

PDS Lab Section 05

Lab Day 7 (Lab Test 2) – June 02, 2022

Time: 9AM to 11:15AM

The top two lines of your programs must contain the following information:

//Roll No.: <Type in your roll no.>

//Name: <Type in your name>

You have to give names to your C files as specified below and upload them in Moodle well before end time. Please read the instructions given below.

Document your programs meaningfully using appropriately named variables and sufficient amount of comments. There will be marks for documentation and proper code indentation.

1. Write a C program to achieve the following.
 - a. In the main function, declare an integer array named **plainData** of size 25.
 - i. Call the function **random-fill** with the array **plainData** as argument to fill the array **plainData** with randomly generated integers in the range [10, 25].
 - ii. Then, call the function **encode** with **plainData** as argument. [1 Mark]
 - b. The function **random-fill** should take the integer array **plainData** as its argument and fill the array **plainData** with randomly generated integers in the range [10,25]. Display the filled array. [1 Mark]
 - c. The function **encode** should take the integer array **plainData** as one of its arguments. It should declare a local integer array named **cypherData** of size 10. It should **recursively** add three numbers from the most significant position in the array **plainData** and store the result in the array **cypherData** and display the contents of the array **cypherData**. For example, if the contents of the array **plainData** is [10,15,12,17,20,25,19], then the array **cypherData** should contain [37,62,19]. **Note: Non-recursive encode routine will be awarded 0 marks.** [8 Marks]

Name your C program file as LD7_1_<roll_no>.c.

[Total=10 Marks]

2. Write a program that can be used to operate on two sets of integers. It should have the following functions:
 - a) **main**: Define two integer arrays named **set1** and **set2** of size 20 each. Fill the arrays **set1** and **set2** with random integers in the range [10,50] while taking care that an array does not contain any duplicate numbers. Then, display the two arrays properly formatted. Next, in an infinite loop, prompt the user to enter a choice (integer value). Depending on whether the user enters 1 or 2, appropriately call either the function **union** or **intersection**, and on entering any other number as choice other than 1 or 2, your program should exit. [5 Marks]
 - b) **union**: This function should take **set1** and **set2** as its arguments and form the union of the two sets (**set1** and **set2**) and store the resultant set in an array **tmp** (size 40), and then display the union properly formatted. **set1** and **set2** should remain undisturbed. Nothing needs to be returned to the calling function. [5 Marks]
 - c) **Intersection** : This function should take **set1** and **set2** as its arguments and form the intersection of the two sets (**set1** and **set2**) and store the resultant set in an array **tmp** (size 20).

It should display the resultant intersection set properly formatted. **set1 and set2** should remain undisturbed. Nothing needs to be returned to the calling function. **[5 Marks]**

Name your C program file as LD7_2_<roll_no>.c. **[Total=15 Marks]**

3. In the main program, read a character string of size at most 20 from the user and display it. Then call the functions **check**, **encode1** and **encode2**.

- a) **check**: This function should take a string of size less than 20 as its parameter and check how many times the letter 'a' to 'y' appear before the succeeding character (i.e. 'b' to 'z') separated by exactly 1 character in the string and display the result. **[5 Marks]**

Example 1. Input string "acdb": Expected output: 0

Example 2. Input string "acdbde": Expected output: 3

Example 3. Input string "aaabbbb": Expected output: 2

- b) **encode1**: This function should take a string of size less than 20 as parameter and then encode the characters of the string as follows: $a \rightarrow c, b \rightarrow d, \dots, y \rightarrow a, z \rightarrow b$. The string received as argument should be undisturbed. It should then display the encoded string. **[5 Marks]**

Example 1. Input string "acdb": Expected output: cefd

Example 2. Input string "acdbde": Expected output: cedfdg

Example 3. Input string "aaabxyz": Expected output: cccdza

- c) **encode2**: This function should take a string of size less than 20 as parameter and substitute the character 'a' before every character in the string. The string received as argument should be undisturbed. It should then display the encoded string. **[5 Marks]**

Example 1. Input string "a": Expected output: "aa"

Example 2. Input string "acdb": "aaacadab"

Example 3. Input string "aaab": Expected output: "aaaaaaab"

Name your C program file as LD7_3_<roll_no>.c.

[Total=15 Marks]

Submit your .c files in Moodle against the assignment submission link for Lab Day 7.

----- **The End** -----