P- Box

Key → Value

0 → 4	6 → 3
1 → 6	7 → 2
2 → 1	8 → 10
3 → 11	9 → 7
4 → 8	10 → 0
5 → 5	11 → 9

S-Box

Key → Value

0 → 4	4 → 7
1 → 2	5 → 0
2 → 6	6 → 5
3 → 1	7 → 3

Input

Chiphertext : VT_¼i ñ

Key 1 : 3L

Key 2 : KrT

Step 1: Conversion To Binary

①

Char	ASCII	Binary
VT	1	00001011
_	95	01011111
¼	188	10111100
;	59	00111011
ñ	241	11110001

②

Char	ASCII	Binary
3	51	00110011
L	76	01001100

③

Char	ASCII	Binary
K	75	01001011
r	114	01110010
T	84	01010100

Step 2 : Removing Extra Zeroes from the Chipertext To get a Multiple of 12

Chiphertext : 0000101101011111101111000011101111110001 (40 bit)

the multiple of 12 nearest To 40 from below is : 36

Chiphertext : 101101011111101111000011101111110001 (36 bit)

Step 3 : Padding with Zeroes by Setting a Multiple of 12

Chiphertext : 101101011111101111000011101111110001 (36 bit)
Key 1 : 0011001101001100 (16 bit)
Key 2 : 010010110111001001010100 (24 bit)

the multiple of 12 next To The maximum lenght : 36

Chiphertext : 101101011111101111000011101111110001
Key 1 : 000000000000000000000011001101001100
Key 2 : 000000000000010010110111001001010100

Step 4 : Division of the Chipertext into 12 bit Blocks

block 3 : 101101011111
block 2 : 101111000011
block 1 : 101111110001

Step 5: S-Box Application

For each 12-bit block, divide into groups of 3 and apply the S-Box

block 3 : 110110111100

block 2 : 110100101111

block 1 : 110100010011

Step 6: P-Box Application

block 3 : 111010100111

block 2 : 011110101011

block 1 : 001100101110

Step 7 : Xor between Key 1 and Key 2

```
000000000000000000000011001101001100
000000000000010010110111001001010100
____________________________________
000000000000010010110100000100011000
```

Step 8 : Xor between the previous Xor and P-Box

```
000000000000010010110100000100011000
111010100111011110101011001100101110
____________________________________
111010100111011110101011001100101110
```

Step 9 : Division into 12 bit Blocks

block 3 : 111010100111

block 2 : 011110101011

block 1 : 001100101110

Step 10: S-Box Application

For each 12-bit block, divide into groups of 3 and apply the S-Box

block 3 : 100001000100

block 2 : 011000111100

block 1 : 011101010010

Step 11: P-Box Application

block 3 : 000001000011

block 2 : 011010010101

block 1 : 001001111100

Step 12: Xor between Plaintext and Key 1

```
00001000011011010010110000100110000
00000000000000000000011001101001100
-----------------------------------
00001000011011010010101001001111100
```

Step 13: Removing Exstra Zeroes To get a Multiple of 8

Chiphertext : 0000101101011111101111000011101111110001 (36 bit)

multiple of 8 next To the maximum lenght : 32

Chiphertext : 101101011111101111000011101111110001

Step 14 : Final Conversion into Text

Binary	ASCII	Char
01000011	67	C
01101001	105	i
01100001	97	a
00110000	48	0

K4r4Try4