**МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ І СПОРТУ УКРАЇНИ**

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені ІГОРЯ СІКОРСЬКОГО”

**КАФЕДРА КОНСТРУЮВАННЯ ЕОА**

ЗВІТ

з лабораторної роботи №2 по курсу «Алгоритмічні мови та програмування – 2» на тему «Алгоритми сортування»

Виконав:

студент гр. ДК-92

Бодак Є.Є

Перевірив:

ст. викладач

Губар В.Г.

Київ – 2020

Завдання:

· Ознайомлення з простими алгоритмами сортування: бульбашки, вибору, вставки

· Визначити складність кожного з алгоритмів сортування та від чого залежить складність

· Практично визначити час сортування, що затрачений кожним з алгоритмів для сортування масиву

2. Масив заповнити випадковими символами (a-z, A-Z). Заповнення організувати у вигляді окремої функції. Результат записати до файлу

Блок Схеми знаходяться в каталозі з файлами

mine – головний файл

Функції:

printArray – виводить масив в консоль

randomLetterArray – створює масив

printInFile – виводить масив в файл

deleteArray – видаляєт масив

Сортування :

bubbleSort - бульбашкою

chooseSort - виборкою

pasteSort - вставкою

Посилання на файли :

<https://github.com/sgubar/2020/tree/master/dk92/%D0%91%D0%BE%D0%B4%D0%B0%D0%BA%20%D0%84%D0%B3%D0%BE%D1%80/LAB2BODAK>

Висновок

В цій лабораторній роботі ми розглянули три базові способи сортування в мові С

Бульбашка – спосіб порівняння двох елементів масиву і постійне зсування більшого елементу до правого краю масиву

4 9 6 8 7 5 2 3 1 0

Спочатку перевіряємо перші два елементу, якщо правий більший то продовжуемо перевірку другого і третього, а тут вже робимо заміну. В кінці ми отримуємо масив в зростанні

0 1 2 3 4 5 6 7 8 9

Виборка - способом ми вибираємо найбільший елемент масиву і вставляємо в правий край і так з кожним

4 9 6 8 7 5 2 3 1 0

4 «9» 6 8 7 5 2 3 1 0

4 0 6 8 7 5 2 3 1 (9)

4 0 6 «8» 7 5 2 3 1 (9)

4 0 6 8 1 5 2 3 (8) (9)

Самий цікавий і ефективний алгоритм сортування на мою думку це вставка так я він знаходе сполуки зростання і сортує їх як групи, або розбиває на групи і фасує у вигляді зростання

7 8 9 1 2 3 4 5 6 0

(7 8 9) (1 2 3) (4 5 6) (0)

(0) (1 2 3) (4 5 6) (7 8 9)

Логіку кожного алгоритму можна розуміти по різному, головне це результат і продуктивність методу