**Міністерство Освіти І НАУКИ України**

**Національний університет "Львівська політехніка"**

Інститут **ІКНІ**

Кафедра **ПЗ**

### ЗВІТ

До лабораторної роботи № 1

**З дисципліни:** *“Алгоритми та структури даних”*

**На тему:** *“ Метод сортування бульбашкою”*

**Лектор:**

доц. каф. ПЗ

Коротєєва Т.О.

**Виконав:**

ст. гр. ПЗ-26

Матвіїв Т.Т.

**Прийняв:**

асист. каф. ПЗ

Франко А.В.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2020 р.

∑= \_\_\_\_\_ .

Львів – 2020

**Тема роботи:** Метод сортування бульбашкою.

**Мета роботи:** Вивчити алгоритм сортування бульбашкою. Здійснити програмну реалізацію алгоритму сортування бульбашкою і бульбашки з прапорцем. Дослідити швидкодію алгоритму.

**TЕОРЕТИЧНІ ВІДОМОСТІ**

Алгоритм сортування бульбашкою (англійською «Bubble Sort») відноситься до класу алгоритмів сортування вибіркою. Алгоритм працює наступним чином — у заданому наборі даних (списку чи масиві) порівнюються два сусідні елементи. Якщо один з елементів не відповідає критерію сортування (є більшим, або ж, навпаки, меншим за свого сусіда), то ці два елементи обмінюються місцями. Прохід по масиву продовжується до тих пір, доки дані не будуть відсортованими. Алгоритм отримав свою назву від того, що процес сортування згідно нього нагадує поведінку бульбашок повітря у резервуарі з водою: найлегша бульбашка піднімається до гори першою. Оскільки для роботи з елементами масиву алгоритм використовує лише порівняння, це сортування на основі порівнянь.

Складність алгоритму у найгіршому випадку дорівнює *О*(*n*²), де *n* —кількість елементів для сортування. Даний алгоритм має низьку ефективність у випадках, коли *n* є досить великим, за винятком рідких конкретних випадків, коли заздалегідь відомо, що масив з самого початку буде добре відсортований.

Одним із способів підвищення ефективності алгоритму є використання прапорця перестановок , який дозволяє швидше завершити процес сортування.

**ЗАВДАННЯ**

1. Ознайомитись з теоретичним матеріалом.
2. В обраному середовищі програмування створити віконний проект та написати програму, яка реалізує алгоритм сортування бульбашкою з використанням прапорця перестановок, та бульбашкою без прапорця.
3. Передбачити вивід у вікні вхідних даних, результатів виконання варіанту, проміжних результатів згідно роботи алгоритму та результуючий відсортований масив даних.
4. Оформити звіт про виконання лабораторної роботи.

**ХІД ВИКОНАННЯ**

Покроковий опис роботи алгоритму сортування бульбашки:

**Алгоритм B.**

Алгоритм сортування одновимірного масиву в порядку зростання.

Задано масив елементів array, SIZE – розмір масиву array, i та j – індекси проходження по масиву.

B1. Цикл за індексом проходження i. Повторювати крок В2 при i = 1..SIZE-1.

B2. Якщо arrayj+1 < arrayj, то переставити місцями елементи arrayj󠇌 і arrayj+1.

Повторювати поки j = 1,2,…,n-1.

B3. Вихід.

Покроковий опис роботи алгоритму сортування бульбашки з прапорцем

**Алгоритм BF.**

Алгоритм сортування одновимірного масиву в порядку зростання.

Задано масив елементів array, p – прапорець, SIZE – розмір масиву, i та j – індекси проходження.

BF1. Цикл за індексом проходження i. Повторювати кроки ВF2 і ВF3 при i=1..SIZE-1.

BF2. Ініціалізація прапорця перестановки: встановити p = false.

BF3. Якщо Rj+1 < Rj, то встановити p = true та переставити місцями елементи Rj і Rj+1; Повторювати поки j = 1,2,…,n-1 якщо p = false, то завершити виконання алгоритму.

BF4. Вихід.

Код алгоритму бульбашки.

void **Bubble\_Sorting\_Without\_Flag**(double\* array, int SIZE)

{

*for*(int i = 0; i < SIZE; ++i)

{

*for*(int j = 0; j < SIZE-1; ++j)

{

*if*(array[j] > array[j+1])

{

double temp = array[j];

array[j] = array[j+1];

array[j+1] = temp;

}

}

}

}

Код алгоритму бульбашки з прапорцем.

**void** Bubble\_Sort(**double**\* array, **int** SIZE)

{

**bool** p;

**for**(**int** i = 0; i < SIZE; ++i)

{

p = **false**;

**for**(**int** j = 0; j < SIZE-1; ++j)

{

**if**(array[j] > array[j+1])

{

**double** temp = array[j];

array[j] = array[j+1];

array[j+1] = temp;

p= **true**;

}

}

**if**(p == **false**)

**break**;

}

}

**РЕЗУЛЬТАТИ**

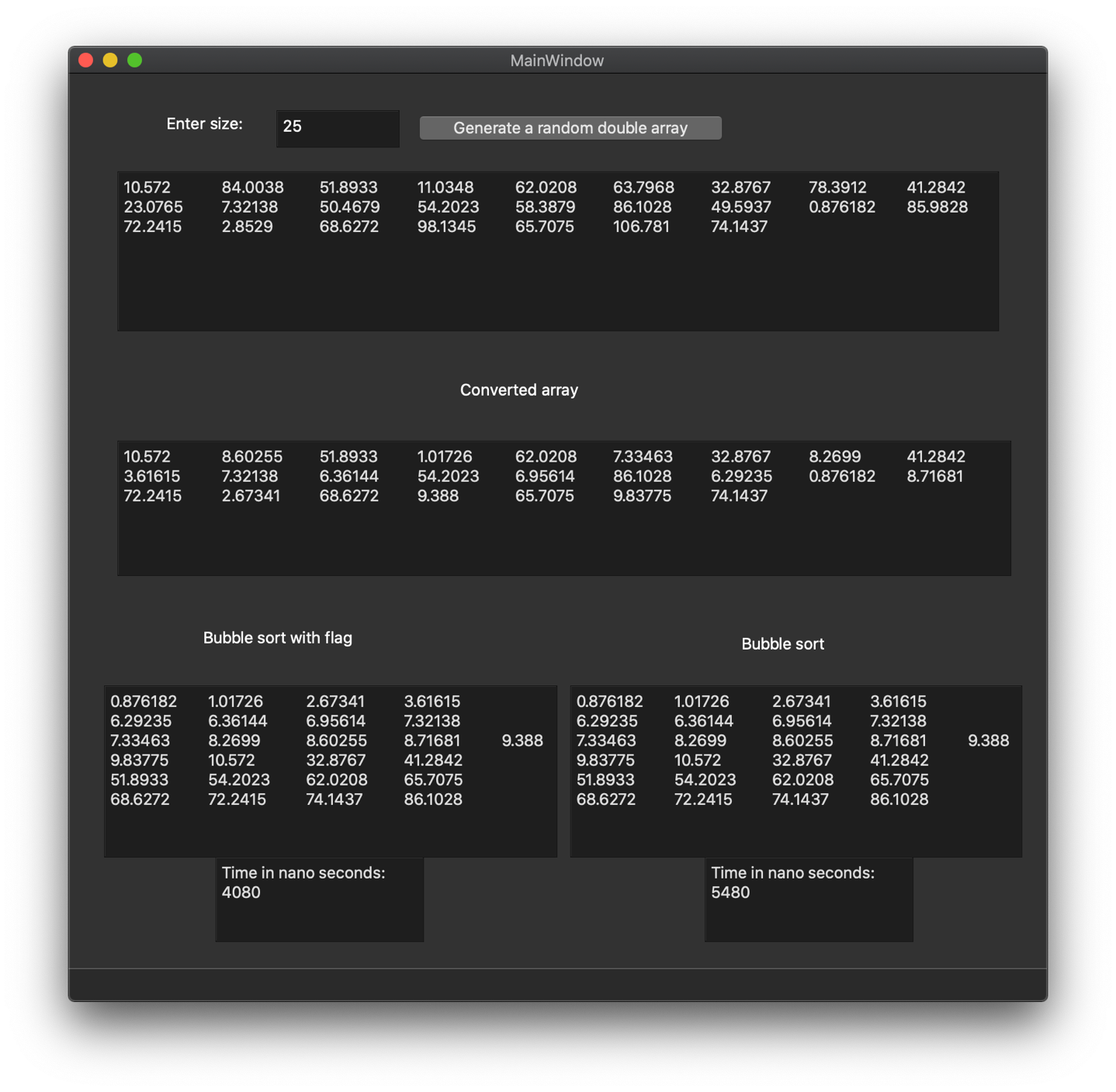


Рис. 1. Результати роботи програми

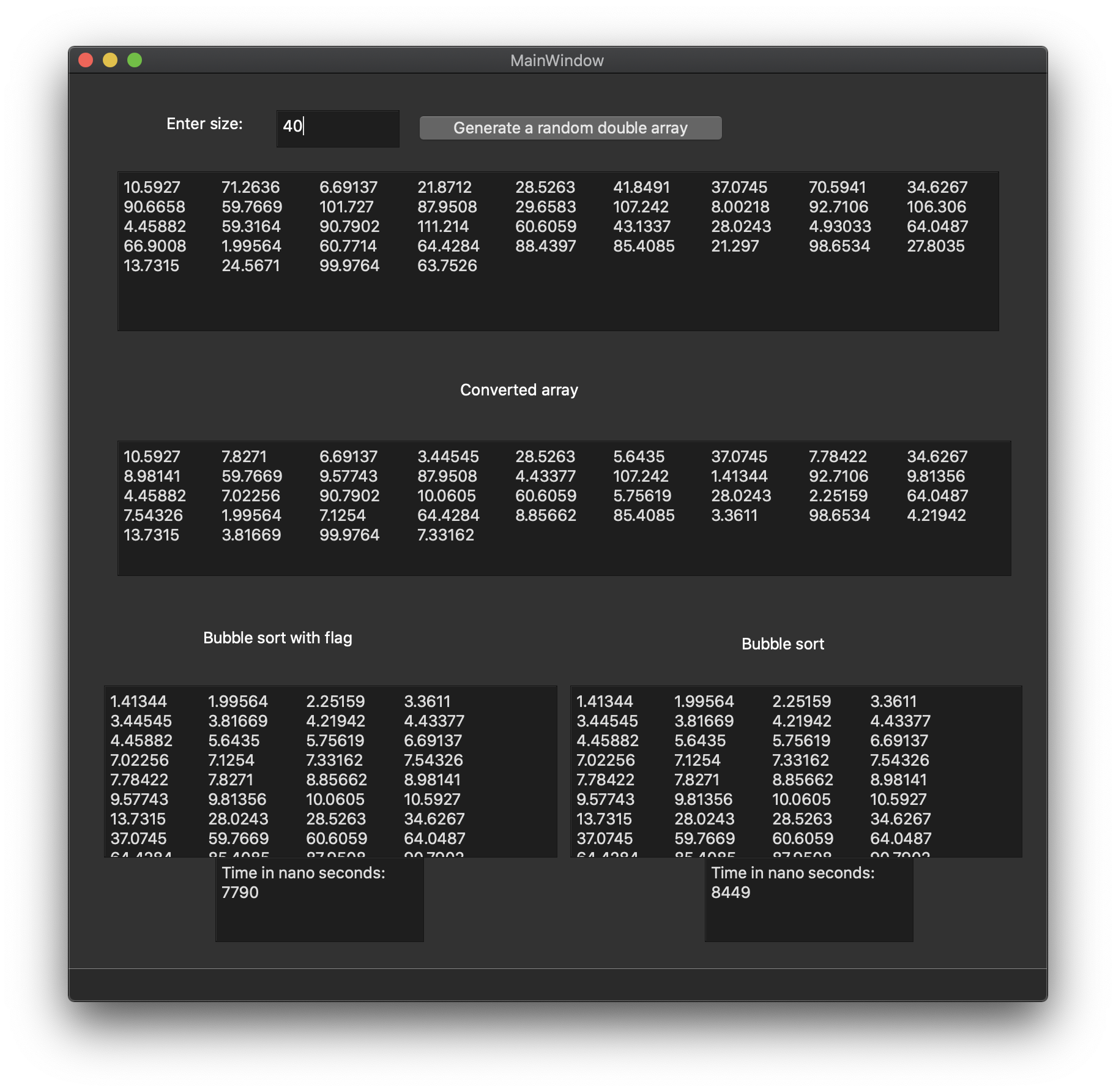


Рис. 2. Результати роботи програми

**ВИСНОВКИ**

На даній лабораторній роботі я ознайомився з алгоритмом сортування бульбашкою. Суть алгоритму в тому, що порівнюють сусідні елементи і міняють їх за потреби. В бульбашці з прапорцем закінчення сортування відбувається коли більше не відбувається перестановок. Складність алгоритму бульбашки дорівнює *О*(*n*²), де *n* — кількість елементів для сортування. При сортуванні з прапорцем алгоритм закінчується коли масив вже посортований, що дозволяє уникнути лишніх ітерацій.