实验二 分组密码算法DES

一、实验目的

通过用DES算法对实际的数据进行加密和解密来深刻了解DES的运行原理。

二、实验原理

DES算法将明文分成64位大小的众多数据块，即分组长度为64位。同时用56位密钥对64位明文信息加密，最终形成64位的密文。如果明文长度不足64位，即将其扩展为64位（如补零等方法）。

具体加密过程首先是将输入的数据进行初始置换（IP），即将明文M中数据的排列顺序按一定的规则重新排列，生成新的数据序列，以打乱原来的次序。然后将变换后的数据平分成左右两部分，左边记为L0，右边记为R0，然后对R0实行在子密钥（由加密密钥产生）控制下的变换f，结果记为f（R0，K1），再与L0做逐位异或运算，其结果记为R1，R0则作为下一轮的L1。如此循环16轮，最后得到L16、R16，再对L16、R16实行逆初始置换IP－1，即可得到加密数据。解密过程与此类似，不同之处仅在于子密钥的使用顺序正好相反。

三、实验环境

运行Windows操作系统的PC机，具有VC等语言编译环境

四、实验内容和步骤

1．算法分析：对课本中DES算法进行深入分析，对初始置换、E扩展置换，S盒代换、轮函数、密钥生成等环节要有清晰的了解，并考虑其每一个环节的实现过程。

2．DES实现程序的总体设计：在第一步的基础上，对整个DES加密函数的实现进行总体设计，考虑数据的存储格式，参数的传递格式，程序实现的总体层次等，画出程序实现的流程图。

3．在总体设计完成后，开始具体的编码，在编码过程中，注意要尽量使用高效的编码方式。

4．利用3中实现的程序，对DES的密文进行雪崩效应检验。即固定密钥，仅改变明文中的一位，统计密文改变的位数；固定明文，仅改变密钥中的一位，统计密文改变的位数。

五、执行结果

程序代码在压缩包的源代码文件中，下面给出程序执行结果（对每一组样例进行验证）。

```
请输入(x,y)       x:加密0,解密1     y:加解密分别10组，输入第y组
0 0
0x82, 0xDC, 0xBA, 0xFB, 0xDE, 0xAB, 0x66, 0x02
```

```
请输入(x,y)       x:加密0,解密1     y:加解密分别10组，输入第y组
0 1
0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

```
请输入(x,y)       x:加密0,解密1     y:加解密分别10组，输入第y组
0 2
0x40, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

```
请输入(x,y)       x:加密0,解密1     y:加解密分别10组，输入第y组
0 3
0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

```
请输入(x,y)       x:加密0,解密1     y:加解密分别10组，输入第y组
0 4
0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

```
请输入(x,y)       x:加密0,解密1     y:加解密分别10组，输入第y组
0 5
0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
0 6
0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
0 7
0x02,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
0 8
0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
0 9
0x00,0x80,0x00,0x00,0x00,0x00,0x00,0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 0
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 1
0x95,0xF8,0xA5,0xE5,0xDD,0x31,0xD9,0x00
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 2
0xDD,0x7F,0x12,0x1C,0xA5,0x01,0x56,0x19
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 3
0x2E,0x86,0x53,0x10,0x4F,0x38,0x34,0xEA
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 4
0x4B,0xD3,0x88,0xFF,0x6C,0xD8,0x1D,0x4F
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 5
0x20,0xB9,0xE7,0x67,0xB2,0xFB,0x14,0x56
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 6
0x55,0x57,0x93,0x80,0xD7,0x71,0x38,0xEF
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 7
0x6C,0xC5,0xDE,0xFA,0xAF,0x04,0x51,0x2F
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 8
0x0D,0x9F,0x27,0x9B,0xA5,0xD8,0x72,0x60
```

```
请输入(x,y)       x:加密0,解密1      y:加解密分别10组，输入第y组
1 9
0xD9,0x03,0x1B,0x02,0x71,0xBD,0x5A,0x0A
```

六、验证雪崩效应

利用上面的程序对DES的密文进行雪崩效应验证。

- 固定每一组的密钥，利用随机数随机改变明文中的一位，共八次，统计每一次加密后密文改变的位数，计算平均值。
- 固定每一组的明文，利用随机数随机改变密钥中的一位，共八次，统计每一次加密后密文改变的位数，计算平均值。

下面是明文变化的程序执行结果。

```
明文第51位发生变化
0x82,0xDC,0xBA,0xFB,0xDE,0xAB,0x66,0x2,
0xC9,0xEC,0x03,0x0B,0xB1,0x65,0x13,0xBB
密文有36位发生变化
```

统计结果

明文变化一位										
	第1组	第2组	第3组	第4组	第5组	第6组	第7组	第8组	第9组	第10组
1	24	24	32	31	33	22	34	32	24	34
2	33	32	22	37	27	28	35	34	40	28
3	23	32	39	33	31	33	30	36	41	34
4	37	36	33	32	37	35	34	34	31	37
5	29	27	33	37	31	35	32	42	33	34
6	24	29	35	40	26	30	32	33	24	32
7	30	28	31	37	25	32	27	34	39	38
8	28	34	30	37	30	32	35	30	31	27
合计	228	242	255	284	240	247	259	275	263	264
平均值	22.8	24.2	25.5	28.4	24	24.7	25.9	27.5	26.3	26.4
总和	2557									
总平均	31.963									

下面是密钥变化的执行结果。

```
密钥第45位发生变化
0x82, 0xDC, 0xBA, 0xFB, 0xDE, 0xAB, 0x66, 0x2,
0x22, 0x4D, 0xD0, 0x08, 0x84, 0x9C, 0xF6, 0xA1
密文有30位发生变化
```

统计结果

密文变化一位										
	第1组	第2组	第3组	第4组	第5组	第6组	第7组	第8组	第9组	第10组
1	34	27	35	35	38	40	37	29	26	33
2	30	29	36	38	32	36	29	41	34	28
3	34	31	33	31	35	32	36	42	37	30
4	30	33	32	27	34	31	36	29	31	26
5	28	27	32	27	28	29	35	32	28	39
6	31	33	28	31	43	39	28	29	30	31
7	29	30	26	32	29	31	34	32	32	34
8	37	28	32	29	32	34	28	29	31	32
合计	253	238	254	250	271	272	263	263	249	253
平均值	25.3	23.8	25.4	25	27.1	27.2	26.3	26.3	24.9	25.3
总和	2566									
总平均	32.075									

结论

可以看到在仅仅改变一位明文或密钥的情况下，得出的密文和正确的相差位数在32位左右，即占密文的一半。