**Міністерство освіти і науки України**

**Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"**

**Факультет інформатики та обчислювальної техніки Кафедра ІПІ**

Звіт

з лабораторної роботи № 5 з дисципліни

«Алгоритми та структури даних 2. Структури даних»

**“Метод швидкого сортування”**

**Виконав(ла)**

Скрипець Ольга Олександрівна

(шифр, прізвище, ім'я, по батькові)

**Перевірив**

(прізвище, ім'я, по батькові)

Київ 2023

**Постановка завдання:** В даній роботі необхідно розв'язати наступну задачу визначення послідовності медіан для заданого вхідного масиву. Нагадаємо, що медіаною для масиву називається елемент, який займає середнє положення у відсортованому масиві. Так, якщо кількість елементів у масиві непарна, то медіана одна та індекс її у відсортованому масиві визначається як [n/2] (де n — розмір вхідного масиву). Якщо кількість елементів у масиві парна, то медіан буде дві та їх індекси визначаються за формулами [n/2] та [n/2] + 1.

Задача формулюється наступним чином. Нехай заданий вхідний масив A = [x1, ..., xN]. Припустимо, що елементи масиву поступають на вхід програми послідовно: в кожний момент часу розглядається новий елемент xi. Необхідно для кожного i (від 1 до N) визначити медіану підмасиву A' = [x1, ..., xi], тобто медіану для масиву елементів, які були отримані програмою на даний момент часу. Необхідно розв’язати цю задачу, використовуючі структури даних пірамід і так, щоб кожна медіана визначалась за час O(log(i)).

**Вихідний код:**

import heapq

def find\_medians(arr):

n = len(arr)

Hlow = []

Hhigh = []

medians = []

for i in range(n):

x = arr[i]

if not Hlow or x <= -Hlow[0]:

heapq.heappush(Hlow, -x)

else:

heapq.heappush(Hhigh, x)

if len(Hlow) - len(Hhigh) > 1:

heapq.heappush(Hhigh, -heapq.heappop(Hlow))

elif len(Hhigh) - len(Hlow) > 1:

heapq.heappush(Hlow, -heapq.heappop(Hhigh))

if i % 2 == 0:

if len(Hlow) > len(Hhigh):

medians.append(-Hlow[0])

else:

medians.append(Hhigh[0])

else:

medians.append((-Hlow[0], Hhigh[0]))

return medians

with open('input\_02\_10.txt', 'r') as file:

lines = file.readlines()

n = int(lines[0])

arr = [int(lines[i]) for i in range(1, n+1)]

medians = find\_medians(arr)

with open('output.txt', 'w') as file:

for median in medians:

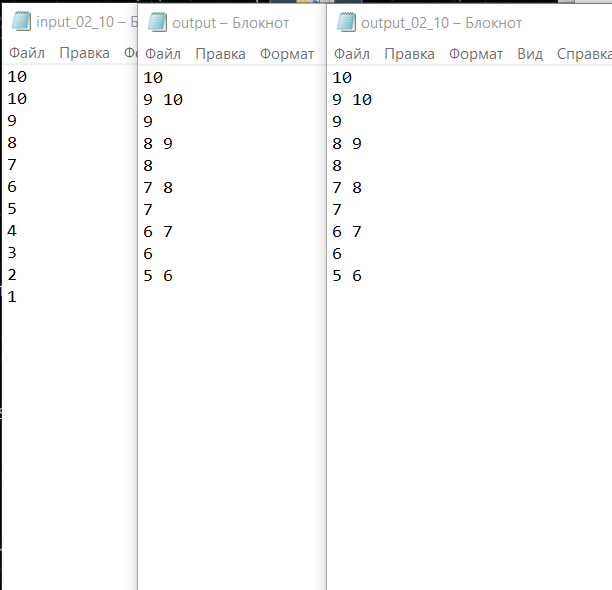
if isinstance(median, int):

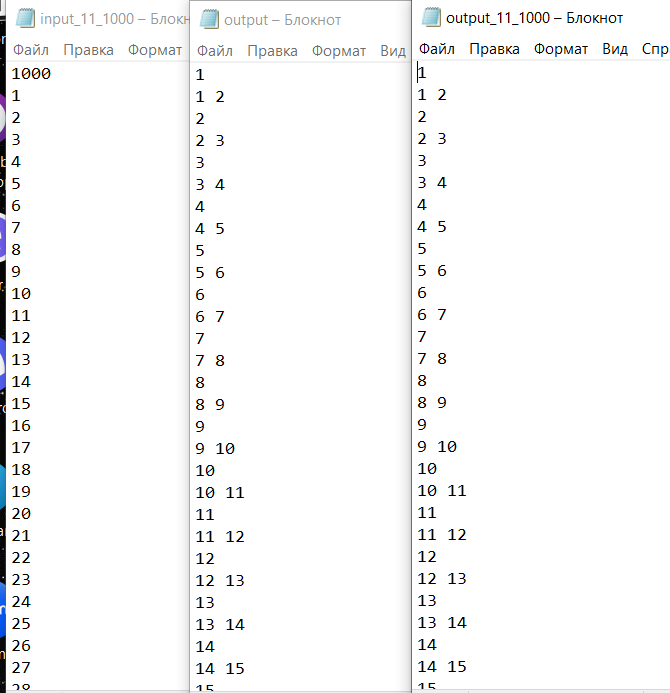
file.write(str(median) + '\n')

else:

file.write(' '.join(str(num) for num in median) + '\n')

**Приклад роботи :**

****

****