Lưu ý: Bài quy đổi ra 2 slot short hoặc 1 slot Long

Title

Array and string manipulations.

Background

N/A

Program Specifications

Implement a program that manages a 100-element array of student names with the following functions:

- Add a student name into the array.
- Remove a student name.
- Search a student name.
- Output student names in ascending orders alphabetically.
- Exit the program.

Function details:

1. Function 1: The program displays a menu and asks users to input an option from 1 to 5.
   - Users run the program. The program displays and asks users to input an option from 1 to 5:
     1. Add a student name into the array.
     2. Remove a student name.
     3. Search a student name.
     4. Output student names in ascending orders alphabetically.
     5. Exit the program.
   - Users input 1: perform Function 2
   - Users input 2: perform Function 3.
   - Users input 3: perform Function 4.
   - Users input 4: perform Function 5.
   - Users input 5: exit the program.
2. Function 2: Add a student name
   - The program asks users to input a student name.
     - If the array size is already 100, then the program should display: “The list has full. It cannot add a new student.”
     - If the array size is less than 100, then the program add the student name into the array and display the message: “Student has been added to list successfully!”
   - Go back to Function 1.
3. Function 3: Remove a student name.

- Search for a student name (Function 4).
- If the student name is not existed, display a message: “Student name doesn’t exist in list.”
- If the student name is in the array, remove the name, and then display the message: “Student name has been removed from list successfully!”
- Go back to Function 1.

4. Function 4: Search a student name.
   - If the student name is not existed, return -1.
   - If the student name is in the array, return the corresponding index.
   - Go back to Function 1.
5. Function 5: Output student names in ascending orders alphabetically.
   - After printing student names’ list, go back to Function 1.

Expectation of User interface:

```
========== Student Management ==========

**************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
**************************************

Enter your choice: 1

Enter new student name: Nguyen Van A
Student has been added to list successfully!

**************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
**************************************

Enter your choice: 1
```

Add a student name into the array

```
************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
************************************

Enter your choice: 2

Enter student name to remove: Le Van B
Student name has been removed successfully!

************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
************************************

Enter your choice: 4

Total students: 1
1. Nguyen Van A
```

Remove a student name

```
************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
************************************

Enter your choice: 4

Total students: 2
1. Le Van B
2. Nguyen Van A

************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
************************************

Enter your choice: 3

Enter student name to search: Nguyen Van A
The position of student name in list is 2.
```

Search a student name

```
**************************************
1 - Add a student.
2 - Remove a student.
3 - Search a student.
4 - Print list student in an ascending folder.
5 - Exit
**************************************

Enter your choice: 4

Total students: 2
1. Le Van B
2. Nguyen Van A
```

Output student names in ascending orders alphabetically

The program exits if users input the number 5.

Guidelines

** Students should use the given dump_line() function to clear buffer after scanf() function (dump_line(stdin)).*

Use bubble sort algorithm to sort the array.