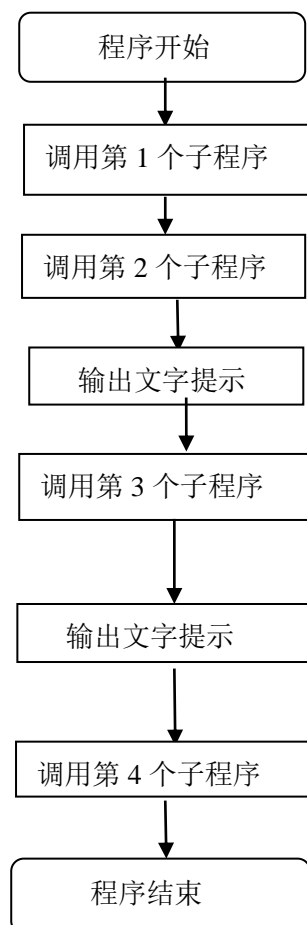


一、设计题目

某学院举行联欢晚会，在入口签到处登记参会学生的学号存入 `stu.txt` 文件（文件为只含 7 位数学号的文本文件，请自行用记事本产生该文件，也可直接使用附件中的 `mustu.txt` 文件）。编一个开奖程序，打开并读取文件中的数据，计算参会人数 N ，利用 BIOS 的 1AH 时钟中断产生一个随机数 M （1- N 之间），根据该随机数确定本次中奖人员为文件中的第 M 个学生，并在屏幕上显示该幸运数和中奖学号。

二、设计说明

1. 本程序通过读取一个现有的存储若干个学生学号的文件，计算文件中存入的学号个数 N ，并生成一个 1- N 之间的随机数 M ，显示该随机数 M 以及文件中第 M 个学号。
2. 整个程序可分为 5 个部分（4 个子程序以及 1 个主程序），分别完成以下功能。
第 1 个子程序：打开文件并读取文件中所有数据，计算文件中存入的学号个数 N ；
第 2 个子程序：利用 BIOS 的时钟中断产生随机数，并用产生的随机数除 N 取余，得到 0- $(N-1)$ 之间的随机数，该随机数加一后得到 1- N 之间的随机数 M ；
第 3 个子程序：将随机数 M 按照十进制输出；
第 4 个子程序：打开文件并移动指针至第 M 个学号处，读取第 M 个学号并显示在屏幕上。
主程序按照顺序依次调用 4 个子程序，并输出必要的提示语。
3. 程序框图如下：



4. 子程序说明以及各子程序流程图如下：

第 1 个子程序：

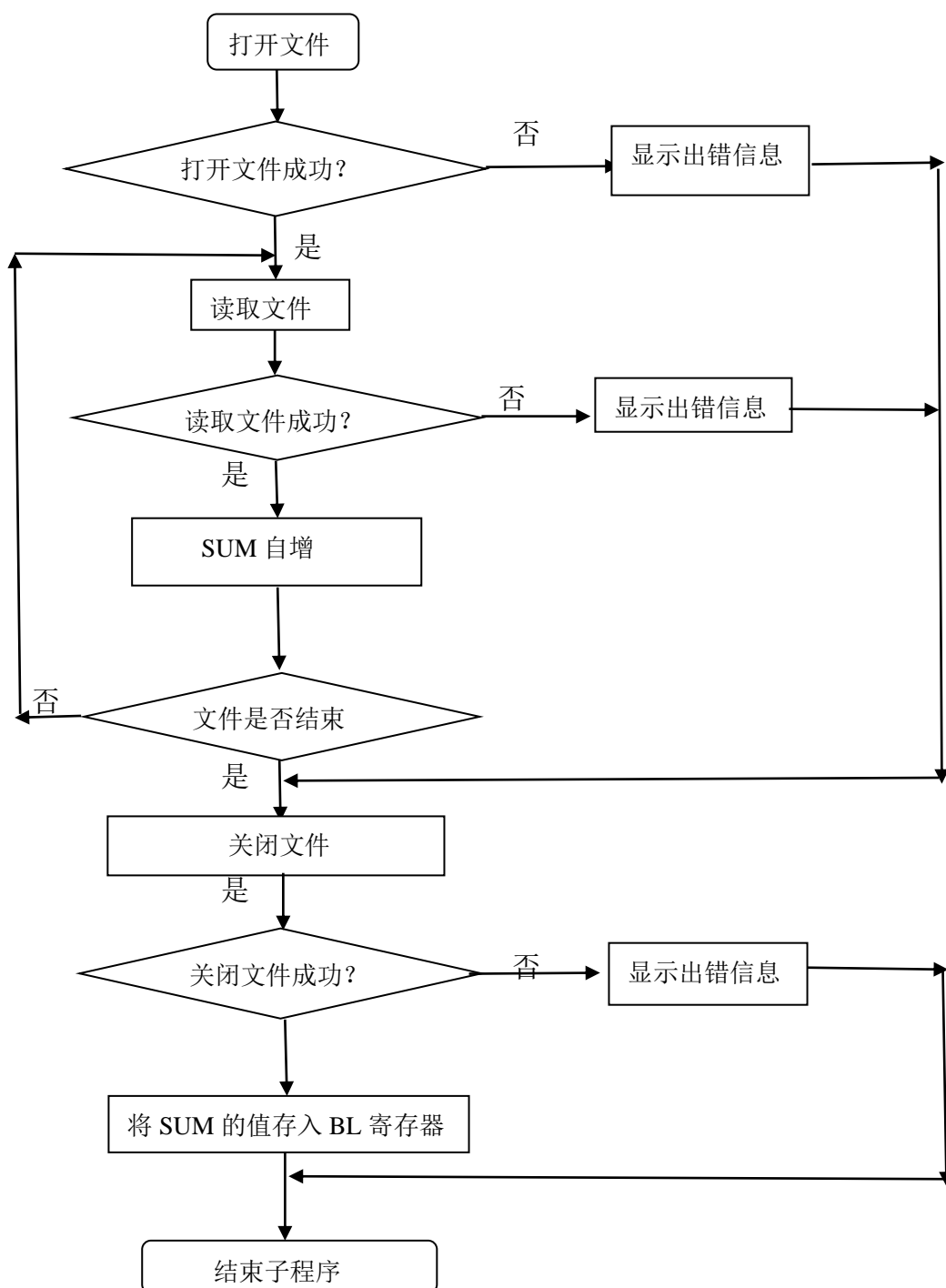
子程序名：COUNT_NUM

子程序功能：打开文件并读取文件中所有数据，计算文件中存入的学号个数 N

子程序的入口参数：FNAME, SUM

子程序的出口参数：BL 寄存器中存放的学号个数 N

所占用的寄存器：AX, BX, CX, DX



第 2 个子程序:

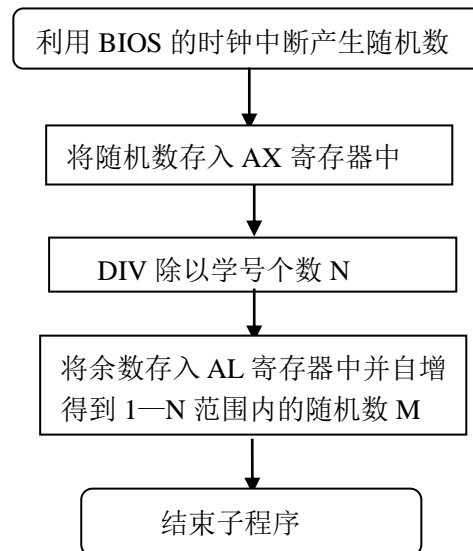
子程序名: GENER_NUM

子程序功能: 生成 1-N 之间的随机数 M

子程序的入口参数: BL 寄存器中存放的学号个数 N

子程序的出口参数: AL 寄存器中存放的随机数 M

所占用的寄存器: AX, BL, DX



第 3 个子程序:

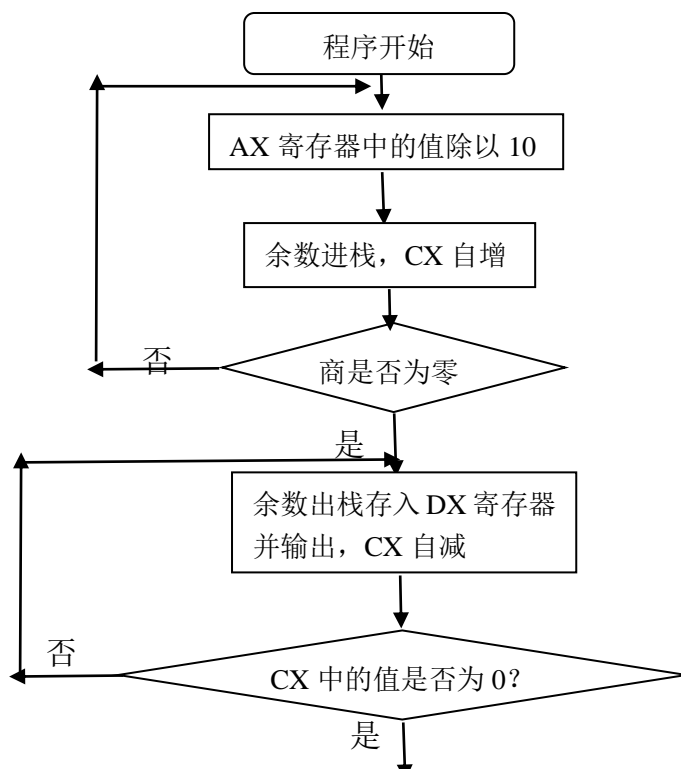
子程序名: TRANS_NUM

子程序功能: 十六进制转十进制并以 ASCII 码输出

子程序的入口参数: AL 寄存器中存放的随机数 M

子程序的出口参数: 无

所占用的寄存器: AX, BX, CX, DX



程序结束

第 4 个子程序：

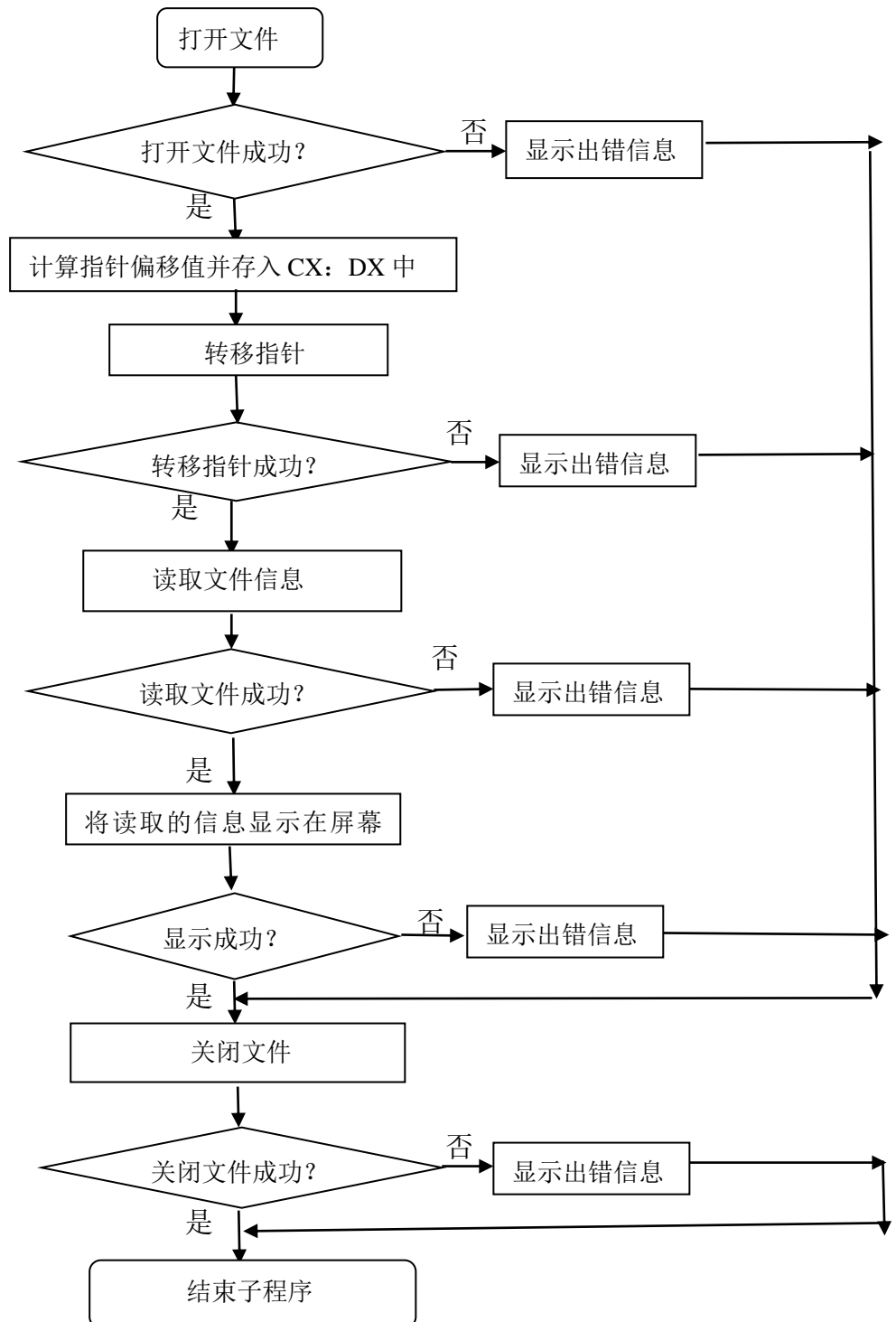
子程序名：PRINT_NUM

子程序功能：找到文件中的第 M 个学号并显示在屏幕上

子程序的入口参数：AL 寄存器中存放的随机数 M

子程序的出口参数：无

所占用的寄存器：AX, BX, CX, DX



三、调试说明

1. 调试情况

上机时遇到的问题 1: 文件指针转移时对偏移值理解计算错误, 直接将随机数赋值给 DX 寄存器; 解决办法 1: AL 寄存器存放的随机数乘以 1 条记录的字节数, 所得乘积赋值给 CX:DX, 用于记录文件指针偏移值。

上机时遇到的问题 2: 正确的偏移值存入 CX:DX 中后, 无论产生的随机数是几, 输出的学号始终为同一个; 解决办法 2: CX、DX 寄存器先清零。

上机时遇到的问题 3: BIOS 时钟中断指令产生的随机数不在所需范围内; 解决办法 3: BIOS 时钟中断指令产生的随机数除以总数 N 后取余数, 余数加 1 即为 1-N 范围内的随机数。

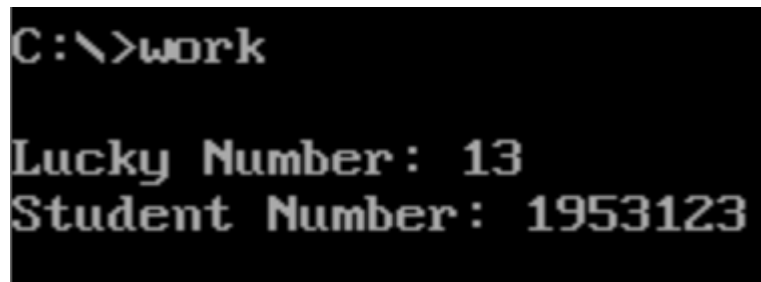
程序设计一些技巧的总结: 程序设计时将一整个程序划分为几个子程序分别完成相应的功能, 互不干涉, 能有效提高写程序的效率; 在程序整体完成后, 再总体浏览一遍程序, 把一些重合的部分重新组合成一个子程序, 提高程序的可读性和质量; 程序运行出现错误时, 在调试时可以一个子程序一个子程序调试, 确定了错误出现的大致范围之后再单步调试, 或者对比错误答案和正确答案的区别来发现错误出现的原因。

2. 程序运行情况及结果分析: mystu.txt 文件第 1 个学号为: 1950122, 第 21 个学号为 1951032, 第 15 个学号为 1854028, 第 13 个学号为 1953123. 程序运行结果正确, 符合题目要求。

```
C:\>work  
  
Lucky Number: 1  
Student Number: 1950122
```

```
C:\>work  
  
Lucky Number: 21  
Student Number: 1951032
```

```
C:\>work  
  
Lucky Number: 15  
Student Number: 1854028
```



```
C:\>work

Lucky Number: 13
Student Number: 1953123
```

四、使用说明

1. 程序运行的软硬件环境、适用范围：

8086 汇编语言程序需要在 DOS 环境下运行，对于 32 位的 Windows7 可以直接进入 DOS 环境，对于 64 位 Windows7 或者 Windows8/10，则需先下载并安装 DOSBox，用以模拟 DOS 环境。汇编程序 MASM 是用于汇编的系统软件，这里可以使用 Microsoft 公司推出的宏汇编程序 MASM (Micro Assembler) 5.10 版本。

本程序需要打开文件并读取文件中的信息，在此程序中直接使用附件中的 mystu.txt 文件，因此本程序适用于文件名为 mystu.txt 的文件。如需读取其它文件中的信息，请在程序数据段定义中更改变量名 FNAME 对应的 ASCIZ 串。

2. 程序的使用方法、调试方法、操作步骤：

在 DOS 环境下完成对程序的汇编 (masm)、连接 (link) 即可生成可执行文件，直接运行可执行文件即可得到相应结果，无需输入信息。

16 位环境下可使用 DOS 下的调试工具 DEBUG 对程序进行调试，运行 DEBUG 后显示的提示符是 “-”，此时可输入各种命令如：查看和修改寄存器内容的命令 R、显示内存单元内容的命令 D、输入汇编指令的命令 A、反汇编命令 U、执行命令 G、单步执行命令 T、退出 DEBUG 命令 Q 等。

3. 出错信息的含义及注意事项等：

出错信息含义如下：

OPEN ERROR: 打开文件错误

CLOSE ERROR: 关闭文件错误

READ ERROR: 读取文件错误

WRITE ERROR: 写文件错误 (文件信息显示到屏幕错误)

POINTER ERROR: 文件指针转移错误

其它文字提示含义如下：

Lucky Number: 幸运数字

Student Number: 中奖学号

注意事项：本程序需要打开文件并读取文件中的信息，在此程序中直接使用附件中的 mystu.txt 文件，因此本程序适用于文件名为 mystu.txt 的文件。如需读取其它文件中的信息，请在程序数据段定义中更改变量名 FNAME 对应的 ASCIZ 串。