To Do's for Lab Week 7

Please follow the following steps while solving the Lab exercises.

Try to solve the exercises yourselves before watching the Lab 6 recording from Spring 2021.

Part 1

Exercise 1:

Follow the next step and record and <u>submit in a separate text file the items in italics in addition to the C code for the exercise</u>.

1) What steps would solve the problem? Did we solve any similar exercises in class? What was similar and what was different in those exercises as compared to this exercise?
   Describe in common language the solving of the exercise.

   We should first initialize our variables and then read the string of characters one by one comparing them and isolating the incremented values to count characters or non letters, we should then make a formula in an if statement which checks whether y the amounts equal a situation in which its a sparse array or not

2) Identify the variables that are needed to solve the exercise.
   Indicate the needed variables and specify their purpose.

   Int i is an index which we use to internet properly through the array and the read chars of the file
   Char string[] will hold our incremented values and allow us to compare the read characters to the values that will be either letters or not letters
   Int Lettercount holds the amount of letters read
   Int notlettercount holds the amount of invalid characters read

3) Identify the type of the variables.
   Indicate the type for each variable and explain your choice.
   What is a string and how does it differ from a char variable?
   Char string

Int is able to hold discreet absolute values which is helpful for comparing the amount of chars, and is used for our index pointer which us helpful for iterating through all the valid amounts of characters

4) Read the input information from the keyword and assign the information to the related variables.
   Describe your understanding of the used C instructions.
   How do you read a string variable?

   It firsts open the file and then reads through it and compares each value with Boolean, adds numbers and finishes with a if statement to determine a spare array

   We read through the string variable with an integrated for loop or you can use string copy or printf("%s")

5) Display the values of each variable.
   Explain if the displayed value was correct. If it was not, what change did you do to your program, and why?

6) How do you distinguish letter characters from the other characters? How do you distinguish lower case letters from uppercase letters?
   Explain your reasoning.
   With Boolean comparators, since ascii values have int values baked into them we can compare them lower case or upper are two different ranges so we include those as well

7) How do you repeat steps 4-6? What is the stopping condition of the repetition?
   Explain your reasoning.
   I<=15. This is outlined in the exercise desc which says 15 elements only. The stop comes from i reaching the correct value

8) Display the required output.
   Describe the output. If it is not, what error did you get, and how do you plan to fix it? Did your correction solve the problem?

   How similar was the solution at Step 1 to the final solution?

It was correct i beleive, it may have an extra char variable in there because i <15, the solution was very similiar since this was relatively easy to figure out

Exercise 2:

Follow the next step and record and <u>submit in a separate text file the items in italics in addition to the C code for the exercise</u>.

How is Exercise 2 different from Exercise 1?
How do these differences change the steps of Exercise 1?

This is different cause we cant just read the current character we need to think about the next one as well. This requires us to store everything we need in a straight and refer to this incrementally higher and higher in the values of the array, this changes the movement through it, and the complexity of the Boolean

9) What steps would solve the problem? Did we solve any similar exercises in class? What was similar and what was different in those exercises as compared to this exercise?
   Describe in common language the solving of the exercise.

   We initialize, read, compare the characters to make sure they aren't duplicates, and then set the "last characte" to the current one and then print it out if it suits the conditions

10) Identify the variables that are needed to solve the exercise.
    Indicate the needed variables and specify their purpose.

    Ints to iterate through the string

    Chars to hold the values of the letters in order to hold the string and its values as well as the ones to hold the necessary charactets to check

11) Identify the type of the variables.
    Indicate the type for each variable and explain your choice.
    What is a string and how does it differ from a char variable?

12) Read the input information from the keyword and assign the information to the related variables.

Describe your understanding of the used C instructions.
How do you read a string variable?

You can read a string with fgetc() or with printf() you can also iterate it through EOF, we start and set the first letter as the last letter and compare that to the next and the set the last to the current and eventually we get to the end while having compared every letter Eliminating the duplicates

13) Display the values of each variable.
Explain if the displayed value was correct. If it was not, what change did you do to your program, and why?

It was correct! It matched the test case

14) How do you distinguish letter characters from the other characters? How do you distinguish lower case letters from upper case letters?
Explain your reasoning.

Was not exactly neccesary , since the objective never stated it

15) How do you repeat steps 4-6? What is the stopping condition of the repetition?
Explain your reasoning.
It's loops through till end off file since instructions never gave a loimoiting size

16) Display the required output.
Describe the output. If it is not, what error did you get, and how do you plan to fix it? Did your correction solve the problem?

How similar was the solution at Step 1 to the final solution?

I got it correct and all the test strings i put through openly came back with 1 of that duplicate letter!