（正文格式：宋体，小四，首行缩进2汉字，单倍行距）

1. 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。

1. 设计说明
2. 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。
3. 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。

1. 程序框图如下：

程序开始

调用第1个子程序

调用第2个子程序

输出文字提示

调用第3个子程序

输出文字提示

调用第4个子程序

程序结束

1. 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

打开文件

否

显示出错信息

打开文件成功？

是

读取文件

显示出错信息

读取文件成功？

否

是

SUM自增

文件是否结束

否

是

关闭文件

是

关闭文件成功？

显示出错信息

否

将SUM的值存入BL寄存器

结束子程序

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

利用BIOS的时钟中断产生随机数

将随机数存入AX寄存器中

DIV除以学号个数N

将余数存入AL寄存器中并自增

得到1—N范围内的随机数M

结束子程序

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

程序开始

AX寄存器中的值除以10

余数进栈，CX自增

商是否为零

否

是

余数出栈存入DX寄存器并输出，CX自减

CX中的值是否为0？

否

是

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

打开文件

打开文件成功？

否

显示出错信息

是

计算指针偏移值并存入CX：DX中

转移指针

否

显示出错信息

转移指针成功？

是

读取文件信息

否

显示出错信息

读取文件成功？

是

将读取的信息显示在屏幕上

显示成功？

否

显示出错信息

是

关闭文件

关闭文件成功？

否

显示出错信息

是

结束子程序

1. 调试说明
2. 调试情况

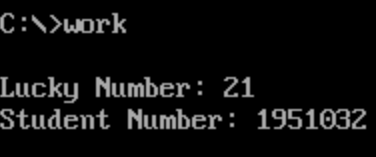
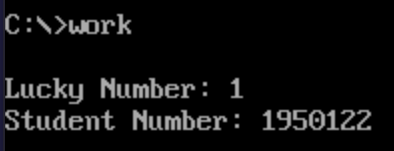
上机时遇到的问题1：文件指针转移时对偏移值理解计算错误，直接将随机数赋值给DX寄存器；解决办法1：AL寄存器存放的随机数乘以1条记录的字节数，所得乘积赋值给CX:DX，用于记录文件指针偏移值。

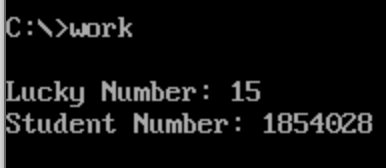
上机时遇到的问题2：正确的偏移值存入CX:DX中后，无论产生的随机数是几，输出的学号始终为同一个；解决办法2：CX、DX寄存器先清零。

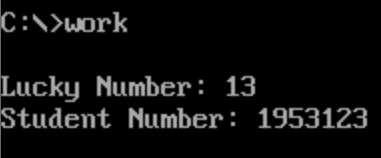
上机时遇到的问题3：BIOS时钟中断指令产生的随机数不在所需范围内；解决办法3：BIOS时钟中断指令产生的随机数除以总数N后取余数，余数加1即为1-N范围内的随机数。

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。

1. 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。







1. 使用说明
2. 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。

1. 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。

1. 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。