

重要：程序文件名对应作业题号!!!

作业 1：通过改造 Hello,world 程序实现：

在屏幕上显示自己的联系方式，包括自己姓名（用汉语拼音表示），学号，联系电话和 email。显示格式如下：

Name: lijianming

Number: XXXXXX

Phone: 84708549

Email: lijm@dlut.edu.cn

运行结果：

```
C:\DESKTOP\DASM>1.com
Name:Sung Lientao
Number:201685030
Phone:15542386793
Email:1063539881@qq.com
```

作业 2：写出下列数据在计算机的十六进制表示形式：

1. 无符号整数 100: $0110\ 0100B = 64H$
2. 有符号整数 -100: $1001\ 1100B = 9CH$
3. 字符串 'DLUT': $44\ 4C\ 55\ 54$

075A:0100

44 4C 55 54

DLUT

作业 3：使用 BIOS 调用实现：

1. 重新改造第一次作业，在屏幕上显示自己的信息，每一行显示为不同的颜色
2. 在屏幕上显示五角星图案，背景色和星星的颜色、图案自己设计

运行结果：

```
Name:Sung Lientao
Number:201685030
Phone:15542386793
Email:1063539881@qq.com
```



作业 4: 使用 8 位二进制加法运算, 对 CF, OF, SF, ZF, PF, AF, 复位与置位情况, 各举一个例子说明。

3AH+7CH=B6H, 没有进位, CF=0

AAH+7CH=126H, 有进位, CF=1

3AH+7CH=B6H, 结果不是 0, ZF=0

84H+7CH=100H, 但由于只保存 8 位, 有溢出, 结果是 0, ZF=1

3AH+7CH=B6H, 最高位是 1, SF=0

84H+7CH=100H, 最高位是 0, SF=1

3AH+7CH=B6H=10110110B, 低八位中有 5 个 1, 是奇数, PF=0

4AH+7CH=C6H=11000110B, 低八位中有 4 个 1, 是偶数, PF=1

3AH+7CH=B6H, 两个正数相加得到负数, OF=0

AAH+7CH=126H, 正数相加仍然得到正数, 没有溢出, CF=1

说明: 只要符号不同的数进行运算, 都不会产生溢出。

33H + 78H = ABH, 低四位 3, 加上低四位 8, 进位为 0, AF=0

39H + 78H = B1H, 低四位 9, 加上低四位 8, 进位为 1, AF=1

作业 5: 用 VC 写一段程序, 通过指针操作数组和字符串。分析其反汇编程序, 用汇编语言的理解, 对指针等数据的变换进行解释说明。

```
int aa[5] = {0, 1, 2, 3, 4};
```

开了 20 个字节的连续存储空间, 每个数以双字的形式存储。

```
char *string = "DLUT";
```

与数组定义类似, 每个字符也以双字形式存储, string 表示 DLUT 的首地址。

```
char c='a';
```

开了一个字节存 41h。

```
char *p = &aa[0];
```

一级指针, 在内存中接着开 4 个字节的存储空间, 取 aa[0]的地址, 放入。

```
char **pc = &p;
```

二级指针, pc 中存的是指针变量 p 的地址, pc 仍然占 4 个字节。

```
char c1 = **pc;
```

将 pc 所指的变量所指的变量内容 aa[0]存入 c1。

```
*p = i;
```

将变量 i 在内存中代表的数据取出, 放入 p 指针所指的内容中。

```
string = string+3;
```

将 string 所代表的地址加 3, 指针的加减法 =个数 * sizeof(类型)。

数组的定义

```
5:    int aa[] = {0, 1, 2, 3, 4};
0040D428  mov     dword ptr [ebp-14h],0
0040D42F  mov     dword ptr [ebp-10h],1
0040D436  mov     dword ptr [ebp-0Ch],2
0040D43D  mov     dword ptr [ebp-8],3
0040D444  mov     dword ptr [ebp-4],4
```

一、二级指针的定义与赋值

```
5:    int n = 2;
0040D428  mov     dword ptr [ebp-4],2
6:    int *p1 = &n;
0040102F  lea     eax,[ebp-4]
00401032  mov     dword ptr [ebp-8],eax
7:    int **p2 = &p1;
00401035  lea     ecx,[ebp-8]
00401038  mov     dword ptr [ebp-0Ch],ecx
8:    *p1 = 5;
0040103B  mov     edx,dword ptr [ebp-8]
0040103E  mov     dword ptr [edx],5
9:    **p2 = 3;
00401044  mov     eax,dword ptr [ebp-0Ch]
00401047  mov     ecx,dword ptr [eax]
00401049  mov     dword ptr [ecx],3
```

指针的加法

```
7:    p++;
0040D47A  mov     edx,dword ptr [ebp-28h]
0040D47D  add     edx,4
0040D480  mov     dword ptr [ebp-28h],edx
```

作业 6：使用 DOS 文件操作系统调用编程实现：

- 1. 编写 writeInfo.com, 执行该程序, 在 c 盘根目录下创建 myinfo.txt 文件, 写入自己的信息
- 2. 编写 InsertInfo.com, 执行该程序, 计算已经创建的 myinfo.txt 文件长度, 例如文件长度为 30 个字节, 在 myinfo.txt 文件开始处写入信息: filesize: 30

代码 6-1-1.com 是创建文件, 6-1-2.com 是写入自己的信息

代码 6-2.com 是写入 filesize 的程序, DOS 下不支持超过 8 位的文件名, 所以做了一点修改。

第 2 问示意图如右图所示,

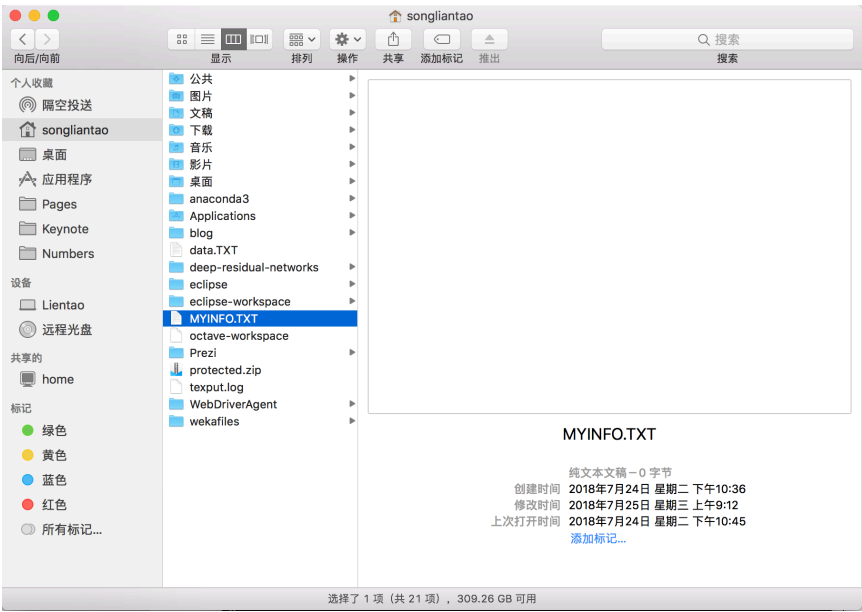
代码思路描述:

- 1) 在内存中开辟出一个区域存入 filesize
- 2) 利用移动文件指针统计文件字符个数, 个数以十六进制的形式存在 AX 寄存器中
- 3) 将 16 进制数每次除以 10 取余数, 压入栈中
- 4) 倒序弹出余数, 加上 '0' 的 ASCII 码存在 filesize 的后面
- 5) 为了美观, 在输出 filesize:80 完成后加入 0dh, 0ah
- 6) 读文件, 将文件内容读在 0ah 后
- 7) 从 filesize 开始将内存中数据重新写回文件

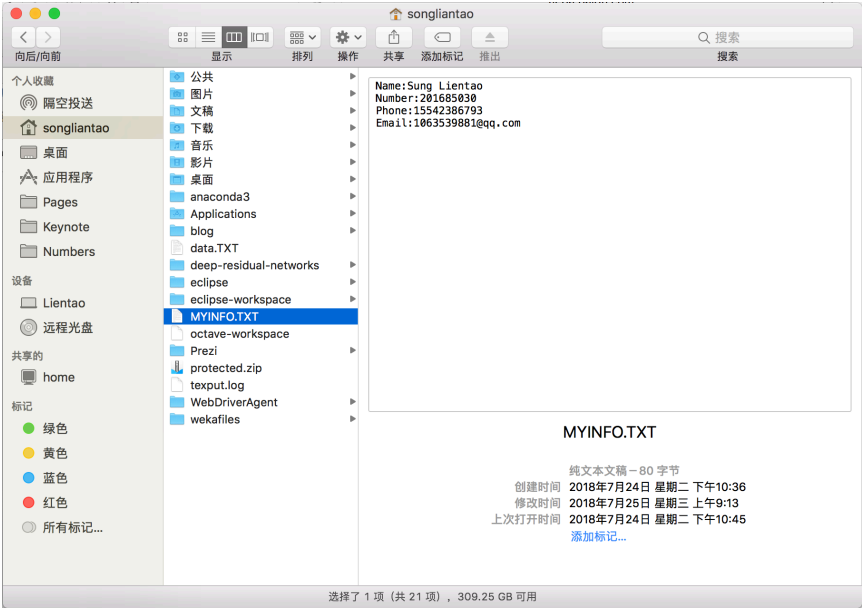
⋮
m
a
N
回车
换行
0
8
:
e
⋮
i
f

运行结果截图:

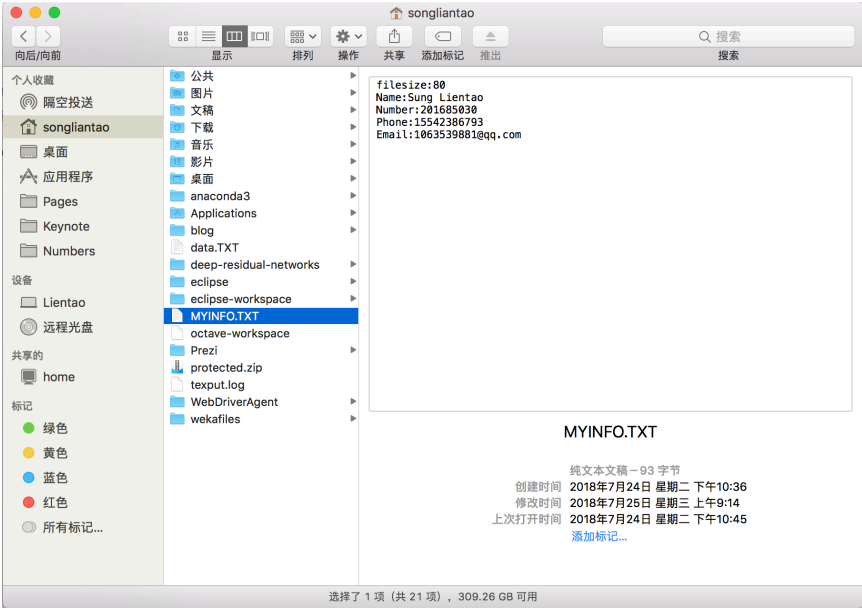
执行 6-1-1.com



执行 6-1-2.com



执行 6-2.com



作业 7：使用算数运算和位操作指令设计一套字符串加密解密算法，实现：

1. 加密：输入待加密明文字符串，比如“DLUTASM”。输出加密后的密文字符串，比如“897HEGH”
2. 解密：输入加密后的密文，比如“897HEGH”，输出加密前的明文，比如“DLUTASM”

思路：

利用 int 中断 02h 读入一个字符串，写在缓冲区后面，读入的字符个数写在缓冲区第二个字节中，记录下来。接下来对每一位利用异或进行加密，将加密后的字符串输出。同理，对每一位再次异或同一个字符串，就可以完成解密的工作。

输入：yidingyaoshang90fen

```
C:\DESKTOP\DASM>7.com
|ú«úñ|½ñ|ó½ñ|≤¼»ñ
yidingyaoshang90fen
```

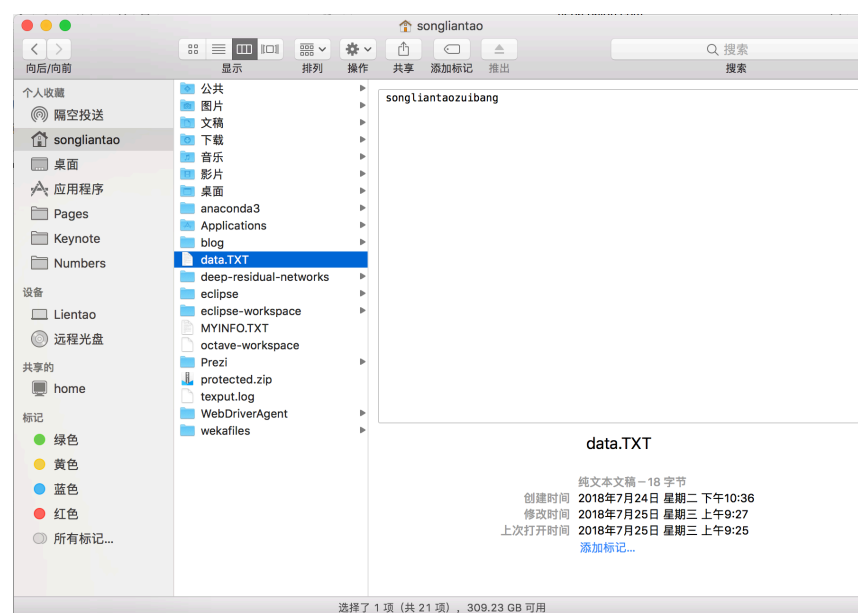
作业 8：

综合使用 DOS 调用的 0A 号功能和文件操作实现如下功能：

1. 程序运行，等待用户键盘输入，输入结束后将用户输入的内容写到文本文件 data.txt 中。
比如用户输入 ABC,打开 data.txt 文件，应能显示 ABC。

做完 6-2 题，这题就异常简单。需要说明的是，缓冲区设置能接受 18 个字节的输入。

```
C:\DESKTOP\DASM>8.com
songliantaozuibang
```



作业 9: 用 Debug 重新实现第一次课作业, 生成 com 文件, 要求个人信息的偏移位置为 150H。

思路:

首先用 e 命令将个人信息的 ASCII 码预先放在内存偏移位置 150H 处, 以 \$ 符结束。

```
e 150
```

```
4e 61 6d 65 3a 53 75 6e 67 20 4c 69 65 6e 74 61 6f 0d 0a
```

```
4e 75 6d 62 65 72 3a 32 30 31 36 38 35 30 33 30 0d 0a
```

```
50 68 6f 6e 65 3a 31 35 35 34 32 33 38 36 37 39 33 0d 0a
```

```
45 6d 61 69 6c 3a 31 30 36 33 35 33 39 38 38 31 0d 0a 24
```

然后执行 a 命令输入汇编程序代码:

```
a
```

```
mov dx, 150
```

```
mov ah, 9
```

```
int 21 ; 显示以 "$" 为结尾的字符串
```

```
int 20 ; 结束程序
```

最后, 保存文件并退出。

```
r cx
```

```
cx 0000: 200
```

```
n 4.com
```

```
w
```

```
q
```