Algorithms and Data Structures 2

Laboratory Sheet 3

1. Using the program to create a 2D array of students and grades (code is in week 3), add to this code by

   i) Creating a new one dimensional array called *averages* which calculates, stores and displays the average grade for each student.
   ii) Asking the user to input which student results they wish to display (identified by number 0 to 3) and display those results.

2. Write a program that declares two strings to represent a surname (maximum character size 30) and a module name (maximum size 50). Experiment with using *scanf* and *gets* to input the values and *printf* and *puts* to display the surname. Assign the value "Algorithms2" to the module name. Using a loop, display the surname backwards one character at a time and create and display a new variable that contains just the first and last characters of the module name (get your program to extract the first and last characters, don't just assign A2 to a variable!). Remember to include the header file **string.h** at the start of the program to use the string functions. Check out the program called *stringFunctions.c* for an example of using string functions.

3. Using the **stringFunct2_long.c** program from Week 3 (which reads 5 character strings from the keyboard and displays the longest), update the program to add code to find the alphabetically smallest and largest strings and display them. Note you will need to use the string functions provided by C to do this.

4. Write a program which reads in a string (assume input is lowercase). Count the number of words in that string and the number of occurrences of each of the vowels in the text. Assume every word is separated by at least one white space character.

   for example, the following input :

   ```
   this is a sample
   ```

   would produce the following output:

   ```
   Number of words : 4
   a = 2  e = 1  i = 2 o = 0  u = 0
   ```

5. Open the program called **stringPointers.c** which is an array of months declared using a pointer. Add to this code by displaying the characters of each of the months backwards (note that even though the array is declared using a pointer, you don't need pointer notation in your code).