МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

Факультет електроніки та комп’ютерних технологій

**Звіт**

Про виконання лабораторної роботи №6

з дисципліни «Алгоритми та структури даних»

На тему «Швидке сортування»

**Виконав**:

Малярчук О.В. ФЕП-24

**Перевірив**:

**Львів 2022**

**Мета роботи**: ознайомитись з алгоритмом сортування, зрозуміти його реалізацію, та навчитись реалізовувати його на довільній мові програмування.

**Програмне забезпечення**: PyCharm.

**Теоретичні відомості:**

Швидке сортування (англ. Quick Sort) — алгоритм сортування, розроблений Тоні Гоаром, який не потребує додаткової пам'яті і виконує у середньому

O(n logn)

{\displaystyle \;O(n\log \;n)} операцій. Однак, у найгіршому випадку робить

O

(

n

2

)

O(n^{2}) порівнянь. Позаяк алгоритм використовує дуже прості цикли і операції, він працює швидше за інші алгоритми, що мають таку ж асимптотичну оцінку складності. Наприклад, зазвичай більш ніж удвічі швидший порівняно з сортуванням злиттям.

Ідея алгоритму полягає в переставлянні елементів масиву таким чином, щоб його можна було розділити на дві частини і кожний елемент з першої частини був не більший за будь-який елемент з другої. Впорядкування кожної з частин відбувається рекурсивно. Алгоритм швидкого сортування може бути реалізований як у масиві, так і в двозв'язному списку.

Швидке сортування є алгоритмом на основі порівнянь і не є стабільним.

**Зміст роботи**

* Створив файл Sort.py, де буду реалізувати весь функціонал, який вимагається в лабораторній роботі.
* Реалізував клас Heap, в якому буде виконаний наступний функціонал:
* Partition()
* RandomizedPartition()
* RandomaizedSelect()
* QuickSort()
* QuickSortRandomized()

1. В файлі main.py прописав імпорт файлу Sort.py.

**Висновок:** на даній лабораторній роботі я ознайомився з алгоритмом швидкого сортування. Також, навчився реалізовувати даний алгоритм на мові програмування Python.

Додаток 1.

import random

def Partition(arr: list = None, start: int = 0, end: int = 0):

if arr is None:

arr = []

pivot\_index = (start + end) // 2

pivot = arr[pivot\_index]

while start < end:

while arr[start] < pivot:

start += 1

while arr[end] > pivot:

end -= 1

if start < end:

arr[start], arr[end] = arr[end], arr[start]

return end

def RandomizedPartition(arr: list = None, start: int = 0, end: int = 0):

if arr is None:

arr = []

randpivot = random.randrange(start, end)

pivot\_index = (start + end) // 2

arr[randpivot], arr[pivot\_index] = arr[pivot\_index], arr[randpivot]

return Partition(arr, start, end)

def RandomaizedSelect(arr: list = None, start: int = 0, end: int = 0, i: int = 0):

if arr is None:

arr = []

if start == end:

return arr[start]

q = RandomizedPartition(arr, start, end)

k = q - start + 1

if i == k:

return arr[q]

elif i < k:

return RandomaizedSelect(arr, start, q - 1, i)

else:

return RandomaizedSelect(arr, q + 1, end, i-k)

def QuickSort(arr=None, start: int = 0, end: int = 0):

if arr is None:

arr = []

if start < end:

pointer = Partition(arr, start, end)

QuickSort(arr, start, pointer - 1)

QuickSort(arr, pointer + 1, end)

def QuickSortRandomized(arr=None, start: int = 0, end: int = 0):

if arr is None:

arr = []

if start < end:

pointer = RandomizedPartition(arr, start, end)

QuickSort(arr, start, pointer - 1)

import copy

from Sort import \*

def readFromfile(filename: str = 'array.txt'):

file = open(filename)

res = file.readlines()

file.close()

res = [int(item) for item in res]

return res

def readFrominput():

res = input('Please, input your array: ')

res = [int(item) for item in res.strip().split()]

return res

func = {

1: readFromfile,

2: readFrominput,

}

# while True:

answer = input('Please, write way which do you want to use\n 1) Read from file\n 2) Read from terminal\n'

'Your answer: ')

try:

answer = int(answer)

except ValueError:

print('Please, write number\n')

else:

if answer not in func.keys():

print('Please, write correct number\n')

else:

array = func[answer]()

array\_1 = copy.copy(array)

array\_1.sort()

array\_2 = copy.copy(array)

answer = int(input("Which element do you want to find (from 1): "))

if answer > len(array):

print("Please, write correct number")

else:

print(f"Your array is:", \*array, sep=' ')

QuickSortRandomized(arr=array, start=0, end=len(array) - 1)

print(f"Sorted array is:", \*array, sep=' ')

print(f"Is sorted right? {array == array\_1}\n\n", )

print(f"Your element", RandomaizedSelect(array\_2, start=0, end=len(array\_2)-1, i=answer))

print(f"Min element = {array[0]}\nMax element = {array[-1]}")

index = len(array) // 2

if len(array) % 2 == 0:

print(f"Mediana = {array[index]} and {array[index+1]}")

else:

print(f"Mediana = {array[index]}")