# Introduction to Computer Programming (CSE1001)

# Major Assignment-1

*(This assignment is designed to give you practice with Strings, User input, and Loops.)*

## Problem Statement:

To design a Caesar Cipher, this encrypts plaintext to generate ciphertext and decrypts ciphertext to generate plaintext.

## Description:

Cryptography involves creating and decoding secret messages. Suppose you want to send a message, called the **plaintext**. You encode the plaintext to create the string you will actually transmit, called the **ciphertext**. Your goal is to prevent adversaries from understanding the message if they intercept it.

Your program performs encryption on a message consisting entirely of uppercase letters using a **Caesar cipher** with a specified **key**, and decryption of a message (all uppercase letters) using a Caesar cipher with all possible keys (1, 2, ..., 25).

A Caesar cipher encrypts a plaintext message using a *key* from 1 to 25 by replacing each letter in the message by the letter *key* positions ahead in the alphabet, with wraparound to the beginning of the alphabet as needed. So with key 5, the letter 'B' is replaced with 'G', and the letter 'Y' is replaced with 'D'. Decryption reverses this process; so during decryption with key 5, 'G' is replaced by 'B'. (Hint: use the modulus operator).

***Example:*** Let the letter A to Z are numbered form 0 to 25.

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

With key=5 each letter encrypted to the following letters

| F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

With key=5 Plaintext="ZOO" encrypted to Ciphertext="ETT"
With key=4 Plaintext="HOOK" encrypted to Ciphertext="LSSO"

Your program will prompt the user for an encryption key, and a plaintext message to be encoded, and then produce the ciphertext. Then your program will prompt the user for a ciphertext message, and display the corresponding plaintext, for each possible encryption key from 1 to 25.

***Hints:***

1. You can do arithmetic on characters. For example, the ASCII value of 'A' is 65. The value of the expression 'A' + 5 is 70, and you can convert that value to the char 'E' by using an explicit cast: (char) ('A' + 5)
2. You will need to use the **length()** and **charAt()** methods of String class to find length of string and character at specific index.

```
String s = "Sachin";
int len = s.length();          //length of string is len = 6
char ch = s.charAt(4);         //character at index 4 is ch='i'
```

3. You can use append(+) operator to add characters to a blank string.

```
String s=" ";
s=s+'a';        //string s is "a"
s=s+'b';        //string s is "ab"
```

---

**Sample Run (with values entered by the user underlined):**

Enter encryption key: 4
Plaintext - uppercase letters only: HOOK
Ciphertext: LSSO
Enter ciphertext - uppercase letters only: LSSO
Plaintext for each encryption key from 1 to 25:
key 1: plaintext = KRRN
key 2: plaintext = JQQM
key 3: plaintext = IPPL
key 4: plaintext = HOOK
key 5: plaintext = GNNJ
key 6: plaintext = FMMI
key 7: plaintext = ELLH
key 8: plaintext = DKKG
key 9: plaintext = CJJF
key 10: plaintext = BIIE
key 11: plaintext = AHHD
key 12: plaintext = ZGGC
key 13: plaintext = YFFB
key 14: plaintext = XEEA
key 15: plaintext = WDDZ
key 16: plaintext = VCCY
key 17: plaintext = UBBX
key 18: plaintext = TAAW
key 19: plaintext = SZZV
key 20: plaintext = RYYU
key 21: plaintext = QXXT
key 22: plaintext = PWWS
key 23: plaintext = OVVR
key 24: plaintext = NUUQ
key 25: plaintext = MTTP

***Deadline of assignment submission by Date-30/09/2016 Time - 11.55PM***
***Email java code and output to concern faculty within the deadline.***