LAB 09: Sorting and Searching Algorithms

1. Scope of Knowledge:

- Understand the concept and usage of arrays
- Understand of sort algorithms (bubble sort)
- Understand of search algorithms (linear search)

2. Marterials/Softwares/Tools:

- Visual Studio Code
- Draw IO (online) or Microsoft Word

3. Coding Convention:

- All identifiers must be in English and lower case
- Follow the valid identifers naming rules in C
- Tab is 4 characters
- Curly braces must be aligned
- Statements in curly brackets must be indented by 1 tab

4. Exercise:

Exercise 1:

Draw a flowchart and write a program to input an integer n, then input data for an array of n elements. Sort the array in ascending order and display the data on the screen.

Exercise 2:

Write a program that declares an integer array with 100 elements, generates a random number between 0 and 1000 assigned to that array. Print that array into 10 rows and 10 columns as follows:

```
223      25      55     954     873     665     739     875     688     266
109     779     912     266     867     526     717     529     751      69
275     137     863     963     868     934     530     423     282     663
911     960     598     542     592     580     928     453     416     809
288     585     736     279     845      75     128     309     820     884
932     139     115     529      76     232     486      56     177     638
497     802     189     811     295     266      67     801     827     292
111     160     542     857     678     593     200     402     553     654
530     691     334     657     563     315     768     707     873     880
 72     126     394     776     555     549     644     709     974     858
```

After that, sort the array in ascending order (can be done by one of the three sorting algorithms learned). Print the sorted array list to the screen in the format as above.

Exercise 3*:

Write a program to manage student names including the following menu functions:

```
            MENU
==========================
1. Add Student Name
2. Search Student Name
3. Sort Student Names
4. Normalize Student Names
5. Exit
==========================
Choose Option:
```

Hint:

Student name normalization like in the following:

```
No      Old Name            New Name
-----------------------------------------
1.      nguyen van A        Nguyen Van A
2.      Hoang A x           Hoang A Xu
3.      pham hoang nam      Pham Hoang Nam
```