

**Псевдокод**

a = [[Випадкове число на проміжку(-50, 50) **для** j **на проміжку** [0,8)] **для** і **на проміжку** [0,4)]

b = []

**Визначити** prodColumn(m, column)

total = 1

**Для row на проміжку [0, len(m))**

**Повторити**

total = total \* m[row][column]

**Все повторити**

**Повернути** total

**Визначити** shell(data)

gap = len(data) // 2

**Поки gap > 0**

**Повторити**

**Для i на проміжку [gap, len(data))**

**Повторити**

current = data[i]

pos = i

**Все повторити**

**Поки pos >= і i data[pos - gap] > current**

**Повторити**

data[pos] = data[pos - gap]

pos = pos - gap

data[pos] = current

gap //= 2

**Все повторити**

**Повернути** data

Вивід a

**Для column на проміжку [0,8)**

**Повторити**

Вивід prodColumn(a, column)

b.**додати**(prodColumn(a, column))

**Все повторити**

Вивід shell(b)

**Блок-схема**



**Код програми**

import random

a = [[random.uniform(-50, 50) for j in range(8)] for i in range(4)]

b = []

def prodColumn(m, column):

total = 1

for row in range(len(m)):

total = total \* m[row][column]

return total

def shell(data):

gap = len(data) // 2

while gap > 0:

for i in range(gap, len(data)):

current = data[i]

pos = i

while pos >= gap and data[pos - gap] > current:

data[pos] = data[pos - gap]

pos = pos - gap

data[pos] = current

gap //= 2

return data

print(a)

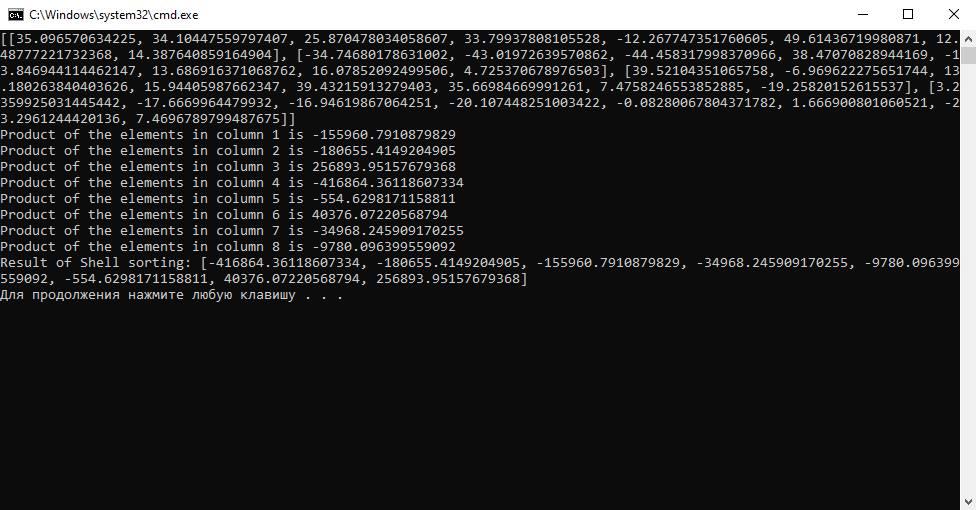
for column in range(8):

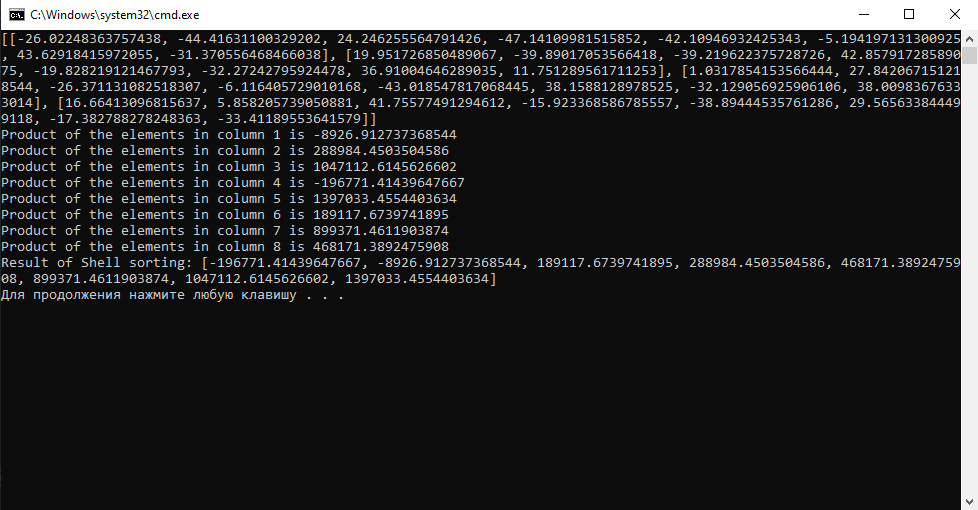
print("Product of the elements in column", column + 1, "is", prodColumn(a, column))

b.append(prodColumn(a, column))

print("Result of Shell sorting:", shell(b))

**Випробування програми**

****

****

**Висновок**

На цій лабораторній роботі було досліджено підходи до пошуку та перетворення на матрицях та було набуто практичних навичок використання укладених керувальних дій повторення і їх з’єднання під час складання програмних специфікацій.