Name: Muhammad Rafiq

Seat No: B17101061

Course: Network Security & Cryptography Lab

Assignment Number: Assignment 2 (Affine Cipher)

Section: A

Modulo Algorithm

In [2]:
```
def modulo(a, m)->int:
    R = abs(a) % m
    if a>= 0:
        R =R
    elif a < 0 and R != 0:
        R = m - R
    elif a < 0 and R == 0:
        R = 0
    return R
```

Affine Cipher

1- Encryption of Affine Cipher

2- Decryption of Affine Cipher

3- Finding key of Affine Cipher

4- Breaking of Affine Cipher

In [22]:
```
def affine_encrypt(plain_text, key):
    plain_text = plain_text.replace(" ", "").upper()
    enc_string = ""
    for character in plain_text:
        enc_string += chr(modulo(key[0] * (ord(character)-65) + key[1], 26) + 65)
    print(enc_string)
    return enc_string

def find_a_inverse(a):
    a_list = [1,3,5,7,9,11,15,17,19,21,23,25]
    for i in a_list:
        if modulo((i*a), 26) == 1:
            return i

def affine_decrypt(cipher_text, key):
    cipher_text = cipher_text.replace(" ", "").upper()
    a_inverse = find_a_inverse(key[0])
    dec_string = ""
    for character in cipher_text:
        dec_string += chr(modulo(a_inverse * ((ord(character) -65) - key[1]),26) + 65)
    print(dec_string)

def find_key_affine(cipher_text, my_guess):
    cipher_text = cipher_text.replace(" ", "").upper()
    a_list = [1,3,5,7,9,11,15,17,19,21,23,25]
    dec_list = []
    for a in a_list:
        a_inverse = find_a_inverse(a)
        for b in range(26):
            dec_string = ""
            i = 0
            for character in cipher_text:
                dec_string += chr(modulo(a_inverse * ((ord(character) -65) - b),26) + 65)
                if dec_string in my_guess:
                    return (a ,b)
            i += 1
    return dec_list

print("Encryption of text \"PAKISTAN\" with key (15,8) is : ", end = "")
dec_string = affine_encrypt("pakistan", (15,8))
print("Decryption of text \"ZICYSHIV\" is : ", end = "")
affine_decrypt(dec_string,(15,8))
print("\n\n\n")
print("Breaking Cipher text: \"UVOHCBN NDU OYRU WGND IXXGVU OGBDUH NUODVGEQU\" ")
key = find_key_affine("NDU",["THE","ARE"])
print("My Best Guess: ")
affine_decrypt("UVOHCBN NDU OYRU WGND IXXGVU OGBDUH NUODVGEQU", key)
print()
print()
print("Question No: 02;")
key = (19,10)
affine_decrypt("piwqpgxu kbbgxi wgjniv owniei go oike hq di pgbbgwalh dah ikoy",key)
print()
print()

print("a \t a_inverse")
print("--\t--")
a_list = [1,3,5,7,9,11,15,17,19,21,23,25]
for i in a_list:
    print(i ,"\t", find_a_inverse(i))
```

```
Encryption of text "PAKISTAN" with key (15,8) is : ZICYSHIV
Decryption of text "ZICYSHIV" is : PAKISTAN



Breaking Cipher text: "UVOHCBN NDU OYRU WGND IXXGVU OGBDUH NUODVGEQU"
My Best Guess:
ENCRYPTTHECODEWITHAFFINECIPHERTECHNIQUE

Question No: 02;
DECODINGAFFINECIPHERSCHEMEISSEAMTOBEDIFFICULTBUTEASY


a        a_inverse
--      --
1        1
3        9
5        21
7        15
9        3
11       19
15       7
17       23
19       11
21       5
23       17
25       25
```