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 № 1 з дисципліни

«Проєктування алгоритмів»

«Проектування і аналіз алгоритмів зовнішнього сортування»

Варіант 3

Виконав студент ІП-14 Гайдучек Максим Андрійович

(шифр, прізвище, ім'я, по батькові)

Перевірив Соколовський Владислав Володимирович

( прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота №1**

**Тема роботи:** Проектування і аналіз алгоритмів зовнішнього сортування

**Мета роботи:** вивчити основні алгоритми зовнішнього сортування та способи їх модифікації, оцінити поріг їх ефективності.

**Варіант:** 3

**Алгоритм**: Збалансоване багатошляхове злиття

**Хід роботи**

* 1. Псевдокод звичайного алгоритму

splitFile:

reader = створити\_зчитувач\_з\_файлу(**ШЛЯХ\_ФАЙЛУ**)

writers = створити\_список\_записувачів\_до\_файлу(**ШЛЯХ\_ФАЙЛІВ\_В**)

writerFileCount = 0

number = reader.read()

previous = 0

sequence = [number]

**ПОКИ** reader.hasNext:

writers[writerFileCount].write(reader.getLine)

number = reader.read()

**ЯКЩО** sequence.length() > 1 **ТА** previous > number

writers[writerFileCount].writeAll(sequence)

sequence = []

writerFileCount = (writerFileCount + 1) % writers.length()

**КІНЕЦЬ ЯКЩО**

sequence.add(number)

Previous = number

**КІНЕЦЬ ПОКИ**

merge:

flag = true

result = NULL

**РОБИТИ**:

outputPath = **ЯКЩО** flag **ТО** **ШЛЯХ\_ФАЙЛІВ\_В** **ІНАКШЕ** **ШЛЯХ\_ФАЙЛІВ\_С**

inputPath = **ЯКЩО** flag **ТО** **ШЛЯХ\_ФАЙЛІВ\_С** **ІНАКШЕ** **ШЛЯХ\_ФАЙЛІВ\_В**

readers = створити\_список\_записувачів\_до\_файлу (outputPath)

writeFilesCount = 0

endedReadersCount = 0

elements = []

**ПОКИ** endedReadersCount < readers.length()

min = MIN\_VALUE

minIndex = NULL

**ВІД** 0 **ДО** numberOfFilesB - 1 **У** i

number = readers[i].read()

**ЯКЩО** number >= elements[-1] **ТА** number < min

min = number

minIndex = i

**КІНЕЦЬ ЯКЩО**

**КІНЕЦЬ ВІД-ДО**

**ЯКЩО** minIndex == NULL

записати\_в\_файл\_все(inputPath + і, elements)

elements = []

writerFileCount = (writerFileCount + 1) % writers.length()

**ІНАКШЕ**

elements.add(minElement)

**КІНЕЦЬ ЯКЩО**

**КІНЕЦЬ ПОКИ**

flag = !flag

result = отримати\_останній\_непустий\_файл(inputPath)

**ПОКИ** result == NULL

* 1. Псевдокод оптимізованого алгоритму

splitFiles:

reader = створити\_зчитувач\_з\_файлу(**ШЛЯХ\_ФАЙЛУ**)

writers = створити\_список\_записувачів\_до\_файлу(**ШЛЯХ\_ФАЙЛІВ\_В**)

writerFileCount = 0

**ПОКИ** reader.hasNext:

writers[writerFileCount].write(reader.getLine)

writerFileCount = (writerFileCount + 1) % writers.length()

**КІНЕЦЬ ПОКИ**

sortFiles:

**ВІД** 0 **ДО** numberOfFilesB - 1 **У** i

reader = створити\_зчитувач\_з\_файлу(**ШЛЯХ\_ФАЙЛІВ\_В** + i)

сортувати(reader.readAllLines())

**КІНЕЦЬ ВІД**

merge:

readers = створити\_список\_записувачів\_до\_файлу (**ШЛЯХ\_ФАЙЛІВ\_В**)

writer = створити \_записувач\_до\_файлу(**ШЛЯХ\_ФАЙЛУ\_C**)

endedReadersCount = 0

**ПОКИ** endedReadersCount < readers.length()

min = MIN\_VALUE

minIndex = NULL

**ВІД** 0 **ДО** numberOfFilesB - 1 **У** i

writers[writerFileCount].write(reader.getLine)

writerFileCount = (writerFileCount + 1) % writers.length()

number = readers[i].read()

**ЯКЩО** number < min

min = number

minIndex = i

**КІНЕЦЬ ЯКЩО**

**КІНЕЦЬ ПОКИ**

**ЯКЩО** minIndex != NULL

writer.write(min)

**КІНЕЦЬ ЯКЩО**

**КІНЕЦЬ ПОКИ**

* 1. Код звичайного алгоритму

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание

* 1. Код оптимізованого алгоритму

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание Изображение выглядит как текст

Автоматически созданное описание

**Висновок**: під час виконання даної лабораторної роботи було розроблено алгоритм збалансованого багатошляхового злиття (K-Ways MergeSort). Розроблено звичайний алгоритм, потім прооптимізовано. Звичайний алгоритм справляється з файлами малих об’ємів, а вже оптимізованіший – з набагато більшими.