First assignment Description

1.Theoretical basis

Supposing that we have two messages : **m1** , **m2** , and we encrypt this two messages **with the same key K** , then we have two CipherTexts **c1, c2**(c1 = K XOR m1 , c2 = K XOR m2) . Cause we can olny have the CiperTexts c1 c2 , so that we can analyze from the CiperTexts. Supposing that

```
c1 XOR c2  = m1 XOR k XOR m2 XOR k = m1 XOR m2
```

Therefore we can start from this equation

Assuming that the word "the" is the begin of m1 or m2 , we can get:

```
c1 XOR c2 XOR "the" = m1 XOR m2 XOR "the"
```

which mean that we can get the three letters at the begin of other message by XOR with c1 XOR c2

Here comes my method: **we guess a word for example "the" XOR with (c1XOR c2) if the result is readable for example "who" means that "the" and "who" will be the first three letters of m1 , m2 . than , we just keep XOR the 11 messgaes, finally we can get the answer**

2. Code

There are three main parts in my python program need to be achieved

- to achieve XOR two string

  ```
  #here are the packaged function
  #decoder is a class
  #give the static function two strings will return a string which is
  (c11 XOR c10)
  res_11_10 = decoder.Two_string_OXR(c11, c10)
  ```

  our assignment gives us eleven messages which represents the encryption of hexadecimal form. However in my program the messgae is stroed by string rather that hex, which means that we need to transform the string type into hex type to operator XOR and tansform the resualt into string to stored , here is the Implementation detail (which defined in class of decoder as a static method)

  ```
  #give two hexadecimal which type of string return a string which is
  the resulat of(StringOne XOR StringTwo)
  ```

```
    @staticmethod
    def Two_string_OXR(StringOne, StringTwo):
        """两个字符串形式的十六进行异或 得到较短的"""
        length1 = len(StringOne)
        length2 = len(StringTwo)
        if length1 > length2:
            length = length2
        else:
            length = length1
        res = ""
        index = 0
        while index < length:
            res += Decode.XOR_Two_Bit(StringOne[index],
StringTwo[index]) # **
            index += 1
        return res
```

there is other function called :

```
#give two hexadecimal char return a hexadecimal with char form
Decode.XOR_Two_Bit(StringOne[index], StringTwo[index])
    @staticmethod
    def XOR_Two_Bit(BitOne, BitTwo):
        """异或两十六进制位 eg 'a' XOR '2'  """
        return hex(Decode.Bit_To_Hex(BitOne) ^
Decode.Bit_To_Hex(BitTwo))[2:]
```

we can see this function calls aothon function:

```
# give a hexadecimal char return a hexadecimal
Decode.Bit_To_Hex(BitOne)
    @staticmethod
    def Bit_To_Hex(Bit):
        """ 'a'==> 0xA  """
        if Bit == 'a':
            return 0xa
        elif Bit == 'b':
            return 0xb
        elif Bit == 'c':
            return 0xc
        elif Bit == 'd':
            return 0xd
        elif Bit == 'e':
            return 0xe
        elif Bit == 'f':
            return 0xf
        else:
            return int(Bit)
```

finally , by using the three functions we can xor two string and return a string

- to get a hex string by giving a readable text

  In order to perform well, we need to encode a text into hex form which in string

```
#give a text and return the hex in string
decoder.get_hex_code("The secret message is")
```

  here is the Implementation detail

```
    @staticmethod
    def get_hex_code(StringText):
        """得到StringText的Ascii码的十六进制 StringToASCII的反过程"""
        length = len(StringText)
        ASCiiCode = ""
        index = 0
        while index < length:
            ASCiiCode += Decode.get_hex_by_bit(StringText[index]) # **
            index += 1
        return ASCiiCode
```

  and we see aother function

```
    @staticmethod
    #give a letter and return it's ASCII with hex
    Decode.get_hex_by_bit(StringText[index])
    def get_hex_by_bit(OneChar):
        """已知OneChar为字符，得到他的ASCii码的十六进制"""
        return hex(ord(OneChar))[2:]
```

  Finally , we can give a random text's ASCII by using this function

- to transform a hex string into a readable text

  our XOR result is stored by string with hex form, however we need to output a readable text, here comes the function

```
# give a hex string and return a readable text
decoder.StringToASCII(slides_10_11)
   @staticmethod
    def StringToASCII(StringText):
        """将string中的16进制 转化为16进制所对应的ascII字符"""
        length = len(StringText)
        index = 0
        AsciiString = ""
        tmp = ""
```

```
        HexFlag = 0
        while index < length:
            if not HexFlag:
                tmp = ""
                tmp += StringText[index]
                HexFlag += 1
            else:
                tmp += StringText[index]
                AsciiString += Decode.TwoBit_To_Hex(tmp)
                HexFlag = 0
            index += 1
        return AsciiString
```

By using that function ,we can XOR all text with two encryptions see if the result is readable to judge if my guessing text is right.

3. Result

```
c11 = "The secret message is: When using a stream cipher, never use the key more than once"

c10 = " The Concise OxfordDictionary (2006) deï¬•nes crypto as the art of writing o r sol"

c9  = "A (private-key)  encryption scheme states 3 algorithms, namely a procedure for gene"

C8 = "We can see the point where the chip is unhappy if a wrong bit is sent and consumes "

c7  = "There are two types of cyptography: one that allows the Government to use brute for"

C6 = "There are two types of cryptography - that which will keep secrets safe from your l"

C5 = "You don't want to buy a set of car keys from a guy who specializes in stealing cars"

c4 = "The ciphertext produced by a weak encryption algorithm looks as good as ciphertext"

c3 = "The nice thing about Keeyloq is now we cryptographers can drive a lot of fancy cars"

C2 = "Euler would probably enjoy that now his theorem becomes a corner stone of crypto - "
```

```
c1 = "We can factor the number 15 with quantum computers. We can also
factor the number 1"
```

4.Conclusion

This program assignment needs some basic program skills. for example the assignment needs us to figure out how to XOR two hex string and how to transform a hex string into readable text and so on. also this assignment leads us to expore why a key cannot be use more than once when using a stream cipher, and gives us a deep understanding about computer security