重庆师范大学实验报告



学 号：2017051604047

姓 名： 秦秀元

课程名称： 深入理解计算机系统

学科专业：17级软件工程

所在学院：计算机与信息科学学院

重庆师范大学

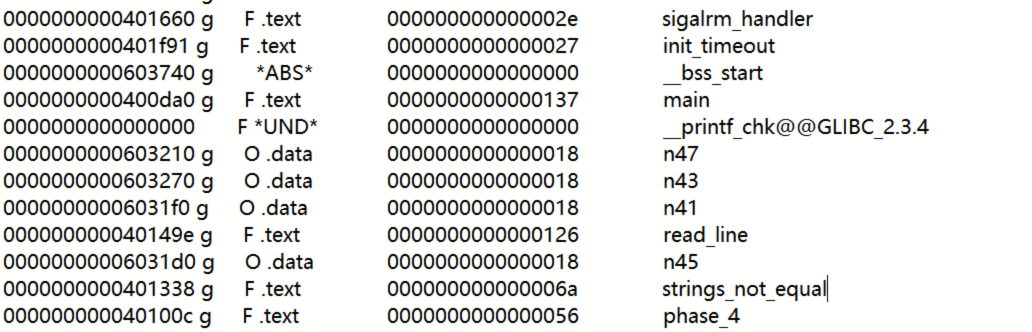
2018年12月25号

1. **实验目的及要求**

利用反汇编器objdump和调试器gdb完成二进制炸弹实验，该实验总共有六关，其中前二关为必做，后四关为选做（此外，还有一隐蔽关）。用文档形式描述分析和拆解过程，并在文档中附上关键步骤截图（自己手写拍照图片）。

本实验评分说明：前二关70分，第三关20分，第四关10分，第五关5分，第六关5分

1. **分析和拆解过程**

首先：通过objdump -d bomb得到反汇编代码，然后通过objdump -t bomb查看符号表，如下图所示：

通过gdb bomb进行调试。

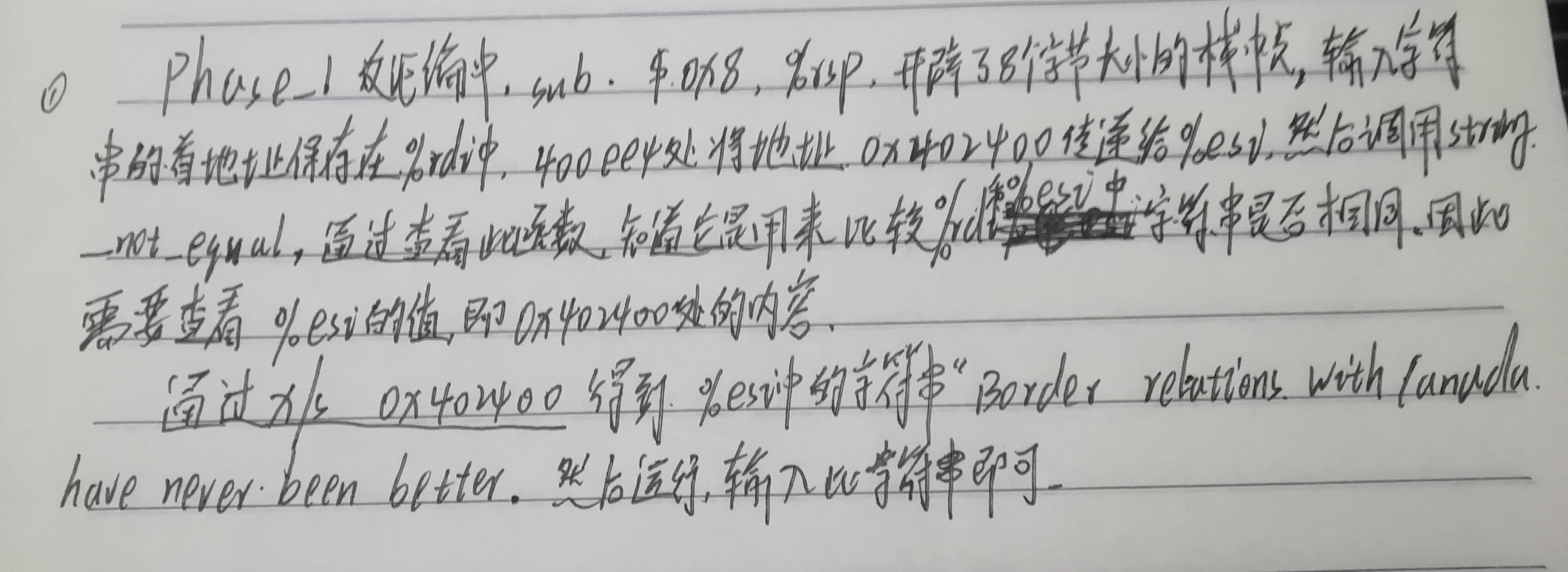
1. phase\_1:

首先查看第一关phase\_1函数的反汇编代码,

400ee4: be 00 24 40 00 mov $0x402400,%esi 将0x402400的值赋给%esi

400ee9: e8 4a 04 00 00 callq 401338 <strings\_not\_equal>

400eee: 85 c0 test %eax,%eax 比较字符串是否相等



1. phase\_2:

首先查看它的反汇编代码，

400f05: e8 52 05 00 00 callq 40145c <read\_six\_numbers> 调用这个函数

400f0a: 83 3c 24 01 cmpl $0x1,(%rsp) 比较1与%rsp

......

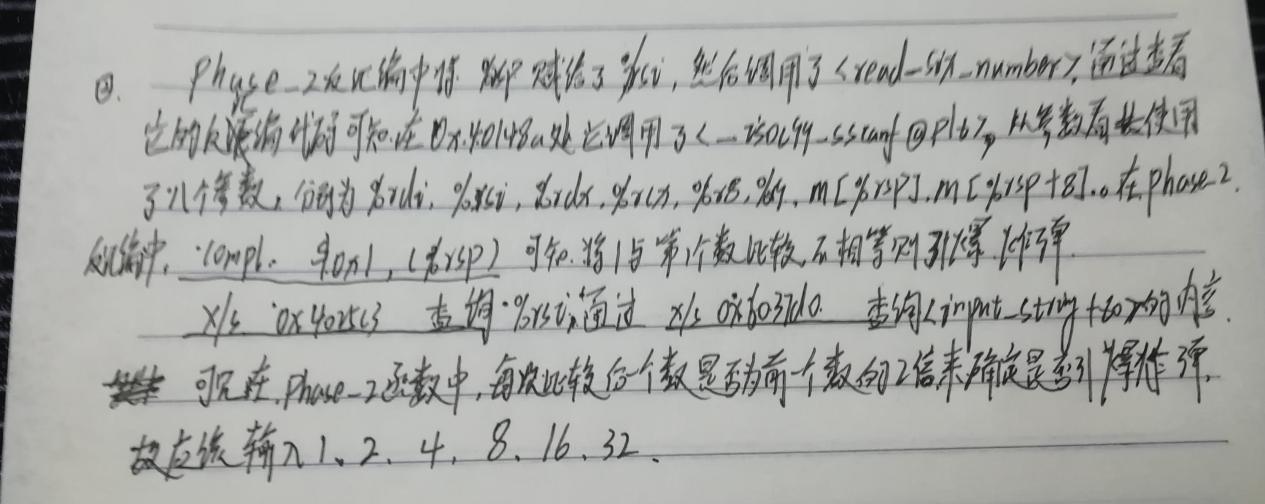
400f1a: 01 c0 add %eax,%eax %eax \*2

400f1c: 39 03 cmp %eax,(%rbx) 比较 %eax与 (%rbx)

......

40148a: e8 61 f7 ff ff callq 400bf0 <\_\_isoc99\_sscanf@plt>

40148f: 83 f8 05 cmp $0x5,%eax 至少输入6个数字



1. phase\_3:

先观察反汇编代码，

400f51: be cf 25 40 00 mov $0x4025cf,%esi 将0x4025cf的值放入%esi

通过gdb x/s 0x4025cf 得到 0x4025cf: "%d %d" 需要输入两个数

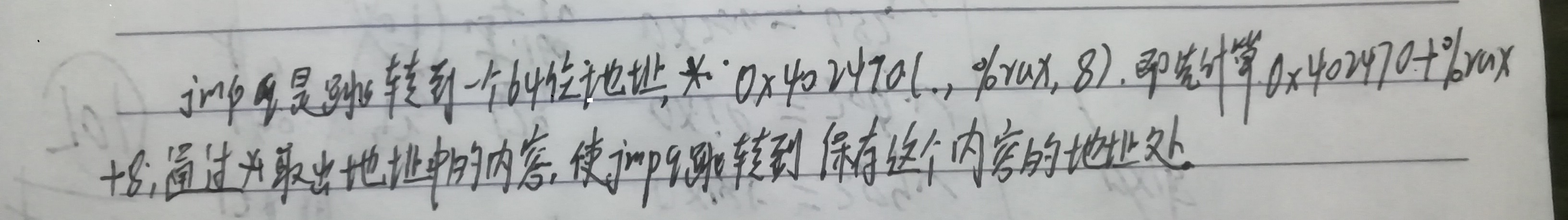
400f56: b8 00 00 00 00 mov $0x0,%eax 将0赋给%eax

400f60: 83 f8 01 cmp $0x1,%eax 将%eax的值与1比较

400f6a: 83 7c 24 08 07 cmpl $0x7,0x8(%rsp) 将第一个输入数与7比较

所以第一个输入数取值范围为0~7之间；

400f75: ff 24 c5 70 24 40 00 jmpq \*0x402470(,%rax,8)



400f7c: b8 cf 00 00 00 mov $0xcf,%eax 与第一个数匹配的第二个输入数

400f81: eb 3b jmp 400fbe <phase\_3+0x7b>

......

后面根据%rax的取值跳到不同的位置，可以得出结论为switch语句，

通过(gdb) x/8gx 0x402470 得到跳转表的跳转地址

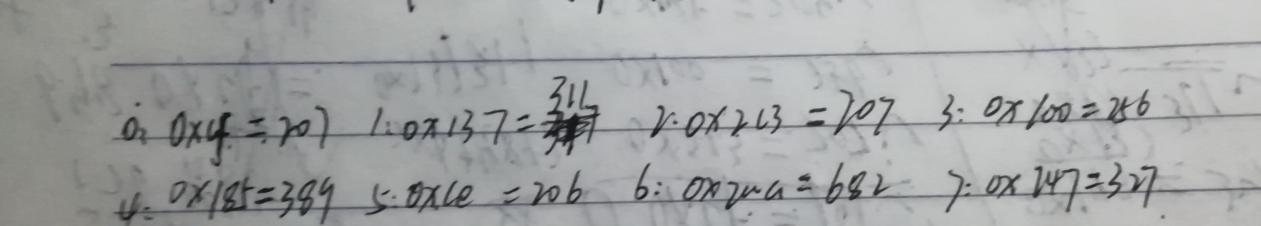
0x402470: 0x0000000000400f7c 0x0000000000400fb9

0x402480: 0x0000000000400f83 0x0000000000400f8a

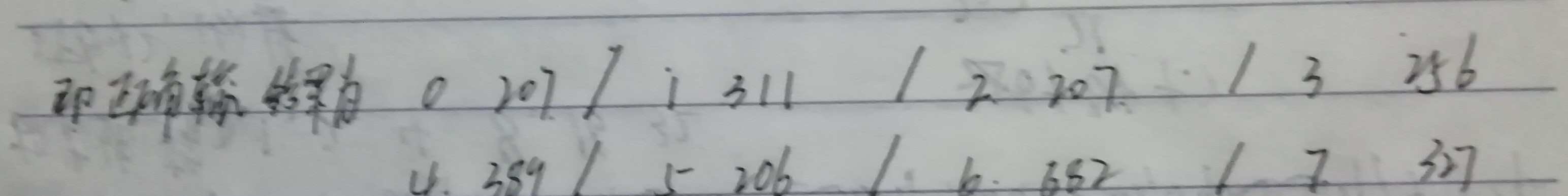
0x402490: 0x0000000000400f91 0x0000000000400f98

0x4024a0: 0x0000000000400f9f 0x0000000000400fa6

可以知道跳转到每一处都会对%eax进行赋值，并与第二个输入数比较，



所以输入结果为：



1. phase\_4:

首先查看它和func4的反汇编代码，

40101a: be cf 25 40 00 mov $0x4025cf,%esi

通过gdb x/s 0x4025cf 得到 0x4025cf: "%d %d" 需要输入两个数

40103a: ba 0e 00 00 00 mov $0xe,%edx 可以知道%edx初始值为14

40103f: be 00 00 00 00 mov $0x0,%esi 可以知道%esi的初始值为0

......

400fdf: 8d 0c 30 lea (%rax,%rsi,1),%ecx

400fe2: 39 f9 cmp %edi,%ecx

......

400ff2: b8 00 00 00 00 mov $0x0,%eax

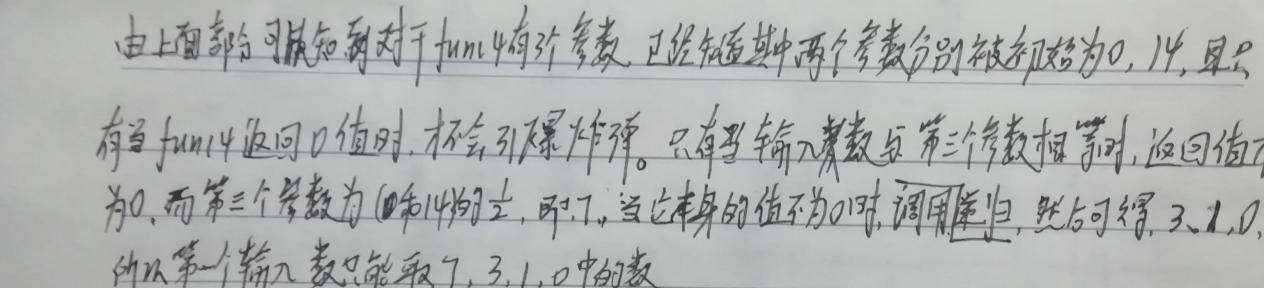
400ff7: 39 f9 cmp %edi,%ecx

......

401003: 8d 44 00 01 lea 0x1(%rax,%rax,1),%eax

......

这一段反汇编代码使用了递归和条件语句，所以有



401051: 83 7c 24 0c 00 cmpl $0x0,0xc(%rsp)可以知道第二个数必为0

因此正确输入结果为：7 0或3 0或1 0或0 0.

1. phase\_5:

首先查看反汇编代码，

40107a: e8 9c 02 00 0 callq 40131b <string\_length>调用string\_length计算字

40107f: 83 f8 06 cmp $0x6,%eax 比较输入字符串长度是否为6

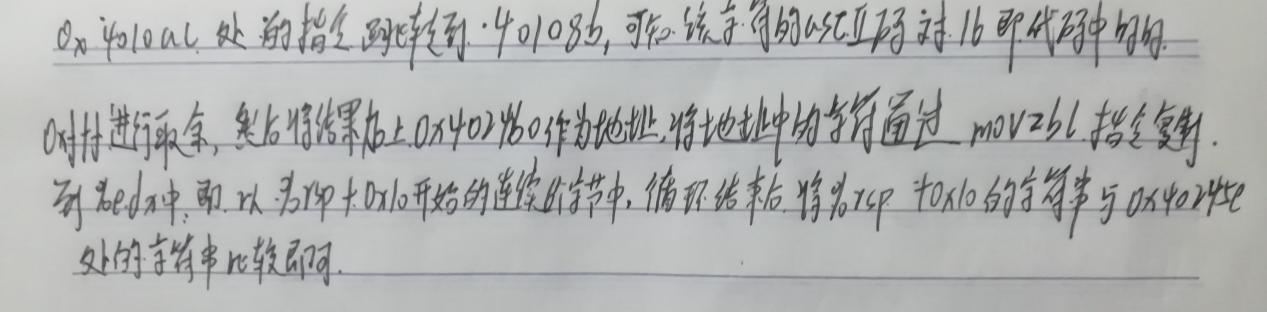
......

401099: 0f b6 92 b0 24 40 00movzbl 0x4024b0(%rdx),%edx

4010ae: c6 44 24 16 00 movb $0x0,0x16(%rsp)

4010b3: be 5e 24 40 00 mov $0x40245e,%esi

4010b8: 48 8d 7c 24 10 lea 0x10(%rsp),%rdi



4010b3: be 5e 24 40 00 mov $0x40245e,%esi

通过(gdb) x/s 0x40245e查看0x40245e处的值 0x40245e: "flyers"

通过(gdb) x/32c 0x4024b0 以字符形式显示内容

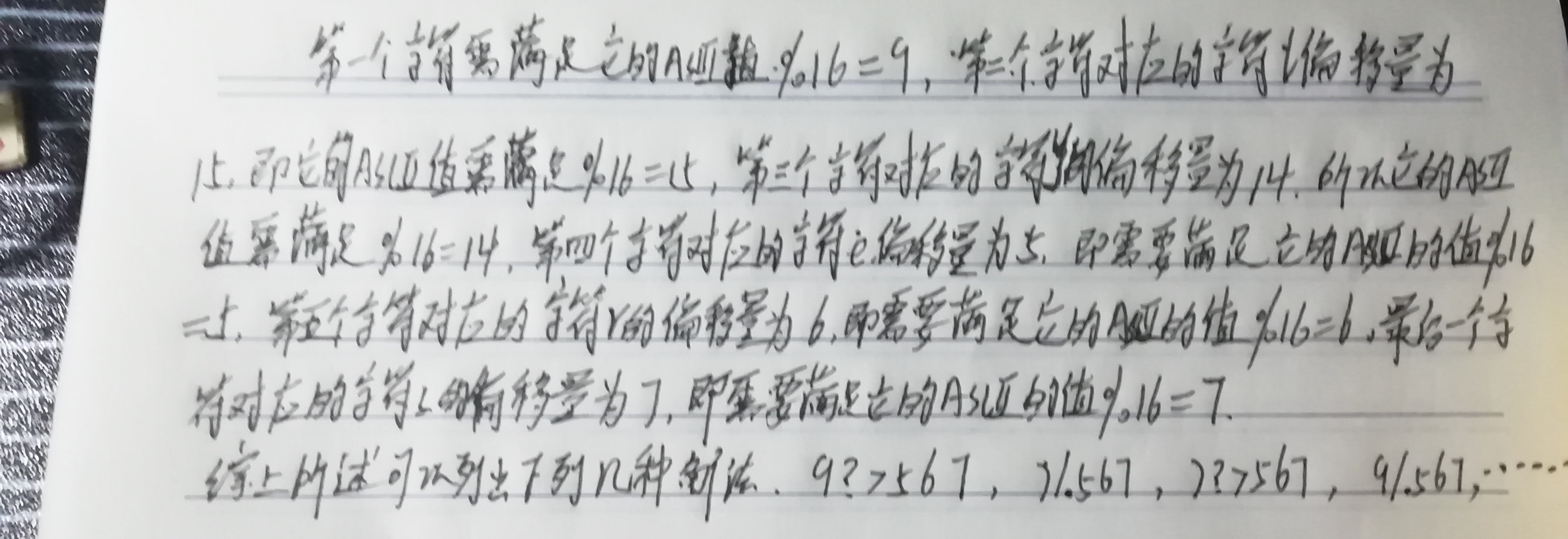
0x4024b0 <array.3449>:109 'm' 97 'a' 100 'd' 117 'u' 105 'i' 101 'e' 114 'r' 115 's'

0x4024b8 <array.3449+8>: 110 'n' 102 'f' 111 'o' 116 't' 118 'v' 98 'b' 121'y' 108 'l'

0x4024c0: 83 'S' 111 'o' 32 ' ' 121 'y' 111 'o' 117 'u' 32 ' ' 116 't'

0x4024c8: 104 'h' 105 'i' 110 'n' 107 'k' 32 ' ' 121 'y' 111 'o' 117 'u'

然后对每个查找从0x4024b0处开始的相应字符的地址，减去0x4024b0,再换成字符即可，其中与第一个字符对应的字符f的偏移值为9，即第一个字符需要满足%16=9，



第一个字符可以从9，大小写i，大小写y，）中选取；

第二个字符可以从/，？，大小写o，下划线中选取；

第三个字符可以从句号，>，大小写n，^中选取；

第四个字符可以从%，5，大小写e，大小写u中选取；

第五个字符可以从&，6，大小写f，大小写v中选取；

第六个字符可以从单引号’，7，大小写g，大小写w中选取。

略：

1. phase\_6：
2. phase\_7：

注意：

1、不要抄袭

2、**截止日期2019年1月8日**