指令系统作业

1. 交换数组元素对

编写循环程序，用变址寻址交换数组中的数值对，每对中包含偶数个元素。即，元素 i 与元素 i+1 交换，元素 i+2 与元素 i+3 交换，以此类推。

```
.386
.model flat, stdcall

include windows.inc
include user32.inc
include kernel32.inc
includelib user32.lib
includelib kernel32.lib

.data
    array db 1, 2, 3, 4, 5, 6, 7, 8
    aSize db 8

.code
main PROC
    mov esi, offset array
    movzx ecx, aSize
    shr ecx, 1                ; 循环次数为数组大小的一半

swap_loop:
    mov al, [esi]
    mov bl, [esi+1]
    mov [esi], bl
    mov [esi+1], al

    add esi, 2
    loop swap_loop

    invoke ExitProcess, 0
main ENDP
END main
```

2. 数组元素间隔之和

编写循环程序，用变址寻址计算连续数组元素的间隔总和。数组元素为双字，按非递减次序排列。比如，数组为{0,2,5,9,10},则元素间隔为 2、3、4 和 1,那么间隔之和等于 10。

```
.386
.model flat, stdcall

include windows.inc
```

```
include user32.inc
include kernel32.inc
includelib user32.lib
includelib kernel32.lib

.data
    array dd 0, 2, 5, 9, 10
    aSize dd 5
    sum dd 0

.code
main PROC
    mov esi, offset array
    mov ecx, aSize
    dec ecx
    xor eax, eax

l:
    mov ebx, [esi+4]
    sub ebx, [esi]
    add eax, ebx
    add esi, 4
    loop l

    mov [sum], eax

    invoke ExitProcess, 0
main ENDP
END main
```

3. 斐波那契数列

编写循环程序，计算斐波那契(Fibonacci)数列前七个数值之和，算式如下：

Fib(1)=1,Fib(2)=1, Fib(*n*)=Fib(*n*-1)+Fib(*n*-2)

```
.386
.model flat, stdcall

include windows.inc
include user32.inc
include kernel32.inc
includelib user32.lib
includelib kernel32.lib

.data
    sum dd 0
```

```
.code
main PROC
	mov eax, 1
	mov ebx, 1
	mov ecx, 5

	add eax, ebx
	add [sum], eax

l:
	mov edx, eax
	add eax, ebx
	mov ebx, edx
	add [sum], ebx
	loop l

	invoke ExitProcess, 0
main ENDP
END main
```

4. 数组元素移位

编写循环程序，用变址寻址把一个 32 位整数数组中的元素向前(向右)循环移动一个位置，数组最后一个元素的值移动到第一个位置上。比如，数组[10,20,30,40]移位后转换为[40,10,20,30]。

```
.386
.model flat, stdcall

include windows.inc
include user32.inc
include kernel32.inc
includelib user32.lib
includelib kernel32.lib

.data
	array dd 10, 20, 30, 40
	aSize dd 4
	temp dd 0

.code
main PROC
	mov esi, offset array
	mov ecx, aSize
	dec ecx
```

```
        mov eax, [esi+ecx*4]
        mov [temp], eax

l:
        mov eax, [esi+ecx*4-4]
        mov [esi+ecx*4], eax
        loop l
        mov eax, [temp]
        mov [esi], eax

        invoke ExitProcess, 0
main ENDP
END main
```

5. 编写指令序列，把三个内存字节左移一位，使用数据如下：
wordArray WORD 810Dh, 0C064h, 93ABh

```
.386
.model flat, stdcall

include windows.inc
include user32.inc
include kernel32.inc
includelib user32.lib
includelib kernel32.lib

.data
        array WORD 810Dh, 0C064h, 93ABh

.code
main PROC
        mov ax, array
        shl ax, 1
        mov array, ax

        mov ax, array[2]
        shl ax, 1
        mov array[2], ax

        mov ax, array[4]
        shl ax, 1
        mov array[4], ax
```

```
    invoke ExitProcess, 0
main ENDP
END main
```

6. 使用 32 位无符号操作数，用汇编语言实现下述 C++表达式：

```
 vall =(val2 *val3)/(val4 -3)
.386
.model flat, stdcall

include windows.inc
include user32.inc
include kernel32.inc
includelib user32.lib
includelib kernel32.lib

.data
    val2 dd 5
    val3 dd 10
    val4 dd 15
    vall dd 0

.code
main PROC
    mov eax, [val2]
    imul eax, [val3]

    mov ebx, [val4]
    sub ebx, 3

    cdq
    idiv ebx
    mov [vall], eax

    invoke ExitProcess, 0
main ENDP
END main
```