

《汇编语言》实验报告

班级	信息十班	实验日期	2022.11.25	实验成绩	
姓名	陈澄	学号	32420212202930		
实验名称	汇编语言第六次实验				
实验目的、要求	1) 汇编指令综合应用 2) 熟悉 32 位 Intel 汇编指令				
实验内容、步骤及结果	<p>实验内容：</p> <p>请使用 32 位的 Intel x86 的指令，编写计算冒泡排序算法的程序（从小到大排序、从大到小 排序）；并在 32 位的 Intel x86 汇编语言环境下运行通过。需要注意的点是，在 32 位系统下，仍可以使用 16 位寄存器，但是地址变成 32 位了。存地址一般就需要用 32 位寄存器。</p> <p>数据原顺序：7,5,3,2,6,9,1,8,4,0</p> <p>从小到大和从大到小都需要输出，可以放在两个程序里分别运行。</p> <p>实验步骤以及结果：</p> <p>步骤：</p> <ol style="list-style-type: none">1. 初始化 EBX 寄存器为 0 用于后续数据寻址2. 依次对比相邻数据若前>(<)后则交换3. 反复 n=8 次使最大（小）数达到末尾4. 减少对比次数并从第一组数据开始对比5. 对比次数为 0 时转到输出程序6. 输出 <p>结果：</p>				

```
命令提示符
ASCII build
*****
D:\masm32\bin\link.exe -subsystem:console Test.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.

D:\masm32\bin\Test.exe
1 2 3 4 5 6 7 8 9
D:\masm32\bin\ml.exe -c -coff Test1.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997. All rights reserved.

Assembling: Test1.asm
*****
ASCII build
*****
D:\masm32\bin\link.exe -subsystem:console Test1.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.

D:\masm32\bin\Test1.exe
1 2 3 4 5 6 7 8 9
从小到大排序
```

```
命令提示符
ASCII build
*****
D:\masm32\bin\link.exe -subsystem:console Test1.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.

D:\masm32\bin\Test1.exe
1 2 3 4 5 6 7 8 9
D:\masm32\bin\ml.exe -c -coff Test2.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997. All rights reserved.

Assembling: Test2.asm
*****
ASCII build
*****
D:\masm32\bin\link.exe -subsystem:console Test2.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.

D:\masm32\bin\Test2.exe
9 8 7 6 5 4 3 2 1
从大到小排序
```

相关代码：

```
21 include \masm32\macros\macros.asm
22 .data
23 sz db '1 ',0
24 DATA1 db 7,5,3,2,6,9,1,8,4,0dh,0ah,'$'
25 szLINE db 0ah,0dh,0
26 .code
27 START: MOV SI,8
28 | MOV EBX,0; 初始化EBX为0, 后用于数据寻址
29 LOOP1: MOV AL,DATA1[EBX+1]
30 | CMP DATA1[EBX],AL; 依次对比相邻数据
31 | JB LOOP2
32 | MOV AH,DATA1[EBX]; 前面数据大于后面时进行交换
33 | MOV DATA1[EBX],AL
34 | MOV DATA1[EBX+1],AH
35 LOOP2: INC BX
36 | CMP BX,SI; 进行n次比较使最大的数到达末尾
37 | JB LOOP1
38 | MOV BX,0
39 | DEC SI; 减少比较次数并返回
40 | CMP SI,0
41 | JA LOOP1; 需要比较次数为0时退出
42 | MOV SI,8
43 | MOV BP,0
44 | LEA EBX,DATA1
45 PRINT1: LEA ecx,sz ; 输出程序
46 | MOV AL,[EBX]
47 | add al,'0'
48 | mov [ecx],al
49 | push offset sz
50 | call Stdout
51 | ADD BP,1
52 | ADD EBX,1
53 | CMP BP,SI
54 | JLE PRINT1
55 END START
```

	<pre>21 include \masm32\macros\macros.asm 22 .data 23 sz db '1 ',0 24 DATA1 db 7,5,3,2,6,9,1,8,4,0dh,0ah,'\$' 25 szLINE db 0ah,0dh,0 26 .code 27 START: MOV SI,8 28 MOV EBX,0;初始化EBX为0,后用于数据寻址 29 LOOP1: MOV AL,DATA1[EBX+1] 30 CMP DATA1[EBX],AL;依次对比相邻数据 31 JA LOOP2 32 MOV AH,DATA1[EBX];前面数据小于后面时进行交换 33 MOV DATA1[EBX],AL 34 MOV DATA1[EBX+1],AH 35 LOOP2: INC BX 36 CMP BX,SI;进行n次比较使最小的数到达末尾 37 JB LOOP1 38 MOV BX,0 39 DEC SI;减少比较次数并返回 40 CMP SI,0 41 JA LOOP1;需要比较次数为0时退出 42 MOV SI,8 43 MOV BP,0 44 LEA EBX,DATA1 45 PRINT1: LEA ecx,sz ;输出程序 46 MOV AL,[EBX] 47 add al,'0' 48 mov [ecx],al 49 push offset sz 50 call StdOut 51 ADD BP,1 52 ADD EBX,1 53 CMP BP,SI 54 JLE PRINT1 55 END START</pre>
总结	<ol style="list-style-type: none">1. 学会了 32 位 Intel 汇编指令的使用方法2. 学会了 masm32 的安装与使用