

山东大学 计算机科学与技术 学院

汇编语言 课程实验报告

学号：202200130053	姓名：陈红瑞	班级：3 班
实验题目：实验 9：实验 2.3		
实验学时：2	实验日期：20241209	
实验目的： 巩固分支程序设计中所涉及的知识点。学会在自编程序中利用分支程序设计的理论与技巧。学会在自编程序中使用系统调用。		
实验环境：Windows11、DOSBox-0.74、Masm64		
源程序清单： Lab9.asm		
编译及运行结果： 这里首先定义数据段，如图，首先定义输入的缓冲区，由于这里是使用 0AH 进行输入，因此需要将缓冲区最大长度设置为 100，然后将第三个变量设置为容量为 100 的缓冲区。下面再定义一个 counter 数组，一共有 4 个数，分别表示小写字母，大写字母，数组和其他字符的个数。		
<pre>datasg segment para 'data' targetStr label byte max db 100 act db ? buf db 100 dup(?) ;标准输入对应的缓冲区 counter db 0,0,0,0 c1 db '0','0',13,10,'\$' c2 db '0','0',13,10,'\$' c3 db '0','0',13,10,'\$' mess1 db '[INFO] Please input the string.',13,10,'\$' mess2 db '[INFO] Successfully got the string.',13,10,'\$' datasg ends</pre>		

在代码段，这里首先调用 INT 21H 中的功能 0AH，实现将输入的内容存入缓冲区，在调试时可以看出这里已经将字符串存入到了内存中，这里输入了 8 个字符，第二个变量的值也变为了 8。

```
lea dx,targetStr
mov ah,0ah
int 21h
cmp act,0
```

```
-g13
Hello123
AX=0A6A BX=0000 CX=01AE DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=0777 IP=0013  NV UP EI PL ZR NA PE NC
0777:0013 803E010000    CMP     BYTE PTR [0001],00          DS:0001=0B
-d0
076A:0000  64 08 48 65 6C 6C 6F 31-32 33 0D 00 00 00 00 00    d.Hello123.....
076A:0010  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00    .....
076A:0020  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00    .....
076A:0030  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00    .....
076A:0040  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00    .....
076A:0050  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00    .....
076A:0060  00 00 00 00 00 00 00 00-00 30 30 0D 0A 24 30    .....00..$0
076A:0070  30 0D 0A 24 30 30 0D 0A-24 5B 49 4E 46 4F 5D 20    0..$00..$[INFO1
```

下面再实现读取每一个字符，这里需要先在 bx 存储字符串所在的地址，并在 CX 中保存字符串的长度，即循环的次数，这里缓冲区为第一个变量，因此字符串所在的首地址为 02H，CX 的值为 8。

```
work:
    lea bx,buf ;定位首字符
    xor cx,cx
    mov cl,act
```

```

AX=0A6A BX=0000 CX=01AE DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=0777 IP=001D  NU UP EI PL NZ NA PO NC
0777:001D 8D1E0200      LEA      BX,[0002]          DS:0002=6548
-t
AX=0A6A BX=0002 CX=01AE DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=0777 IP=0021  NU UP EI PL NZ NA PO NC
0777:0021 33C9          XOR      CX,CX
-t
AX=0A6A BX=0002 CX=0000 DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=0777 IP=0023  NU UP EI PL ZR NA PE NC
0777:0023 8A0E0100      MOV     CL,[0001]          DS:0001=08
-t
AX=0A6A BX=0002 CX=0008 DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=0777 IP=0027  NU UP EI PL ZR NA PE NC
0777:0027 8A07          MOV     AL,[BX]          DS:0002=48
-t

```

下面会依次判断是否为小写字母，大写字母以及数字。首先，将字符与 a 和 z 进行比较，如果小于 a 或者大于 z，就跳转到下面判断是否为大写，否则就将 counter 数组中的第一个数字加 1。

```

loop:
    mov al,[bx]
    cmp al,'a'
    jb next1
    cmp al,'z'
    ja next1
    inc counter[si]
    jmp exit

```

同理，这里会将字符与 A 和 Z 比较，如果小于 A 或者大于 Z 就跳转到下面判断是否为数字，否则将 counter 数组的第二个数加 1。最后会判断是否为数字，如果字符小于字符 0 或者大于字符 9 就跳转，否则将 counter 数组中的第三个数加 1。最后跳转到的部分表示将其他类型的字符的计数加 1。

```

next1:
    cmp al,'A'
    jb next2
    cmp al,'Z'
    ja next2
    inc counter[si+1]
    jmp exit

```

最后需要将数组中的内容进行输出,由于这里的每一个类型的字符的数量可能为一位数或两位数,因此需要先将数组中的每一个数字转换为两个 ASCII 字符,再进行输出。这里需要先在 CX 中将除数设置为 10,用于后面取出个位数和十位数。这里使用了 div 指令,在 8 位除法中,除数在 AL,余数在 AH,在这里分别为十位数和个位数。由于先在字符串中将存储数字的字符部分设置为' 0',因此这里只需要使用 xor 指令来转换为 ASCII 字符。

```
mov cx,10
mov bx,0
xor ax,ax

mov al,counter[si]

div cl

xor c1[bx],al
xor bx,1
xor c1[bx],ah
xor bx,1
xor ax,ax

mov al,counter[si+1]
div cl
xor c2[bx],al
xor bx,1
xor c2[bx],ah
xor bx,1
xor ax,ax
```

最后将这些数字进行输出,由于在数据段中已经定义好了每一个数字对应的字符串,包括换行符,这里还需要考虑第一个数是否为 0,由于计数的值可能有一位数,也可能有两位数。因此这里需要对字符串的首地址进行额外的判断,如果以 0 开头,说明这个值可能为 1 位数或 0,因此从第二个字符,即个位数开始输出即可,否则就从十位数开始进行输出。

```

mov ah,09h
mov bx,0
cmp c1[bx],'0'
je t1
lea dx,c1
jmp t2
t1: lea dx,c1+1
t2:
int 21h
cmp c2[bx],'0'
je t3
lea dx,c2
jmp t4
t3: lea dx,c2+1
t4: int 21h
cmp c3[bx],'0'
je t5
lea dx,c3
jmp t6
t5: lea dx,c3+1
t6: int 21h

```

最后，将程序提交到 OJ 的最终运行结果如图。

提交详情(汇编实验9)

×

✓ 代码

>

✓ 运行

>

3 总结

概览

评测信息

结论	测试通过
编号	917448963420525
提交时间	2024-12-12 15:24:53
评测时间	2024-12-12 15:24:56
评测模板	汇编实验9

关联信息

提交用户	202200130053 (ID: 10996)
题目信息	汇编实验9(1)

关闭

问题及收获：

1. 这个实验掌握了分支语句的实现方法。这里通过循环依次读取每一个字符，然后分别判断其 ASCII 值是否为某一个类型字符的范围内，如果超过范围，就进入到另一个类型的判断，否则将此类型的字符的计数的值加 1.