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

UCS 1211 PROGRAMMING IN C LABORATORY
A5: Pointers in C

==========

Learning Outcome :

To be proficient in using Pointers in C

a) Pointer notation for arrays and strings
b) passing parameters to a function by call-by-reference using pointers
c) constant pointers and pointers to constant data
d) dynamic memory allocation
e) pointers to functions

To learn modular and incremental programming.

To write maintainable code.

Write the algorithm to solve the following problems and implement them in C. Solving all the problems is mandatory.

1. Write a C function that searches a given word in a line of text and returns the frequency count. Make use of pointer notation.
2. Given multiple lines of text, parse the text to separate the tokens. A token is a word separated by a space. Store the multiple lines of text as individual strings whose maximum length is unspecified. Maintain a pointer to each string within a one-dimensional array of pointers. Identify the last line of text in some pre-determined manner. (Eg. END)
3. Implement Example program 11.22 (Adding two tables of numbers) (Page 11.26) of text book (Byron Gottfried). Modify the program creating another version so that each element in the table c is the larger of the corresponding elements in tables a and b (rather than the sum of the corresponding elements in a and b).
    a. Represent each table (each array) as a pointer to a group one dimensional arrays, as in example 11.22. Use pointer notation to access the individual table elements.
    b. Represent each table (each array) as a one-dimensional array of pointers and solve the problem.
4. Modify the program shown in Example 11.28 (displaying the day of the year) (Page 11.37) of text book (Byron Gottfried) so that it can determine the number of days between two dates, assuming both dates are beyond the base date of January 1, 1900. (Hint: Determine the number of days between the first specified date and the base date; then determine the number of days between the second specified date and the base date. Finally, determine the difference between these two calculated values.)
5. Modify the program shown in Example 11.30 (compound interest calculations) (Page 11.44) of text book (Byron Gottfried) so that it generates a table of F-values for various time periods, using different compounding frequencies. Assume that *A* and *n* are input values. Display the output in the following manner.

A = . . .
i = . . .

Interest rate =	5%	6%	7%	8%	9%	10%	11%	12%	13%	14%	15%
Frequency of Compounding											
Annual	---	---	---	---	---	---	---	---	---	---	---
Semiannual	---	---	---	---	---	---	---	---	---	---	---
Quarterly	---	---	---	---	---	---	---	---	---	---	---
Monthly	---	---	---	---	---	---	---	---	---	---	---
Daily	---	---	---	---	---	---	---	---	---	---	---
Continuously	---	---	---	---	---	---	---	---	---	---	---

Notice that the first four rows are generated by one function with different arguments, and each of the last two rows is generated by a different function.

a. Repeat the previous problem, but transpose the table so that each row represents a different value for *n* and each column represents a different compounding frequency. Consider integer values of *n* ranging from 1 to 50. Note that this table will consist of 50 rows and 6 columns. (Hint: Generate the table by columns, storing each column in a two-dimensional array. Display the entire array after all the values have been generated.)
b. Compare the programming effort required for this problem with the programming effort required for the preceding problem.

~~~~~~

If you try to solve problems yourself, then you will learn many things automatically.

Spend few minutes and then enjoy the excitement of problem solving.

~~~~~~