

重 庆 大 学

学 生 实 验 报 告

实验课程名称 汇编语言程序设计

开课实验室 DS1502 机房

学院 大数据与软件学院 年级 2023 专业班 软件工程 01 班

学 生 姓 名 学 号

开 课 时 间 2022 至 2023 学年第 1 学期

总 成 绩	
教师签名	陈蜀宇

大数据与软件工程学院 制

《汇编语言程序设计》实验报告

开课实验室： DS1502 机房

2022 年 10 月 30 日

学院	大数据与软件学院	年级、专业、班	24 级软工 01 班	姓名		成绩	
课程名称	汇编语言程序设计	实验项目名称	实验五：循环、分支程序设计、子程序设计	指导教师	陈蜀宇		
教师评语	教师签名：陈蜀宇 2022 年 10 月 日						

一【实验目的】

- 熟悉汇编语言循环分支程序设计
- 中断调用程序设计

二【实验环境】

- PC 微机
- DOS 操作系统或 Windows 操作系统
- MASM.EXE, LINK.EXE, DEBUG.COM 或宏汇编集成环境

Dosbox.exe 的使用方法

1. 安装。
2. 运行 dosbox.exe 程序。
3. 在输入框 z:\> 状态下输入：
z:\> mount c d:\masm
说明：“z:\>”这是提示符，“c”作为虚拟 c 盘，“d:\masm”是要虚拟的文件夹位置；简洁的讲，把要虚拟的文件夹位置换掉上面的 d:\masm。
4. 在刚才的提示符下输入 c:，这样就切换到虚拟的 c 盘，也就是你要的目录。
5. 按照 dos 操作系统环境要求运行 c: 目录下的 masm.exe, link.exe, debug.exe 等软件。如：

```
c:\> masm 123.asm  
c:\> LINK 123.obj  
c:\> debug 123.exe
```

三【预备知识】

- 常用汇编指令
- 循环程序设计
- 分支程序设计
- 中断调用（DOS 系统功能调用 INT 21H）

四【实验内容】

- （1）调试教材 P170 例 5.6。

说明：体会非固定次循环的特点。区分出循环体、循环条件、循环变换条件、初始化内容、循环结束后续相关内容。

(2) 编写一个程序。

已知数组 A 包含 15 个互不相等的整数，数组 B 包含 10 个互不相等的整数。试编制一程序，把既在 A 中又在 B 中出现的整数存放于数组 C 中。然后将数组 C 逐个显示出来。

要求：要写流程图，代码必须规范，有必要的注释。

说明：数组表示可参考教材例 5.5。

(3) 编写一个程序，接受从键盘输入的 10 个十进制数字，输入回车符则停止输入，然后将这些数字加密后(用 XLAT 指令变换)存入内存缓冲区 BUFFER。加密表为：

输入数字：0, 1, 2, 3, 4, 5, 6, 7, 8, 9

密码数字：7, 5, 9, 1, 3, 6, 8, 0, 2, 4

要求：要写流程图，用子程序实现加密，代码必须规范，有必要的注释。

五【实验步骤】

一.

源代码:

```
dataarea segment
    buffer db 80 dup(?)
    flag    db ?
dataarea ends

prognam segment
main proc far
    assume cs:prognam, ds:dataarea
    start:///初始化内容
    push ds
    sub ax, ax
    push ax
    mov ax, dataarea
    mov ds, ax
    lea bx, buffer
    mov flag, 0

    next:///循环体
    mov ah, 01
    int 21h
    test flag, 01h:///循环条件
    jnz follow
    cmp al, 20h
    jnz exit
    mov flag, 1///循环变换条件
    jmp next

follow:
    cmp al, 20h
```

```

jz exit
mov [bx], al
inc bx
jmp next

exit:///循环结束
ret

main endp
prognam ends
end start

```

运行:

<pre> 0 Warning Errors 0 Severe Errors C:\>link lab50 Microsoft (R) Overlay Linker Version 3.60 Copyright (C) Microsoft Corp 1983-1987. All rights reserved. Run File [LAB50.EXE]: List File [NUL.MAP]: Libraries [.LIB]: LINK : warning L4021: no stack segment C:\>lab50 aasdwe C:\>lab50 j C:\>_ </pre>	<pre> test flag,01h jnz follow cmp al,20h jnz exit mov flag,1 jmp next follow: cmp al,20h jz exit mov [bx],al inc bx jmp next exit: ret main endp prognam ends </pre>
--	--

源代码:

```
data segment
    x db 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
    y db 0, 1, 2, 3, 40, 5, 60, 70, 80, 90///初始化原数组
    c db 10 dup(?)
    count dw 0
data ends

code segment
    assume cs:code, ds:data
start:
    mov ax, data
    mov ds, ax
    mov bx, 0
    mov count, 0
x_circulate:////外层循环，用于遍历 x 数组中的每一个数值
    mov al, x[bx]///关于数组的遍历，这里参考教材使用了类似下标的遍历办法
    mov si, 0

y_circulate:////内层循环，用于遍历 y 数组中的每一个数值
    mov dl, y[si]
    cmp al, dl
    je same
    inc si
    cmp si, 10
    je x_next
    jmp y_circulate

same:////如果相同则放到 c 数组中
    lea si, c
    add si, [count]
    mov [si], al
    inc [count]
```

```
    jmp x_next
```

```
x_next:///外层循环的下一个值
```

```
    inc bx
    cmp bx,10
    jne x_circulate
```

display:///由于 02h 只能显示 0-9 的 asc11 码表，所以将单个数值的每一位取出来并且压入栈内，除了除以 10，也可以采用循环移位，但是这里是除以十

```
    mov ax,data
    mov ds,ax
    mov bx,0
output_loop:
    mov al,c[bx]
    push bx///bx 用于计数，当它等于 count 时则退出 display
    mov ah,0
    mov cx,0
```

```
div_loop:///取位循环
```

```
    mov bl,10
    div bl
    cmp al,0
    je end_div
    inc cx
    push ax
    mov ah,0
    jmp div_loop
```

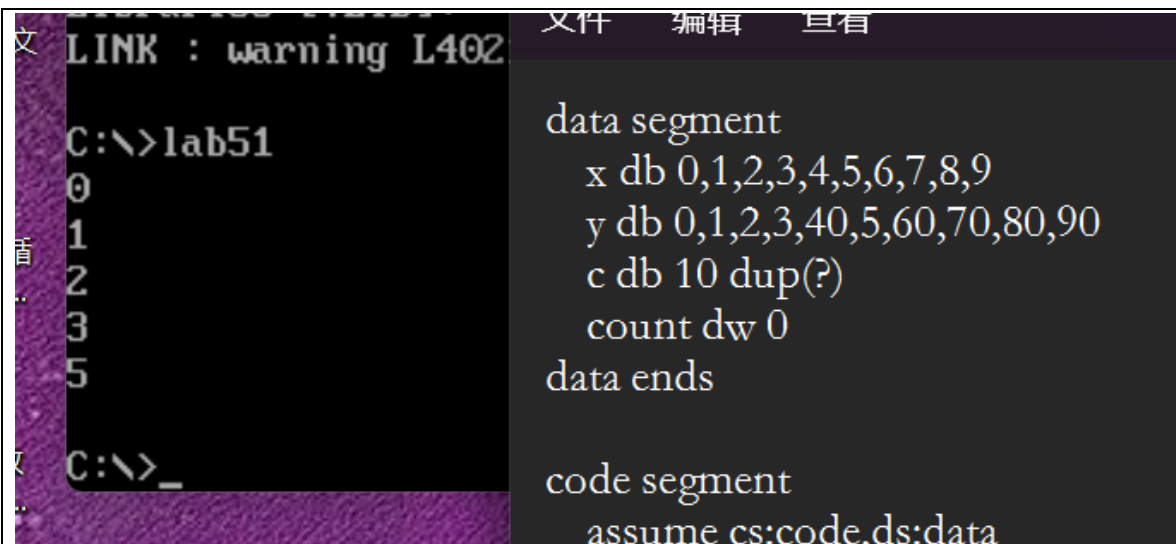
```
end_div:
```

```
    inc cx
    push ax
```

```
pop_loop:
    pop ax
    mov dl, ah
    add dl, 48///这里是将符号转化为 0-9
    mov ah, 02h
    int 21h
    dec cx
    cmp cx, 0
    jne pop_loop
    mov dl, 10
    mov ah, 02h
    int 21h
    pop bx
    inc bx
    cmp bx, [count]///遍历结束则跳出循环
    jne output_loop
    mov ah, 4ch
    int 21h

code ends
end start
```

运行:



三.

源代码:

```
data segment
```

```
buffer db 10 dup(?)
```

```
data ends
```

```
code segment
```

```
assume cs:code,ds:data
```

```
start:
```

```
mov ax,data
```

```
mov ds,ax
```

```
mov cx,10
```

```
input_number:
```

```
mov ah,01h
```

```
int 21h
```

```
cmp al,13///先判断是否为回车，是的话就退出
```

```
je exit
```

```
sub al,30h///由于在 encrypt 子程序中是和十进制数字比较，这里先将 al 减 48
```

```
call encrypt
```

```
add al,30h///在存储到 buffer 中时再加回来
```

```
lea si, buffer
```

```
mov [si], al
inc si
mov dl, al
mov ah, 02h
int 21h
mov dl, 32
mov ah, 02h
int 21h
loop input_number
```

encrypt proc near///由于没找到加密数字的规律，所以只能一一变换，这里用二分法索引数字，即先和 5 比较，然后根据大于或小于在和 3 或 7 比较...

```
cmp al, 5
jl min5
jg max5
mov al, 6
ret
min5:
cmp al, 3
jl min3
jg is4
mov al, 1
ret
max5:
cmp al, 7
jl is6
jg max7
mov al, 0
ret
is6:
mov al, 8
```

```
ret
max7:
cmp al,8
jg is9
mov al,2
ret
is9:
mov al,4
ret
is4:
mov al,3
ret
min3:
cmp al,1
jl is0
jg is2
mov al,5
ret
is0:
mov al,7
ret
is2:
mov al,9
ret
encrypt endp
```

```
exit:
mov ah,4ch
int 21h///连接到下一次 c:\
code ends
end start
```

运行:

```
int 21h
cmp al,13
je exit
sub al,30h
call encrypt
add al,30h
lea si,buffer
mov [si],al
inc si
mov dl,al
mov ah,02h
int 21h
```

Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

Run File [LAB53.EXE]:

List File [NUL.MAP]:

Libraries [.LIB]:

LINK : warning L4021: no stack segment

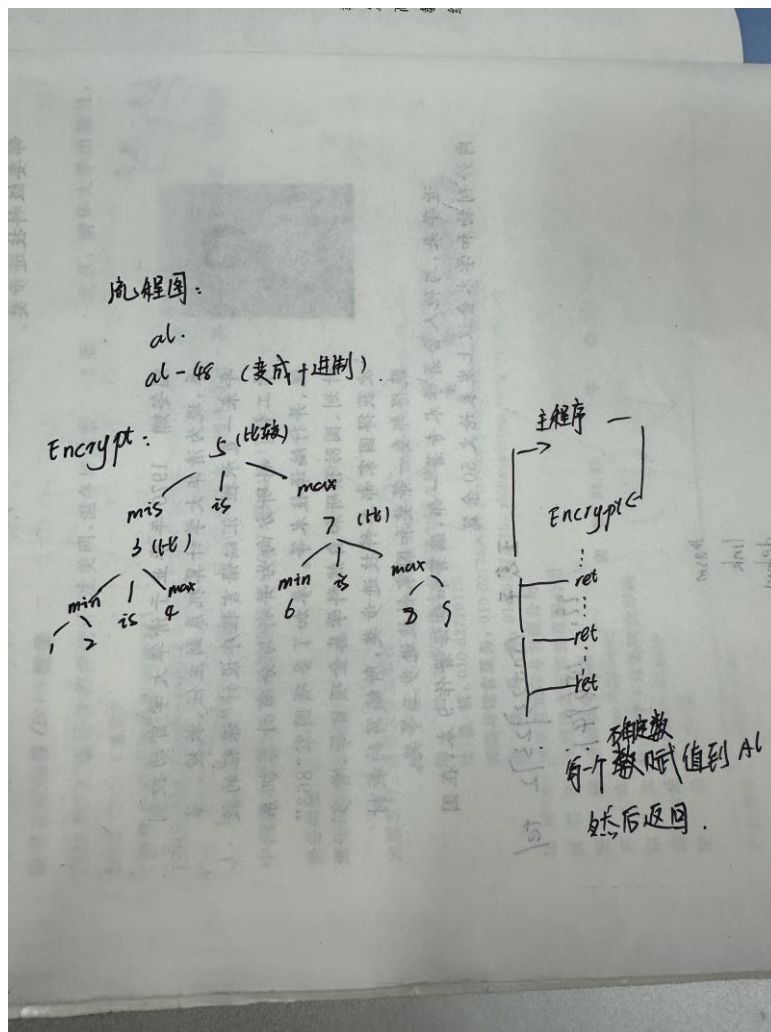
C:\>lab53

15 29 31 43 56 68 70 82 94 07

C:\>

为了方便查看加密结果，这里将原数字和加密数字作为一组输出到屏幕上

简易流程图:



五【实验过程原始记录(数据、图表、计算等)】

如果有则写。

六【实验结果及分析】

实验三中关于循环，在输入回车之前会一直循环，可能是由于后续的子程序会不断跳到 input 中，导致 cx 不断被初始化

七【思考题】

如果有则做。

实验报告打印格式说明

1. 标题：三号加粗黑体
2. 开课实验室：5号加粗宋体
3. 表中内容：
 - (1) 标题：5号黑体
 - (2) 正文：5号宋体
4. 纸张：16开(20cm×26.5cm)
5. 版芯
 - 上距：2cm
 - 下距：2cm
 - 左距：2.8cm
 - 右距：2.8cm

说明：1、实验成绩可按五级记分制（即优、良、中、及格、不及格），或者百分制记载，若需要将实验成绩加入对应课程总成绩的，则五级记分应转换为百分制。