<u>Laboratory Sessions 4</u>

<u>Task 1: Variables, Operators, Expressions and Statements Quiz (20%)</u>
Please finish the quiz "Lab4 Quiz" in the NEWTELP platform.

<u>Task 2: Soccer Tickets (20%)</u>

Write a program that first reads the number of people who bought soccer tickets. For each person, it reads the number of tickets bought and prints it in the following table:

CLIENT NO.	NO. TICKETS
1	2
2	3
3	1
4	4

The program should then compute and print the average numbers of tickets bought by soccer fans.

<u>Task 3: Star Printing (20%)</u>
Write a program that prints the following patterns separately, one below the other. Use "for" loops.

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

*********
********
*******
******
*****
****
***
**
*

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

EM108: Software Development for Engineers

****

Task 4: The Caesar Cipher (40%)

Encryption is the process of disguising a text so that its meaning will not be apparent to anyone who may intercept it, accidentally or not. Encryption is a fundamental technology in the development of secure electronic payment systems, which, among other things, are allowing the increasing commercial exploitation of the Internet.

In this task, you will work with a very simple encryption procedure. This is called a substitution cipher, in which each letter of the original message (plaintext) is replaced by a different letter to generate the coded message (ciphertext).

To simplify matters further, we will restrict our plaintext to use only the upper-case letters: A to Z, together with the space and newline characters; furthermore, space and newline characters will be left unchanged by the encryption process.

In each case, the replacement letter will be a fixed number of letters later in the alphabet compared to the original letter (and wrapping around back to A again after Z, as necessary). The number of letters to count forward will be called the encryption key. For example, with a key of 3, we would replace A by D, B by E and so on, with, finally, W being replaced by Z, X by A, Y by B and Z by C.

This simple kind of cipher is also called Caesar Cipher after Julius Caesar, the Roman emperor, who is said to have used it for secure battlefield messages.

To implement a Caesar cipher with a C program, note that, when accessing individual characters in C, what you are really dealing with are numeric codes for the characters. As such, you can do any normal arithmetic (addition, subtraction etc.) and comparison (greater-than, less-than etc.) operations on them. The relationship between the characters, as displayed on your screen or in your editor, and the numeric codes accessed within your C program is not specified by the C language standard. It is thus open to anyone developing a C compiler to adopt any character numbering scheme they like; you cannot reliably assume that the numbering scheme in effect for one C compiler will be the same as for another. However, there is a particular numbering scheme which is the most widely used, and to which most C compilers conform (including our compiler): American Standard Code for Information Interchange (ASCII) scheme. In ASCII, the upper case letters A to Z have consecutive numeric codes from 65 to 90. Thus, given a particular key value (a number from 1 to 25) simple addition will encypher a letter as required - except that if the sum is greater than 90 you must wrap around to the start of the alphabet again by subtracting 26, which is the length of the alphabet).

You are required to develop a program which will input a series of lines of characters, and encode them with a Caesar cipher. As each line of plaintext is input, the corresponding line of ciphertext should be output. It is up to you to devise a mechanism whereby the user can indicate that all lines have been input, and to implement the program in such a way that it will then terminate.

Originally the key for the encoding should be hard coded into your program (as opposed to be being read in at run time). Later on, **update your program to read the key**.