杭州電子科技大学
HANGZHOU DIANZI UNIVERSITY

《密码学实验》

实验报告

实验 2：Caesar 密码算法编程实验

姓　　名：易涛

学　　号：16031330

专　　业：信息安全

实验时间：2018.04.09

指导老师：吕秋云

一．实验目的

实验环境：Windows10 Visual Studio 2017

实验目的：

掌握 Caesar 密码加解密原理，并利用 Visual C++编程实现

加密原理：Caesar 密码的加密原理是对明文加上一个密钥（偏移值）得到密文，假设密钥为 3，那么字母“a”对应的 ASCII 码为 97，加上 3 得到 100 正好是‘d’的 ASCII 码值。

实验要求：

1. 采用 MFC 编程，实现简单界面编程，界面中包含明文输入、密文输出、密钥设定等编辑框，另外提供加密和解密的按钮。
2. 对密钥的限定说明，密钥为一个整数，无大小限制
3. 对输入明文字符限定说明，只限定英文字母并区分大小写字母，采用如下公式进行加密运算

   密文字符 = ‘a’或‘A’+（明文字符-‘a’或‘A’+password%26 + 26）%26

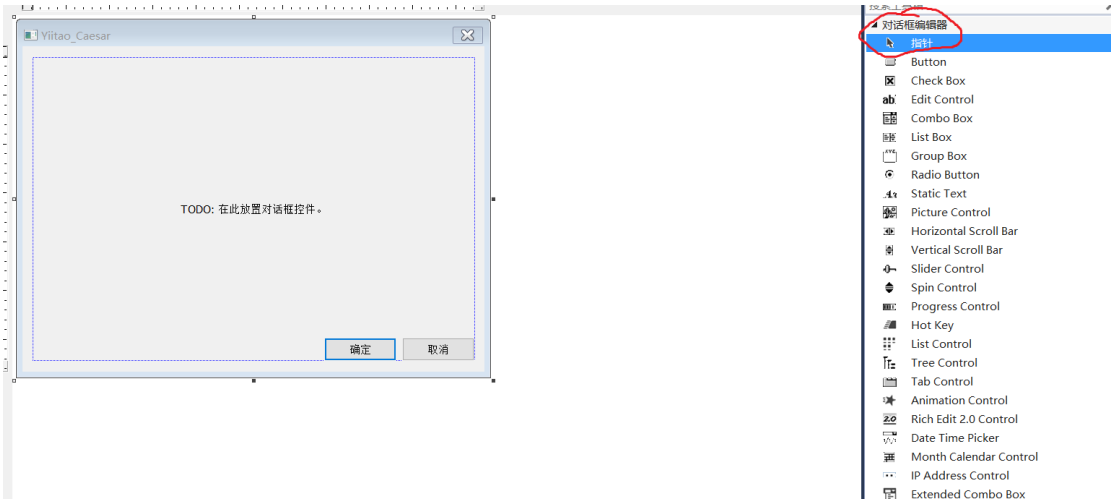
   解密反之，对数字 0~9 也加解密，其余字符不做处理，原样输出。

二．实验内容及实现过程步骤

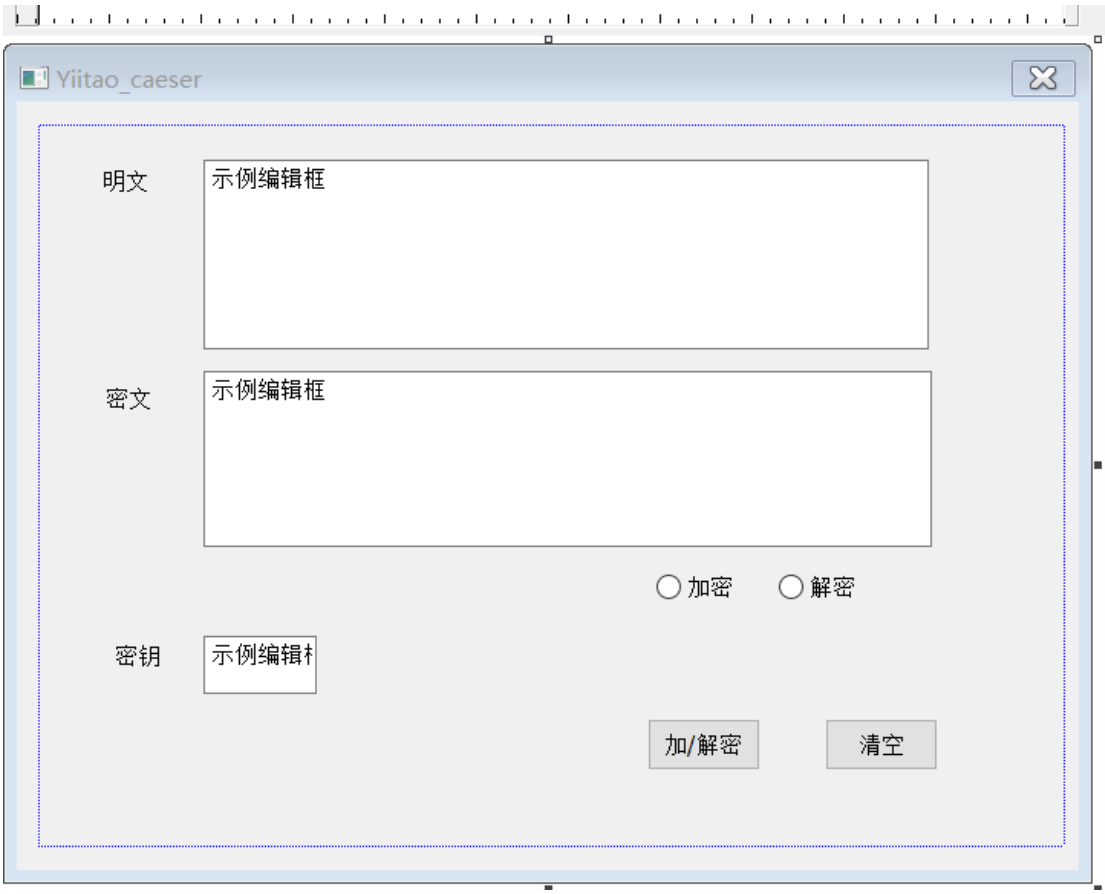
1. 采用 MFC 编程，实现界面编程

1> 在 Visual studio2017 中创建一个 MFC 应用程序,选择基于对话框创建

2> 在界面编辑处，选择对话框编辑器添加自己所需要的组件



3>　　自定义画好界面之后如图所示



2. 加解密程序处理代码

1> 凯撒加解密原理相似，可通过同一个函数实现，双击上图界面上的加/解密按钮进入该按钮的代码编辑，开始先将界面的按钮文本框等控件做一个绑定。如下图

```
void CYiitao_caeserDlg::OnBnClickedButton1()
{
    // TODO: 在此添加控件通知处理程序代码
    int check_1 = ((CButton*)GetDlgItem(IDC_RADIO1))->GetCheck();//选中加密按钮为1，否则为0
    int check_2 = ((CButton*)GetDlgItem(IDC_RADIO2))->GetCheck();//选中解密按钮
    CString message;
    CString cipher;
    CString key;
    GetDlgItemText(IDC_EDIT1, message);//获取文本框内容传给message
    GetDlgItemText(IDC_EDIT3, key);//获取密钥文本框内容传给key
    char *message1 = message.GetBuffer(0);
```

2> 加密和解密密钥不同，其余步骤相同，通过与加密、解密绑定的 check_1 和 check_2 变量选择进行加密或解密操作，同时生成各自加解密密钥，其中 key1 为对字母加解密的密钥，key2 为对数字加解密的密钥，key 为输入的密钥，因为输入密钥不限定其大小，所以对其进行求余操作，同时定义 size 变量等于明文长度，用于

循环明文字符。

```
    if (check_1 == 1 && check_2 == 0)//选中加密
    {
        key1 = _ttoi(key) % 26;//将cstring转换成int型
        key2 = _ttoi(key) % 10;
    }
    else if (check_1 == 0 && check_2 == 1)
    {
        key1 = 26 - _ttoi(key) % 26;
        key2 = 10 - _ttoi(key) % 10;

    }
    cipher1 = message1;//默认密文等于明文
    int size = message.GetLength();
```

3> 加解密代码，循环对明文每一个字符加密，首先判断明文字符属于大写还是小写还是数字，分别处理，由于大小写字母和数字各自的 ascii 码是连续的，故判断完成后根据结果，对字母明文加上经密钥求余 26 的密钥移位，对数字明文加上经密钥

求余 10 的密钥移位，当超出范围时减去 26 或 10。

```
for (int i = 0; i < size; i++)
{
    if (message1[i] >= 'a'&&message1[i] <= 'z')//小写字母
    {
        if (message1[i] + key1 > 122)//122为‘z’的ascii码
        {
            cipher1[i] = message[i] + key1 - 26;
        }
        else
        {
            cipher1[i] = message[i] + key1;
        }
    }
    if (message1[i] >= 'A'&&message1[i] <= 'Z')//大写字母
    {
        if (message1[i] + key1 > 90)//90为‘Z’的ascii码
        {
            cipher1[i] = message[i] + key1 - 26;
        }
        else
        {
            cipher1[i] = message[i] + key1;
        }
    }
    if (message1[i] >= '0'&&message[i] <= '9')
    {

        if (message1[i] + key2 > 57)//57为‘9’的ascii码
        {
            cipher1[i] = message[i] + key2 - 10;
        }
        else
        {
            cipher1[i] = message[i] + key2;
        }

    `
```

4> 加密完成之后，将密文传送到密文文本框，同时对清空按钮进行编写

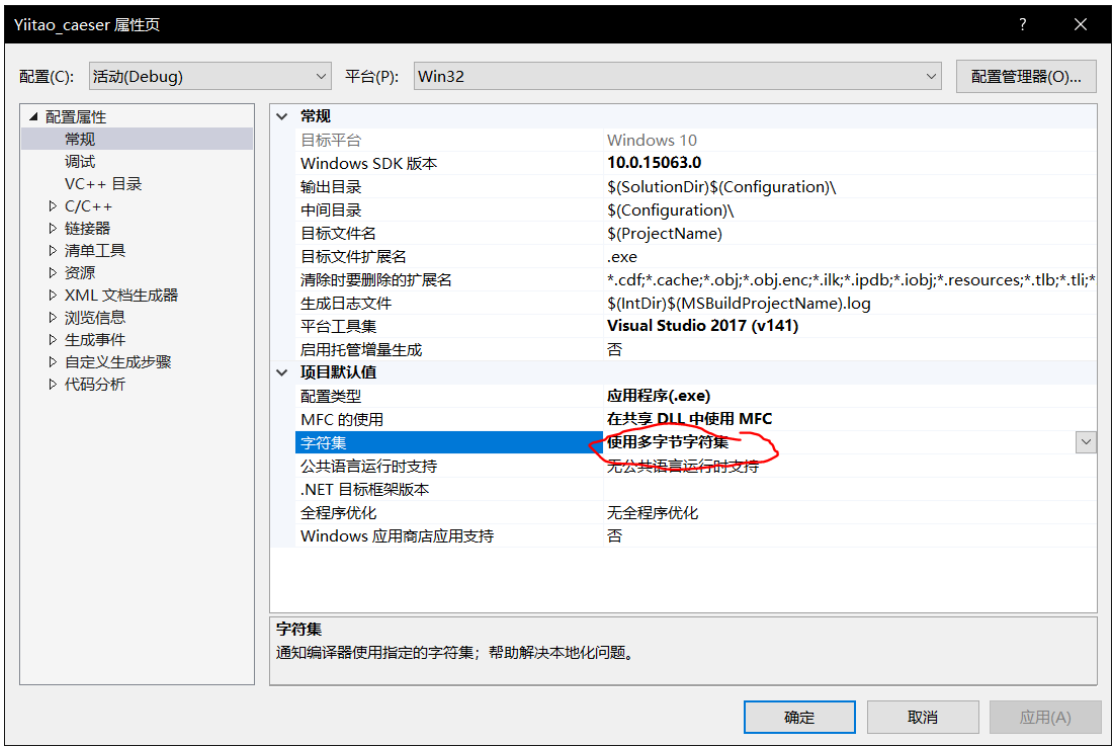
```
    }
    cipher = cipher1;
    SetDlgItemText(IDC_EDIT2, cipher);

}


void CYiitao_caeserDlg::OnBnClickedButton2()
{
    // TODO: 在此添加控件通知处理程序代码
    CString empty("");
    SetDlgItemText(IDC_EDIT1, empty);
    SetDlgItemText(IDC_EDIT2, empty);
    SetDlgItemText(IDC_EDIT3, empty);
}
```

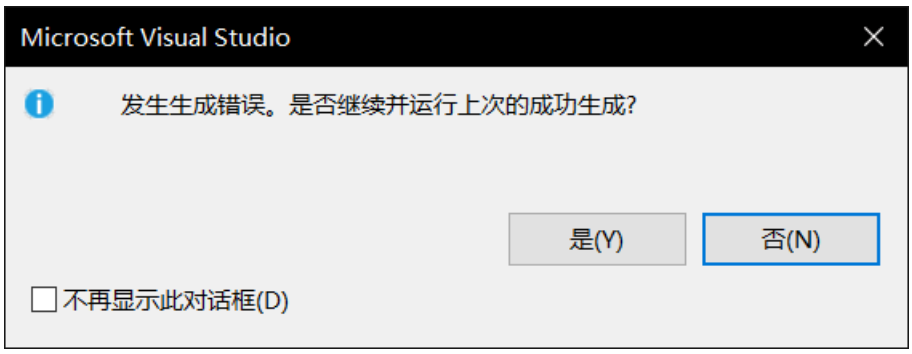
3.验证凯撒加密


Yiitao_caeser
明文
2177 drsc sc Mkockb Mszrob
密文
4399 this is Caesar Cipher
加密
解密
密钥
7758
加/解密
清空

三．实验小结

1.本次实验首次采用 MFC 编程，很多操作还不熟悉，例如在进行编程时要选用多字节字符集而不能使用 Unicode 字符集，vs2017 中创建项目的时候并没有让选择字符集，要在创建项目之后打开项目属性改为多字节字符集，否则将出现编译错误。





2.在 MFC 编程中，注意每个控件都有各自的 ID 用以绑定相关代码函数，在创建界面和进行代码编写时注意 ID 一致

3.Caesar 加密属于移位密码，较为简单，代码思想容易，不容易出错，但尽量把代码写的简洁一点，健壮性强点。