SSN COLLEGE OF ENGINEERING (Autonomous)
(Affiliated to Anna University, Chennai)
DEPARTMENT OF CSE

UCS 1211 PROGRAMMING IN C LABORATORY
A3: Array handling in C

==========

Learning Outcome :
To be proficient in handling arrays in C

a) using one-dimentional and two-dimentional arrays
b) passing arrays to a function

To learn modular and incremental programming.

Write the algorithm to solve the following problems and implement them in C. Solving any 5 of the following problems is mandatory. Use functions to implement the functionality.

1. The number 138 is called well ordered because the digits in the number (1,3,8) increase from left to right. (1 < 3 < 8). The number 365 is not well ordered. Write a program that will find and display all possible three digit well-ordered numbers. Also display the total number of three digit well-ordered numbers. Make use of Arrays.
2. Write a program that accepts a set of digits (0 to 9) as input and prints a vertical histogram representing the occurrences of each digit. Test your program with the set of 13 digits: 1, 7, 2 ,9 ,6, 7, 1, 3, 7, 5, 7, 9, 0

   Example
   Enter a Number : 12
   Enter 12 digits:
   1,7,2,9,6,7,1,3,7,5,7,9

   ```
         *
         *
   *     * *
   *** *** *
   ```

   0123456789
3. Given an array of integers, push all the zero's of a given array to the end of the array. Write a program in C that implements the function *pushZerosToEnd(int arr[], int n).*

   Example:
   Input : arr[] = {5, 3, 0, 1, 3, 0, 8, 0};
   Output : arr[] = {5, 3, 1, 3, 8, 0, 0, 0};

   Input : arr[] = {10, 20, 0, 0, 0, 30, 60};
   Output : arr[] = {10, 20, 30, 60, 0, 0, 0};

4. Write an interactive C program to process the exam scores for a group of students in a C programming course. Begin by specifying the number of exam scores for each student (assume this value is the same for all students in the class). Then enter each student's name and exam scores. Calculate an average score for each student, and an overall class average (an average of the individual student averages). Display the overall class average, followed by the name, the individual exam scores and the average score for each student. Store the student names in a two-dimensional character array, and store the exam scores in a two-dimensional floating-point array. Make the program as general as possible. Label the output clearly.

   Test the program using the following set of student exam grades.

Adams	45 80 80 95 55 75
Brown	60 50 70 75 55 80
Davis	40 30 10 45 60 55
Fisher	0 5 5 0 10 5
Hamilton	90 85 100 95 90 90
Jones	95 90 80 95 85 80
Ludwig	35 50 55 65 45 70
Osborne	75 60 75 60 70 80
Prince	85 75 60 85 90 100
Richards	50 60 50 35 65 70
Smith	70 60 75 70 55 75
Thomas	10 25 35 20 30 10
Wolfe	25 40 65 75 85 95
Zorba	65 80 70 100 60 95

   a) Modify this program to allow for unequal weighting of the individual exam scores. In particular, assume that each of the first four exams contributes 15 percent to the final score, and each of the last two exams contributes 20 percent.
   b) Extend the program so that the deviation of each student's average about the overall class average will be determined. Display the class average, followed by each student's name, individual exam scores, final score, and the deviation about the class average. Be sure that the output is logically organized and clearly labeled.
5. Implement the Example 9.14 program for piglatin generator given in the Text book (Byron Gottfried) . Modify it so that it can accommodate punctuation marks, uppercase letters and double-letter sounds.
6. Implement the children's hand game Rock-paper-scissors: Rock Paper Scissors is a two player game. Each player chooses one of rock, paper or scissors, without knowing the other player's choice. The winner is decided by a set of rules:
   - Rock's strength is doubled (temporarily) when fighting scissors, but halved (temporarily) when fighting paper.
   - In the same way, paper has the advantage against rock, and scissors against paper

   If both players choose the same thing, there is no winner for that round. For this task, the computer will be one of the players. Make 10 rounds of choice, display the score and winner.

~~~~~~

If you try to solve problems yourself, then you will learn many things automatically.
Spend few minutes and then enjoy the study.

~~~~~~