

## 计算机组成原理和汇编程序设计实验

1. 编写一个累计加法，从 1 加到 5，将结果保存至 AX 中。
2. 编写一个累计减法，被减数是 10011000B，减数是 01000000B，连续减 5 次，观察 FLAGS 的变化
3. 编写一个 16 位的乘法，被乘数是 100H，乘数是 100H，观察 Flags 的变化，编写一个 32 位的乘法，被乘数是 0F0FH，乘数是 FF00H，观察 Flags 的变化。
4. 编写一个 16 位的除法，被除数是 100H，除数是 100H，观察 Flags 的变化，编写一个 32 位的除法，被除数是 0F0FH，除数是 00FFH，观察 Flags 的变化。
5. 编写一个累计加法，被加数是 0FH，加数是 01H，观察 Flags 的变化，被加数是 0FFH，加数是 01H，观察 Flags 的变化，被加数是 0FFFH，加数是 01H，观察 Flags 的变化，被加数是 FFFFH，加数是 01H，观察 Flags 的变化，被加数是 FFFFFFFFH 加数是 01H，观察 Flags 的变化。
6. 编写一个移位运算，将 8F1DH 存至 AX，然后用指令右移 1 位然后左移 1 位，显示结果并观察 Flags 的变化。将 8F1DH 存至 AX 中，然后带 CF 位左移 5 位，并右移 7 位，观察 Flags 的变化，并给出结果。
7. 将 71D2H 存至 AX 中，5DF1H 存至 CX 中，DST 为 AX，REG 为 AX，实现

双精度右移 2 次，交换 DST 与 REG，然后左移 4 次，分别查看结果.

8. 实现压缩 BCD 码的加减法,用压缩 BCD 码实现 $(21+71)$ , $(12+49)$ , $(65+82)$ ,  
 $(46-33)$ , $(74-58)$ , $(43-54)$  的十进制加减法。然后又用非压缩 BCD 实现  
上述 6 个式子。
9. 实现 KMP 算法，输入两个字符串（可以直接保存在内存中），实现快速匹配
10. 斐波纳契数列：1, 1, 2, 3, 5, 8, 13。通常可以使用递归函数实现，现用  
汇编实现该过程。