1）冒泡排序升序

```
1    ;冒泡排序升序
2    .386 ;允许汇编80386处理器的非特权指令，禁用其后处理器引入的汇编指令
3    .model flat, stdcall ;.model用于初始化程序的内存模式
4    option casemap :none ;大小写敏感
5    include D:\masm32\include\windows.inc
6    include D:\masm32\include\masm32.inc
7    include D:\masm32\include\kernel32.inc
8    includelib D:\masm32\lib\kernel32.lib
9    includelib D:\masm32\lib\masm32.lib ;函数的常量和链接库声明
10   .stack 4096 ;栈空间声明
11   .data
12       str1 byte "Please input unsigned int number: ",0 ;提示输入数组(以空格分割)
13       str2 byte "The result of bubble sort is: ",0 ;提示输出数组(以空格分割)
14       inputstr byte 80 dup(0) ;存储输入的字符串
15       pArray dword 15 dup(0) ;存储字符串转化成的各个数字
16       outputstr byte 80 dup(0) ;存储输出的字符串
17       const10 dword 10 ;常数10
18       count dword 1 ;数组长度
19       count2 dword 0 ;用于保存ecx
20   .code
21
22   main proc ;main过程
23       invoke StdOut,addr str1 ;输出提示
24       invoke StdIn,addr inputstr,60 ;输入数组
25       call strtonum ;输入字符串转化为数组
26       call BubbleSort ;数组冒泡排序
27       call numtostr ;排序后的数组重新转化为可输出的字符串
28       invoke StdOut,addr str2 ;输出提示
29       invoke StdOut,addr outputstr ;输出排序后数组
30       invoke ExitProcess,0
31   main endp
32
33   strtonum proc uses esi eax ebx ecx ;输入字符串转化为数组
34       mov esi,0
35       mov ebx,0
36       mov ecx,0
37   L1:
38       mov bl,[inputstr+esi] ;以esi为变址寄存器读取输入字符串
39       cmp bl,32 ;判断是否读取到空格(ASCII码值为32)
40       jne L2
41       inc count
42       add ecx,4 ;是空格则count+1,同时ecx+4
43       inc esi
44       mov bl,[inputstr+esi] ;略过空格
45   L2:
46       sub bl,48 ;字符转化为数字
47       mov eax,[pArray+ecx] ;读取数字
48       mul const10 ;eax值乘10
49       add eax,ebx ;加上刚读取的数字
50       mov [pArray+ecx],eax ;存回
51       inc esi
52       cmp [inputstr+esi],0 ;读取到串尾0停止
53       jne L1
54       ret
55   strtonum endp
56
57   BubbleSort proc uses eax ecx esi ;数组冒泡排序
58       mov ecx,count
59       dec ecx ;令ecx等于数组长度-1
60   L3:
```

```asm
61        push ecx ;入栈,保存外层ecx
62        mov esi,0
63 ∨ L4:
64        mov eax,[pArray+esi]
65        cmp eax,[pArray+esi+4] ;比较数
66        jbe L5
67        xchg eax,[pArray+esi+4]
68        mov [pArray+esi],eax ;若前一个数比后一个大则交换
69 ∨ L5:
70        add esi,4
71        loop L4 ;内层循环
72        pop ecx ;出栈,恢复外层ecx
73        loop L3 ;外层循环
74        ret
75    BubbleSort endp
76
77 ∨ numtostr proc ;排序后的数重新转化为可输出的字符串
78        mov ecx,count
79        dec ecx
80        mov esi,0
```

```asm
81     L10:
82        add esi,4
83        loop L10 ;使esi=4*(count-1),即偏移为最后一个数字
84        mov edi,0
85        mov edx,0
86        mov ecx,count
87     L6:
88        mov count2,ecx ;保存ecx
89        mov eax,[pArray+esi] ;获取数字
90        sub esi,4 ;移动到前一个数字
91     L7:
92        div const10 ;除法指令,被除数为edx:eax,eax存商,edx存余数
93        mov ecx,edx
94        mov edx,0
95        add ecx,48 ;数字转化为字符
96        push ecx ;字符入栈
97        inc edi
98        cmp eax,0
99        jne L7
100       push 32 ;商为0则入栈一个空格
101       inc edi
102       mov ecx,count2 ;恢复ecx
103       loop L6
104
105       mov ecx,edi ;循环次数为输出字符长度
106       mov esi,0
107    L8:
108       pop eax ;循环出栈
109       mov [outputstr+esi],al ;出栈字符保存到输出字符串中
110       inc esi
111       loop L8
112       mov [outputstr+esi],0 ;补充串尾0
113       ret
114    numtostr endp
115
116    end main
117
118    ;  ml -c -coff lab6.asm
119    ;  link -subsystem:console lab6.obj
120    ;  lab6.exe
```

2）冒泡排序降序

（将冒泡排序升序中起比较作用的 jbe 改成 jae，即可得到冒泡排序降序）

```
;冒泡排序降序
.386 ;允许汇编80386处理器的非特权指令，禁用其后处理器引入的汇编指令
.model flat, stdcall ;.model用于初始化程序的内存模式
option casemap :none ;大小写敏感
include D:\masm32\include\windows.inc
include D:\masm32\include\masm32.inc
include D:\masm32\include\kernel32.inc
includelib D:\masm32\lib\kernel32.lib
includelib D:\masm32\lib\masm32.lib ;函数的常量和链接库声明
.stack 4096 ;栈空间声明
.data
    str1 byte "Please input unsigned int number: ",0 ;提示输入数组(以空格分割)
    str2 byte "The result of bubble sort is: ",0 ;提示输出数组(以空格分割)
    inputstr byte 80 dup(0) ;存储输入的字符串
    pArray dword 15 dup(0) ;存储字符串转化成的各个数字
    outputstr byte 80 dup(0) ;存储输出的字符串
    const10 dword 10 ;常数10
    count dword 1 ;数组长度
    count2 dword 0 ;用于保存ecx
.code

main proc ;main过程
    invoke StdOut,addr str1 ;输出提示
    invoke StdIn,addr inputstr,60 ;输入数组
    call strtonum ;输入字符串转化为数组
    call BubbleSort ;数组冒泡排序
    call numtostr ;排序后的数组重新转化为可输出的字符串
    invoke StdOut,addr str2 ;输出提示
    invoke StdOut,addr outputstr ;输出排序后数组
    invoke ExitProcess,0
main endp

strtonum proc uses esi eax ebx ecx ;输入字符串转化为数组
    mov esi,0
    mov ebx,0
    mov ecx,0
L1:
    mov bl,[inputstr+esi] ;以esi为变址寄存器读取输入字符串
    cmp bl,32 ;判断是否读取到空格(ASCII码值为32)
    jne L2
```

```
    inc count
    add ecx,4 ;是空格则count+1,同时ecx+4
    inc esi
    mov bl,[inputstr+esi] ;略过空格
L2:
    sub bl,48 ;字符转化为数字
    mov eax,[pArray+ecx] ;读取数字
    mul const10 ;eax值乘10
    add eax,ebx ;加上刚读取的数字
    mov [pArray+ecx],eax ;存回
    inc esi
    cmp [inputstr+esi],0 ;读取到串尾0停止
    jne L1
    ret
strtonum endp

BubbleSort proc uses eax ecx esi ;数组冒泡排序
    mov ecx,count
    dec ecx ;令ecx等于数组长度-1
L3:
```

```asm
61          push ecx ;入栈,保存外层ecx
62          mov esi,0
63  L4:
64          mov eax,[pArray+esi]
65          cmp eax,[pArray+esi+4] ;比较数
66          jae L5
67          xchg eax,[pArray+esi+4]
68          mov [pArray+esi],eax ;若前一个数比后一个大则交换
69  L5:
70          add esi,4
71          loop L4 ;内层循环
72          pop ecx ;出栈,恢复外层ecx
73          loop L3 ;外层循环
74          ret
75  BubbleSort endp
76
77  numtostr proc ;排序后的数组重新转化为可输出的字符串
78          mov ecx,count
79          dec ecx
80          mov esi,0
81  L10:
82          add esi,4
83          loop L10 ;使esi=4*(count-1),即偏移为最后一个数字
84          mov edi,0
85          mov edx,0
86          mov ecx,count
87  L6:
88          mov count2,ecx ;保存ecx
89          mov eax,[pArray+esi] ;获取数字
90          sub esi,4 ;移动到前一个数字
91  L7:
92          div const10 ;除法指令,被除数为edx:eax,eax存商,edx存余数
93          mov ecx,edx
94          mov edx,0
95          add ecx,48 ;数字转化为字符
96          push ecx ;字符入栈
97          inc edi
98          cmp eax,0
99          jne L7
100         push 32 ;商为0则入栈一个空格
101         inc edi
102         mov ecx,count2 ;恢复ecx
103         loop L6
104
105         mov ecx,edi ;循环次数为输出字符长度
106         mov esi,0
107 L8:
108         pop eax ;循环出栈
109         mov [outputstr+esi],al ;出栈字符保存到输出字符串中
110         inc esi
111         loop L8
112         mov [outputstr+esi],0 ;补充串尾0
113         ret
114 numtostr endp
115
116 end main
117
118 ; ml -c -coff lab6.1.asm
119 ; link -subsystem:console lab6.1.obj
120 ; lab6.1.exe
```

3) 使用 cmd 控制台，cd 到 D:\masm32\bin；

使用 masm32 运行命令：

ml.exe　-c -coff Test.asm

link.exe -subsystem:console Test.obj

Test.exe

如图，程序运行，结果正确

```
C:\Users\86187>D:

D:\>cd D:\masm32\bin

D:\masm32\bin>ml.exe  -c -coff lab6.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997.  All rights reserved.

 Assembling: lab6.asm

***********
ASCII build
***********

D:\masm32\bin>link.exe -subsystem:console lab6.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.


D:\masm32\bin>lab6.exe
Please input unsigned int number: 7 5 3 2 6 9 1 8 4 0
The result of bubble sort is:  0 1 2 3 4 5 6 7 8 9
```

```
D:\masm32\bin>ml.exe  -c -coff lab6.1.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997.  All rights reserved.

 Assembling: lab6.1.asm

***********
ASCII build
***********

D:\masm32\bin>link.exe -subsystem:console lab6.1.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.


D:\masm32\bin>lab6.1.exe
Please input unsigned int number: 7 5 3 2 6 9 1 8 4 0
The result of bubble sort is:  9 8 7 6 5 4 3 2 1 0
```