

# 西安电子科技大学

## 物联网安全实验课程 实验报告

实验名称 凯撒密码

物联网工程 1803041 班

姓名 魏红旭 学号 18030400014

同作者

实验日期 2021 年 5 月 25 日

成 绩

指导教师评语：

指导教师：

年月日

### 实验报告内容基本要求及参考格式

- 一、实验目的
- 二、实验所用仪器（或实验环境）
- 三、实验基本原理及步骤（或方案设计及理论计算）
- 四、实验数据记录（或仿真及软件设计）
- 五、实验结果分析及回答问题（或测试环境及测试结果）

## 一、实验目的：

通过实验熟练掌握凯撒密码算法，学会凯撒密码算法程序设计，提高自己的编程能力。

1. 输入一段明文和 key 值，对该明文进行加密，输出密文；
2. 输入一段密文和 key 值，对该密文进行解密，输出明文；
3. 输入一段密文，key 值未知，暴力解密后输出所有情况；

## 二、实验所用仪器（或实验环境）

计算机科学与技术学院实验中心，可接入 Internet 网台式机 44 台。

## 三、实验基本原理及要求

### 实验原理：

密码的使用最早可以追溯到古罗马时期，《高卢战记》有描述恺撒曾经使用密码来传递信息，即所谓的“恺撒密码”，它是一种替代密码，通过将字母按顺序推后起 3 位起到加密作用，如将字母 A 换作字母 D，将字母 B 换作字母 E。因据说恺撒是率先使用加密函的古代将领之一，因此这种加密方法被称为恺撒密码。这是一种简单的加密方法，这种密码的密度是很低的，只需简单地统计字频就可以破译。现今又叫“移位密码”，只不过移动的为数不一定是 3 位而已。

在密码学中，凯撒密码（或称恺撒加密、恺撒变换、变换加密）是一种最简单且最广为人知的加密技术。它是一种替换加密的技术。这个加密方法是以恺撒的名字命名的，当年恺撒曾用此方法与其将军们进行联系。恺撒密码通常被作为其他更复杂的加密方法中的一个步骤，例如维吉尼亚密码。恺撒密码还在现代的 ROT13 系统中被应用。但是和所有的利用字母表进行替换的加密技术一样，恺撒密码非常容易被破解，而且在实际应用中也无法保证通信安全。

凯撒密码是一种移位密码，具有单表密码的性质，密文和明文都使用同一个映射，为了保证加密的可逆性，要求映射都是一一对应。

加密公式：  $f(a) = (a + N) \bmod 26$

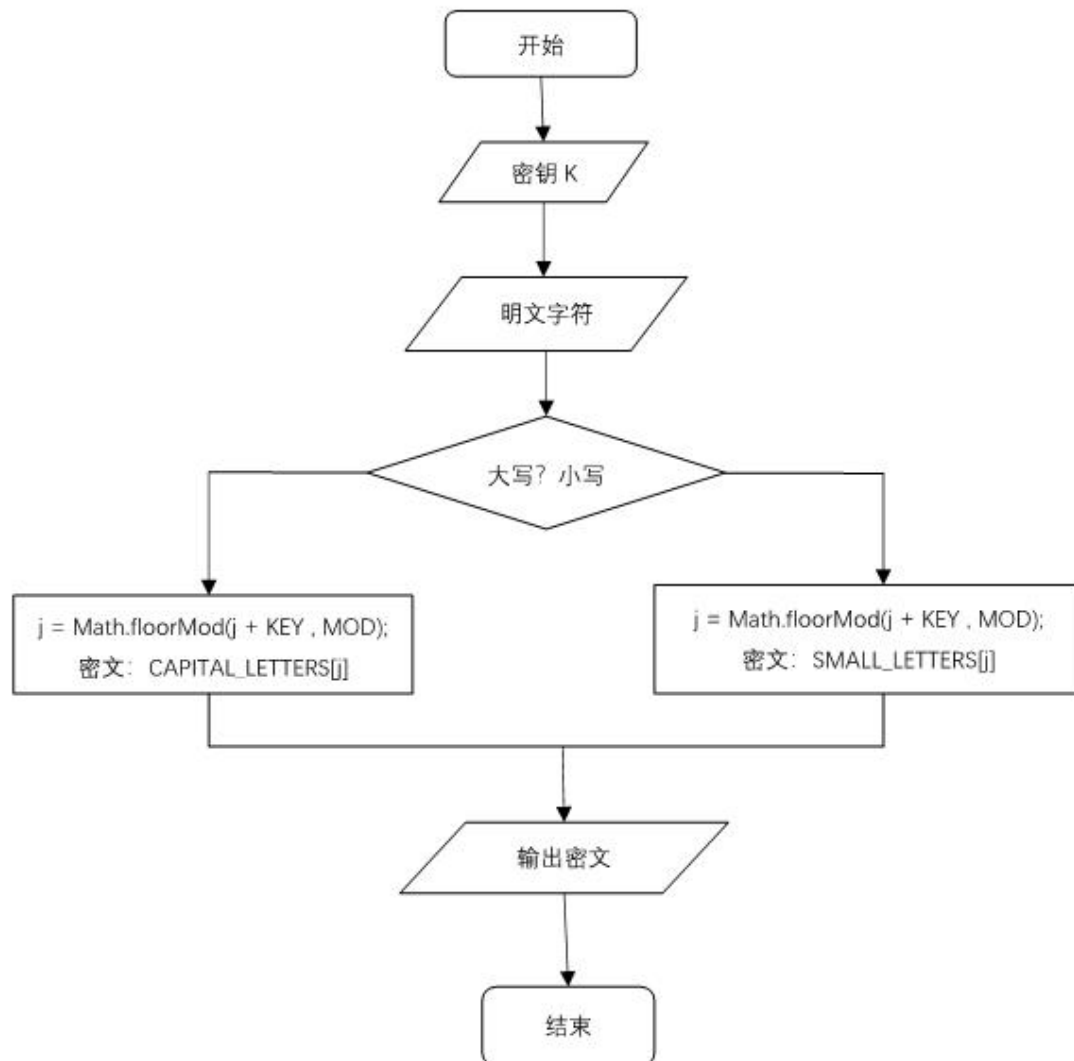
解密公式：  $f(a) = (a + (26 - N)) \bmod 26$

其中 N 代表的是位移数，也可以算是 k；

### 实验要求：

1. 输入一段明文和 key 值，对该明文进行加密，输出密文；
2. 输入一段密文和 key 值，对该密文进行解密，输出明文；
3. 输入一段密文，key 值未知，暴力解密后输出所有情况；

- 加密流程图:



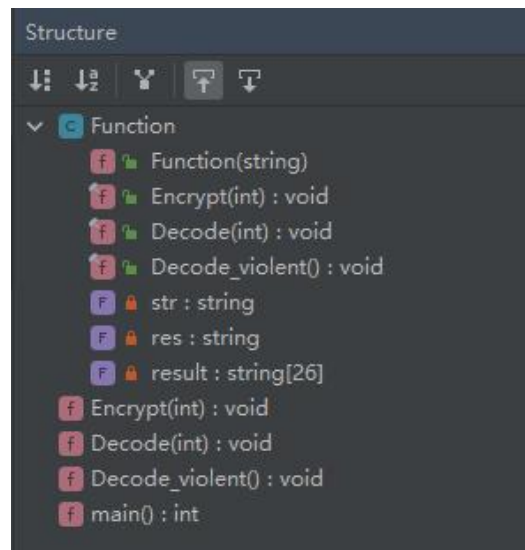
#### 四、实验步骤及实验数据记录：（要有文字描述和必要截图）

- 首先先对程序中比较容易出问题的细节进行说明：

(1) 在编写程序时，我们需要注意字符的边界为[a,z], [A,Z]，当加减 key 值后超出这个范围时，我们需要重新对其或加或减 26，以保证其能处于正确的范围内；

(2) 当对字符进行处理时，对于其中的标点符号、特殊符号或者空格，我们不进行特殊处理。以保证其原有形式的一致性。

- 代码结构:



- 加密算法:

```
void Function::Encrypt(int key) {
    for (int i = 0; i < str.size(); ++i) {
        if((str[i]<='Z' && str[i]>='A') || (str[i]<='z' && str[i]>='a')) {
            if((str[i]<='Z' && str[i]>='A' && str[i]+key>'Z') || (str[i]<='z' && str[i]>='a' && str[i]+key>'z')) res+=str[i]+key-26;
            else res+=str[i]+key;
        } else {
            res+=str[i];
        }
    }

    cout<<"-----Encryption Result-----"<<endl;
    cout<<res<<endl;
}
```

- 解密算法:

```
void Function::Decode(int key) {
    for (int i = 0; i < str.size(); ++i) {
        if((str[i]<='Z' && str[i]>='A') || (str[i]<='z' && str[i]>='a')) {
            if((str[i]<='Z' && str[i]>='A' && str[i]-key<'A') || (str[i]<='z' && str[i]>='a' && str[i]-key<'a')) res+=str[i]-key+26;
            else res+=str[i]-key;
        } else {
            res+=str[i];
        }
    }

    cout<<"-----Decryption Result-----"<<endl;
    cout<<res<<endl;
}
```

- 暴力解密算法：

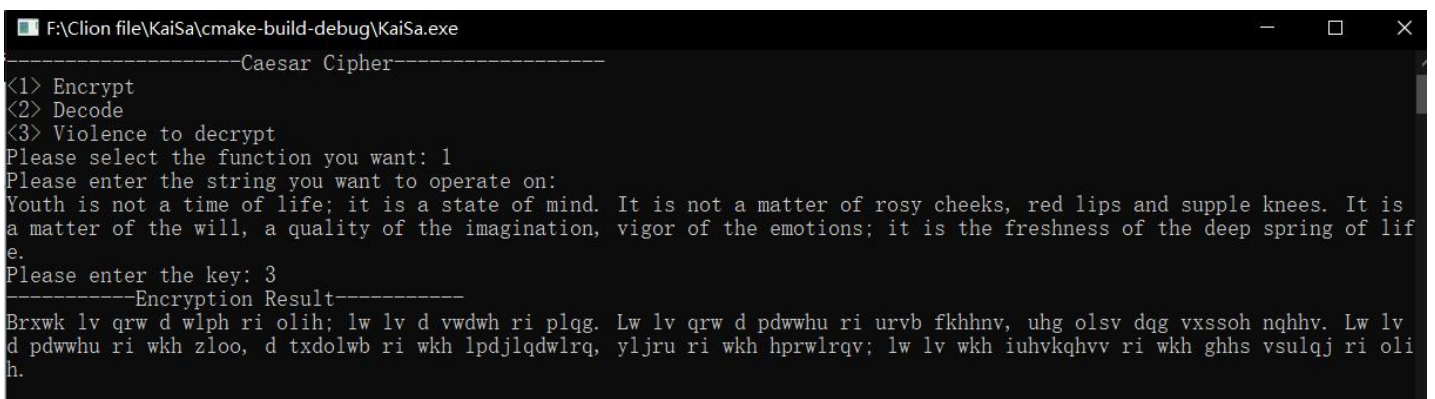
```
void Function::Decode_violent() {
    for (int i = 0; i < str.size(); ++i) {
        for (int j = 1; j < 26; ++j) {
            if((str[i]<='Z' &&str[i]>='A') || (str[i]<='z' &&str[i]>='a')){
                if((str[i]<='Z' &&str[i]>='A' &&str[i]-j<'A') || (str[i]<='z' &&str[i]>='a' &&str[i]-j<'a')) result[j-1]+=str[i]-j+26;
                else result[j-1]+=str[i]-j;
            } else{
                result[j-1]+=str[i];
            }
        }
    }

    for (int i = 0; i < 25; ++i) {
        cout<<"-----KEY: "<<i+1<<"-----"<<endl;
        cout<<result[i]<<endl<<endl;
    }
}
```

## 五、实验结果分析及实验总结与体会

- 实验结果截图：

(1) 功能一：



```
F:\Clion file\KaiSa\cmake-build-debug\KaiSa.exe
-----Caesar Cipher-----
<1> Encrypt
<2> Decode
<3> Violence to decrypt
Please select the function you want: 1
Please enter the string you want to operate on:
Youth is not a time of life; it is a state of mind. It is not a matter of rosy cheeks, red lips and supple knees. It is
a matter of the will, a quality of the imagination, vigor of the emotions; it is the freshness of the deep spring of lif
e.
Please enter the key: 3
-----Encryption Result-----
Brxwk lv qrw d wlph ri olih; lw lv d vwdwh ri plqg. Lw lv qrw d pdwwhu ri urvb fkhnhv, uhg olsv dqg vxssoh nqhhv. Lw lv
d pdwwhu ri wkh zlsoo, d txdolwb ri wkh lpdjldwlrq, yljru ri wkh hprwlrqv; lw lv wkh iuhvkqhvv ri wkh ghhs vsulqj ri oli
h.
```

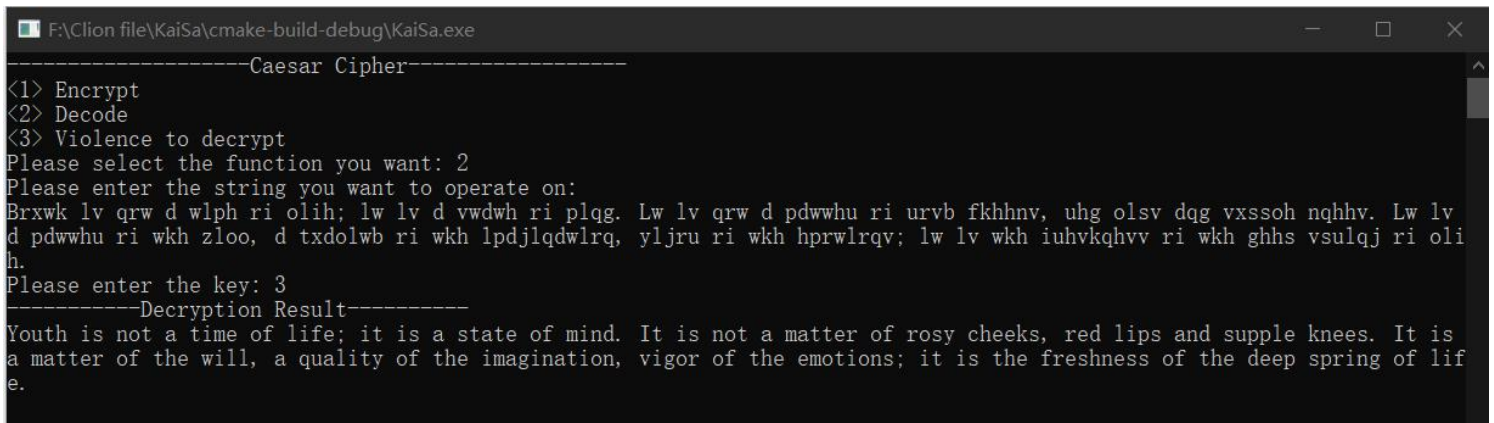
- 英文段落：

Youth is not a time of life; it is a state of mind. It is not a matter of rosy cheeks, red lips and supple knees. It is a matter of the will, a quality of the imagination, vigor of the emotions; it is the freshness of the deep spring of life.

- 加密结果:

Brxwk lv qrw d wlph ri olih; lw lv d vwdwh ri plgg. Lw lv qrw d pdwwhu ri urvb fkhnhv, uhg olsv dqg vxssoh nqhhv. Lw lv d pdwwhu ri wkh zloo, d txdolwb ri wkh lpdjldwlrq, yljru ri wkh hprwlrqv; lw lv wkh iuhvkqhvv ri wkh ghhs vsulqj ri olih.

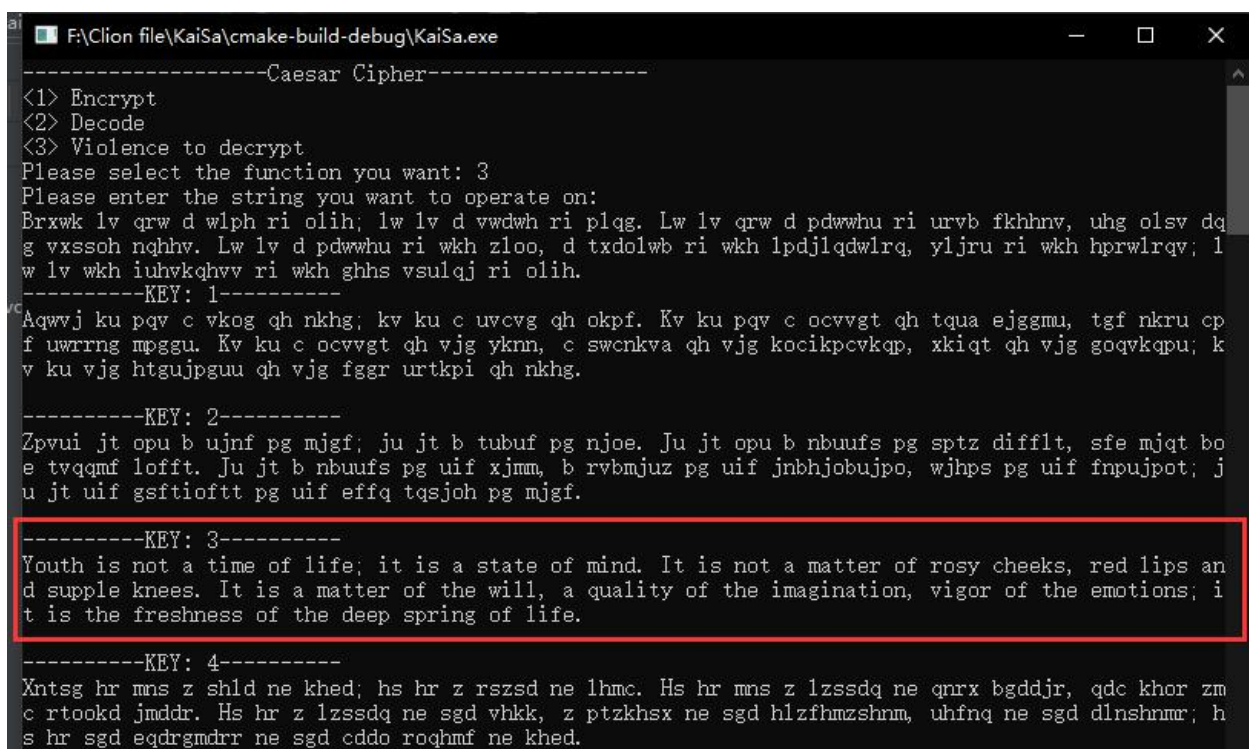
- (功能二) 实验截图



```
F:\Clion file\KaiSa\cmake-build-debug\KaiSa.exe
-----Caesar Cipher-----
<1> Encrypt
<2> Decode
<3> Violence to decrypt
Please select the function you want: 2
Please enter the string you want to operate on:
Brxwk lv qrw d wlph ri olih; lw lv d vwdwh ri plgg. Lw lv qrw d pdwwhu ri urvb fkhnhv, uhg olsv dqg vxssoh nqhhv. Lw lv d pdwwhu ri wkh zloo, d txdolwb ri wkh lpdjldwlrq, yljru ri wkh hprwlrqv; lw lv wkh iuhvkqhvv ri wkh ghhs vsulqj ri olih.
Please enter the key: 3
-----Decryption Result-----
Youth is not a time of life; it is a state of mind. It is not a matter of rosy cheeks, red lips and supple knees. It is a matter of the will, a quality of the imagination, vigor of the emotions; it is the freshness of the deep spring of life.
```

功能二我们选择解密的就是之前英文段落加密的结果，我们观察解密的结果和原段落相同。

- (功能三) 实验截图



```
F:\Clion file\KaiSa\cmake-build-debug\KaiSa.exe
-----Caesar Cipher-----
<1> Encrypt
<2> Decode
<3> Violence to decrypt
Please select the function you want: 3
Please enter the string you want to operate on:
Brxwk lv qrw d wlph ri olih; lw lv d vwdwh ri plgg. Lw lv qrw d pdwwhu ri urvb fkhnhv, uhg olsv dqg vxssoh nqhhv. Lw lv d pdwwhu ri wkh zloo, d txdolwb ri wkh lpdjldwlrq, yljru ri wkh hprwlrqv; lw lv wkh iuhvkqhvv ri wkh ghhs vsulqj ri olih.
-----KEY: 1-----
Aqvwj ku pqv c vkog qh nkhh; kv ku c uvcvg qh okpf. Kv ku pqv c ocvvgt qh tqva ejggmu, tgf nkru cp f uwrng mppgu. Kv ku c ocvvgt qh vjg yknn, c swcnkva qh vjg kocikpcvkqp, xkiqt qh vjg goqvkgpu; kv ku vjg htgujguu qh vjg fggr urtkpi qh nkhh.
-----KEY: 2-----
Zpvui jt opu b ujnfg pg mjgf; ju jt b tubuf pg njoe. Ju jt opu b nbuufs pg sptz diffit, sfe mjqat bo e tvqqmf lofft. Ju jt b nbuufs pg uif xjmm, b rvbmjuz pg uif jnbhjjobujpo, wjhps pg uif fnpujpot; ju jt uif gsftiofft pg uif effq tqsjoh pg mjgf.
-----KEY: 3-----
Youth is not a time of life; it is a state of mind. It is not a matter of rosy cheeks, red lips and supple knees. It is a matter of the will, a quality of the imagination, vigor of the emotions; it is the freshness of the deep spring of life.
-----KEY: 4-----
Xntsg hr mns z shld ne khed; hs hr z rszsd ne lhmc. Hs hr mns z lzssdq ne qnrx bgddjr, qdc khor zm c rtokod jmdrr. Hs hr z lzssdq ne sgd vhhk, z ptzkhsx ne sgd hlzfhmzshnm, uhfnq ne sgd dlnshnmr; h s hr sgd eqdrgmdrr ne sgd eddo roghmf ne khed.
```

```

-----KEY: 6-----
Vlrqe fp klq x qfjb lc ifcb; fq fp x pqxqb lc jfka. Fq fp klq x jxqgbo lc olpv zebbhp, oba ifmp xk
a prmmib hkbbp. Fq fp x jxqgbo lc qeb tfii, x nrxfqv lc qeb fjxdfkxqflk, sfdlo lc qeb bjlqflkp; f
q fp qeb cobpekbpp lc qeb abbm pmofkd lc ifcb.

-----KEY: 7-----
Ukqpd eo jkp w peia kb heba; ep eo w opwpa kb iejz. Ep eo jkp w iwppan kb nkou ydaago, naz helo wj
z oqllha gjaao. Ep eo w iwppan kb pda sehh, w mqwhpu kb pda eiwcejwpekj, reckn kb pda aikpekjo; e
p eo pda bnaodjaoo kb pda zaal olnejc kb heba.

-----KEY: 8-----
Tjpoc dn ijo v odhz ja gdaz; do dn v novoz ja hdiy. Do dn ijo v hvoozm ja mjnt xczzn, mzy gdkn vi
y npkkgz fizza. Do dn v hvoozm ja ocz rdgg, v lpvgdot ja ocz dhvbdvodji, qdbjm ja ocz zhjodjin; d
o dn ocz amznclzn ja ocz yzzk nkmdib ja gdaz.

-----KEY: 9-----
Sionb cm hin u negy iz fezy; cn cm u mnuny iz gchx. Cn cm hin u gunnyl iz lims wbyyem, lyx fcjm uh
x mojjfy ehyym. Cn cm u gunnyl iz nby qcff, u koufens iz nby cguachuncih, pcail iz nby ygincihm; c
n cm nby zlymbhym iz nby xyyj mjcha iz fezy.

-----KEY: 10-----
Rhrma bl ghm t mbfx hy ebyx; bm bl t lmtmx hy fbgw. Bm bl ghm t ftmmxk hy khlr vaxxdl, kxw ebl tg

```

## ● 实验总结:

凯撒密码是一种移位密码，具有单表密码的性质，密文和明文都使用同一个映射，为了保证加密的可逆性，要求映射都是一一对应。实际上凯撒密码的原理是比较简单的，但是在进行实现时，我们需要注意很多细节的实现，比如说字母的大小写、字母的 ASCII 范围、对于符号或者其他特殊标识的处理，处理好这些细节，我们就能保障程序的正确性。

## 六、 源代码

```

1. #include <iostream>
2. using namespace std;
3. class Function{
4. public:
5.     Function(string str){
6.         this->str=str;
7.     }
8.     void Encrypt(int);
9.     void Decode(int);
10.    void Decode_violent();
11.
12. private:
13.     string str,res;
14.     string result[26];
15. };
16.
17. void Function::Encrypt(int key) {
18.     for (int i = 0; i < str.size(); ++i) {

```



```

19.         if((str[i]<='Z'&&str[i]>='A')||(str[i]<='z'&&str[i]>='a')){
20.             if((str[i]<='Z'&&str[i]>='A'&&str[i]+key>'Z')||(str[i]<='z'&&str
                [i]>='a'&&str[i]+key>'z')) res+=str[i]+key-26;
21.             else res+=str[i]+key;
22.         } else{
23.             res+=str[i];
24.         }
25.     }
26.     cout<<"-----Encryption Result-----"<<endl;
27.     cout<<res<<endl;
28. }
29.
30. void Function::Decode(int key) {
31.     for (int i = 0; i < str.size(); ++i) {
32.         if((str[i]<='Z'&&str[i]>='A')||(str[i]<='z'&&str[i]>='a')){
33.             if((str[i]<='Z'&&str[i]>='A'&&str[i]-key<'A')||(str[i]<='z'&&str
                [i]>='a'&&str[i]-key<'a')) res+=str[i]-key+26;
34.             else res+=str[i]-key;
35.         } else{
36.             res+=str[i];
37.         }
38.     }
39.     cout<<"-----Decryption Result-----"<<endl;
40.     cout<<res<<endl;
41. }
42.
43. void Function::Decode_violent() {
44.     for (int i = 0; i < str.size(); ++i) {
45.         for (int j = 1; j < 26; ++j) {
46.             if((str[i]<='Z'&&str[i]>='A')||(str[i]<='z'&&str[i]>='a')){
47.                 if((str[i]<='Z'&&str[i]>='A'&&str[i]-j<'A')||(str[i]<='z'&&str
                    tr[i]>='a'&&str[i]-j<'a')) result[j-1]+=str[i]-j+26;
48.                 else result[j-1]+=str[i]-j;
49.             } else{
50.                 result[j-1]+=str[i];
51.             }
52.         }
53.     }
54.     for (int i = 0; i < 25; ++i) {
55.         cout<<"-----KEY: "<<i+1<<"-----"<<endl;
56.         cout<<result[i]<<endl<<endl;
57.     }
58. }
59.

```



```

60. int main() {
61.     char n[1000]={0};
62.     int choice=0,a;
63.     string by="\n";
64.     cout<<"-----Caesar Cipher-----"<<endl;
65.     cout<<"<1> Encrypt"<<endl;
66.     cout<<"<2> Decode"<<endl;
67.     cout<<"<3> Violence to decrypt"<<endl;
68.     cout<<"Please select the function you want: ";
69.     cin>>choice;
70.     getline(cin,by);
71.     cout<<"Please enter the string you want to operate on: "<<endl;
72.     cin.getline(n,1000);
73.     string str(n);
74.     Function fc(str);
75.     switch (choice) {
76.         case 1:{
77.             cout<<"Please enter the key: ";
78.             cin>>a;
79.             fc.Encrypt(a);
80.             break;
81.         }
82.         case 2:{
83.             cout<<"Please enter the key: ";
84.             cin>>a;
85.             fc.Decode(a);
86.             break;
87.         }
88.         case 3:{
89.             fc.Decode_violent();
90.             break;
91.         }
92.     }
93.     return 0;
94. }

```