ITCS 201 – Fundamentals of Programming
Lecture 13: Lab Assignments
(Submit via PC^2)

Q1: Write a program to count the number of alphabets (i.e., A-Z and a-z) and digits (i.e., 0-9) in an input string. In particular, the program first asks a user to input a string. It then iterates through each character in the string to count the number of alphabets and the number of digits. Finally, it prints out the number of alphabets and digits respectively.

Note: To simplify the problem, you can assume that the maximum length of the given input string in the test cases are always **20 characters**.

<u>**Sample inputs and outputs:**</u>
Case 1:

Input	Output	Expected screen
`sa9ivlkj1!#ja&*`	`9 2`	`sa9ivlkj1!#ja&*` `9 2`

Case 2:

Input	Output	Expected screen
`Good luck!`	`8 0`	`Good luck!` `8 0`

Case 3:

Input	Output	Expected screen
`!@#$%^@`	`0 0`	`!@#$%^@` `0 0`

Q2: Write a program that uses the `strcat()` function in `string.h` to concatenate three input strings. Specifically, it first receives three input strings (each has the maximum length of 10 characters). Then it combines these strings using commas (`", "`). Finally, it prints out the output.

Note
- You are **NOT** allowed to use `printf("%s,%s,%s", str1, str2, str3);`.
- You **MUST** use `strcat()` and/or `strcpy()` to concatenate three strings and store the output strings in a new array (let's say `all_strs`). Please make sure that the new array has enough room to store all characters from the three strings.
- You should use `printf("%s", all_strs);` to print the output.

Sample inputs and outputs:
Case 1:

Input	Output	Expected screen
hello World! students	hello,World!,students	hello World! students hello,World!,students

Case 2:

Input	Output	Expected screen
!$*!1 MUICT Mahidol	!$*!1,MUICT,Mahidol	!$*!1 MUICT Mahidol !$*!1,MUICT,Mahidol

Q3: Write a program to remove symbols, from an input string. The symbols are any characters that are not alphabets (A-Z or a-z), digits (0-9) and blank space (' '). The program prints the number of removed symbols and the output string that contains no symbols with the following pattern:

<number-of-removed-symbols> <output-string>

Note
- This program accepts **20 characters at maximum**. If the input string contains more than 20 characters, the exceed characters will be **discarded**.
- If you are using `fgets()` to receive the input string, you have to **remove the newline ('\n')**, if any, at the end of the input string. You can find more details in the slide or use the provided code below (assuming that the name of the input string is `in_str`).
- The given input strings in the test cases always contain **at least ONE** non-symbol character.

```
char *pos;
if ((pos=strchr(in_str, '\n')) != NULL)
    *pos = '\0';
```

Sample inputs and outputs:
Case 1:

Input	Output	Expected screen
He!!0 wor|d	3 He0 word	He!!0 wor|d 3 He0 word

Case 2:

Input	Output	Expected screen
Th!s !s @ bo0K.kh#!nm	6 Ths s bo0Kkhn	Th!s !s @ bo0K.kh#!nm 6 Ths s bo0Kkhn

Note: There is no 'm' in the output as 'm' is the 21st character, and the maximum length of the input string for this program is 20. So it discards the 'm'.

Q4: Write a program that accepts and stores student information in a structure variable. The student information is **first name**, **last name**, **age**, and **weight**. The program then prints out the student information using the structure variable. You need to define a structure, named `Student`, to store the information of a student. The program prints the output with the following pattern:

<first-name> <last-name>, <age>, <weight>

Note
- To simplify the problem, you can assume that the maximum length of the given first and last names in the test cases are always **20 characters**.
- The weight is printed out with **TWO** decimal places.
- You will get a **ZERO** score if you **DO NOT** define a structure to store student information and print the information using the structure variable.

Sample inputs and outputs:
Case 1:

Input	Output	Expected screen
Akara Supratak 180 267.5	Akara Supratak, 180, 267.50	Akara Supratak 180 267.5 Akara Supratak, 180, 267.50

Case 2:

Input	Output	Expected screen
Harry Potter 25 72.74	Harry Potter, 25, 72.74	Harry Potter 25 72.74 Harry Potter, 25, 72.74