西南大学

**实 验 报 告**

《计算机组成与汇编语言程序设计》课程

2020-2021学年度第2学期

专业年级 19 SE4

姓 名 吴嘉诚 \_

学 号 222019321062111

任课教师 李玉州

实验教师 李玉州

上机地点 25教 802

**西南大学计算机与信息科学学院**

2021年4月

**课程实验报告**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验题目 | | 汇编循环程序设计 | | |
| **实验时间** | | 2021年 4月22日 | | |
| 1. **实验目的及要求**   1、掌握循环控制指令  2、掌握计数控制循环方法  3、掌握条件控制循环方法  题目 :要求使用键盘输入特定字符串,存储在内存中,并对特定字符进行统计，并将统计结果分别存储在自定义的内存单元中，并在屏幕上显示出来相应的次数，该字符串以$符号结束。  （1）特定的字符有:大写字母、小写字母、数字及其他字符四类。  （2）特定的字符串为AAAaaa123456???$共15个字符。  （3）利用masm对源程序进行汇编，运行。  （4）注意，对关键行一定要写注释 | | | | |
| 1. **实验过程** 2. **代码及注释**   assume cs:code,ds:data  data segment  db 30 ;最大字符存放数  db ? ;实际输入数  db 30 dup(0) ;存放  dw 4 dup(0) ;大写，小写，数字，其他(word) 计数区域  upperCase db " Upper: $" ;大写  lowerCase db " Lower: $" ;小写  numberCase db " Numbers: $" ;数字  otherCase db " Others: $" ;其他  crlf db 0Ah,0Dh,'$' ;回车换行  str db "Please enter a string to be counted: ","$"  data ends  stacksg segment stack  db 64 dup(0)  stacksg ends  code segment  show\_num\_ax: ;显示数字通用函数  jmp short s  db 6 dup('$') ;预留给输出的内存，16位二进制数最多为5位十进制  ;因pop对字操作，因此再预留一位,并最后作为$结束符  s:  push ax ;保护寄存器  push bx  push cx  push dx  push si  push ds  push di  mov si,0  mov bx,10 ;设置输出进制  s0:  mov dx,0  div bx  add dx,30h  push dx  mov cx,ax  inc si  jcxz s1 ;判断是否商为0，即进制转换完成  jmp short s0 ;转换未完成则跳转到s0继续  s1:  mov ax,code  mov ds,ax  mov cx,si  mov di,2  lop:  pop [di] ;pop到预留内存中  inc di ;正好低位是有效ASCII码，高位无效，所以每次di只加1  loop lop  mov byte ptr [di],'$' ;结尾$符  mov dx,2 ;jmp指令占2字节  mov ah,9  int 21h ;显示数字  pop di  pop ds  pop si  pop dx  pop cx  pop bx  pop ax  ret ;函数返回  start:  mov ax,data  mov ds,ax  mov ax,stacksg  mov ss,ax  mov sp,64  lea dx,str ;显示提示字符串  mov ah,9  int 21h  mov dx,0  mov ah,0Ah  int 21h ;调用21h中断的10号功能等待输入  ;mov cl,ds:[1] ;可用此字节获取输入字符数，从而统计无$符结尾的字符串，只需修改循环条件（方法二）  ;mov ch,0  mov si,2 ;从[2]处开始为读入的字符串  count:  mov al,[si]  cmp al,48  jb other ;比0小  cmp al,58  jb num ;比:小  cmp al,65  jb other ;比A小  cmp al,91  jb upper ;比[小  cmp al,97  jb other ;比a小  cmp al,123  jb lower ;比{小  jmp other ;剩余其他  upper:  inc byte ptr ds:[32] ;计数  jmp short count\_end  lower:  inc byte ptr ds:[34]  jmp short count\_end  num:  inc byte ptr ds:[36]  jmp short count\_end  other:  inc byte ptr ds:[38]  jmp short count\_end  count\_end:  inc si  cmp al,'$' ;（方法一）  jne count ;用$作为指示符结束循环（方法一）  dec byte ptr ds:[38] ;对$多计了一次  ;loop count ;用cx循环计数（方法二）  lea dx,crlf ;回车换行  mov ah,9  int 21h  lea dx,upperCase ;显示各类字符计数结果  mov ah,9  int 21h  mov ax,ds:[32]  call show\_num\_ax  lea dx,lowerCase  mov ah,9  int 21h  mov ax,ds:[34]  call show\_num\_ax  lea dx,numberCase  mov ah,9  int 21h  mov ax,ds:[36]  call show\_num\_ax  lea dx,otherCase  mov ah,9  int 21h  mov ax,ds:[38]  call show\_num\_ax  mov ah,4ch  int 21h  code ends  end start   1. **实验思路**  * 调用21h中断的10号功能接收字符串 * 循环计数每个字符，计数储存到指定内存中 * 为了方便输出结果和对应的提示字符，我选择编写一个10进制数字输出函数，需输出数存储在ax中 * 调用21h的9号功能输出结果  1. **测试截图** | | | | |
| **三、实验总结与收获**   * **总结** * 成功完成实验并验证结果 * 开始时我使用10号功能的计数字符的那个字节来决定循环次数，后来发现要求用$符号结尾，并且$不计入’other’类中，所以再写了一种方法。 * 具有较高可拓展性：程序可根据需求灵活改写 * **收获** * 进一步提高了对内存的理解，提高了汇编语言编程技巧 * 学习巩固了21h号中断的知识，尤其加深了对9，10号功能的理解 * 学习到了子函数相关知识，巩固了对call，ret指令的理解 * 成功编写了一个10进制数字显示函数，可通用于以后的程序中 | | | | |
| **成绩** |  | | **评阅老师** |  |