التشفير بطريقه *Affine Cipher*

القوانين :

```
/* to Encrypt With Affine Cipher :
     c = m * p + key (mod n)          where m is Multiplirer , p is character in plaintext , key is shift number , n is alpha size
 Note : to choose the right m , it must the GCD(m,n) = 1 , i.e m and n is relative prime number , i.e must use GCD method to check

 To Decryption with Affine Cipher :
     p = m` * ( c - key) (mod n)       where m` is the inverse of m
 Note : to obtain the m` it must use Extended Euclid's Method
```

الآن لمعرفه هل m و n أوليان فيما بينهما ، يجب أن يكون القاسم المشترك الأعظم لهما يساوي واحد ، فنقوم باستخدام داله GCD لإيجاد القاسم (والتي قد ذكرناها سابقا) ، ونقارن الناتج هل يساوي واحد ، اذا كان كذلك هم أوليان فيما بينهما ، والا هم غير ذلك .

```
bool AffineCipher :: checkKey (int m ,int n)
{
    if ( GCD(m,n) == 1 )
        return true ;

    else
        return false;
}
```

ونطبق قانون التشفير :

```
string AffineCipher :: encryption (string plainText  , int m , int key , int n )
{
    string str2 = "" ;

    for (int i=0 ; i<plainText.length() ; i++)
    {
        int x = ( ( ( ((int)plainText[i] - 65) * m)  + key ) % n ) + 65 ;
        str2 += (char) x ;
    }

    return str2 ;
}
```

نطبق القانون $c = p * m + key \% n$

PDF created with pdfFactory Pro trial version www.pdffactory.com