

南京林业大学



汇编语言上机实验

任务书

实验一 查看 CPU 和内存，用机器指令和汇编指令编程

一. 实验目的

加深对进程调度的理解，熟悉进程调度的不同算法，比较其优劣性。

二. 实验内容

什么是 debug

我们用到的 debug 功能

进入 debug



```
DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

C:\>dir
Directory of C:\.
.
<DIR>                29-10-2025 10:05
..
<DIR>                29-10-2025 10:05
    0 File(s)          0 Bytes.
    2 Dir(s)      262,111,744 Bytes free.

C:\>edbug
Illegal command: edbug.

C:\>debug
Illegal command: debug.

C:\>-r
Illegal command: -r.

C:\>debug
-rr
br Error
-r
AX=0000  BX=0000  CX=0000  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=073F  ES=073F  SS=073F  CS=073F  IP=0100  NV UP EI PL NZ NA PO NC
073F:0100 0000          ADD     [BX+SI],AL           DS:0000=CD
_-
```

用 R 命令查看、改变寄存器的内容

```
● ○ ● DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

-r
AX=0100 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=0B39 IP=0100 NV UP EI PL NZ NA PO NC
0B39:0100 40           INC     AX
-e 0b39:0200
0B39:0200 00.5b

-e ff00:0200
FF00:0200 00.51

-r ip
IP 0100
:200
]r
AX=0100 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=0B39 IP=0200 NV UP EI PL NZ NA PO NC
0B39:0200 5B           POP     BX
-r cs
CS 0B39
:ff00
-r
AX=0100 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=FF00 IP=0200 NV UP EI PL NZ NA PO NC
FF00:0200 0000         ADD     [BX+SI],AL             DS:0000=CD
[
```

用 debug 的 D 命令查看内存中的内容

```
● ○ ● DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

:ff00
-r
AX=0100 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=FF00 IP=0200 NV UP EI PL NZ NA PO NC
FF00:0200 0000         ADD     [BX+SI],AL             DS:0000=CD
-d 1000:0
1000:0000 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0010 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0020 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0030 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0040 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0050 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0060 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0070 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
-d 1000:9
1000:0000          00 00 00 00 00 00 00 00 ...
1000:0010 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0020 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0030 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0040 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0050 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0060 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0070 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
1000:0080 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ...
[
```

```
 DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-d 1000:9
1000:0000          00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1000:0010  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0020  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0030  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0040  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0050  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0060  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0070  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0080  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
-d
1000:0080          00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1000:0090  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:00A0  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:00B0  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:00C0  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:00D0  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:00E0  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:00F0  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
1000:0100  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
-d 1000:0 9
1000:0000  00 00 00 00 00 00 00 00 00-00 00
-d 1000:0 0
1000:0000  00
-
```

用 debug 的 E 命令来改写内存中的内容

```
 
d 1000:0 f
1000:0000  00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00
-e 1000:0 0 1 2 3 4 5 6 7 8 9
-d 1000:0 f
1000:0000  00 01 02 03 04 05 06 07-08 09 00 00 00 00 00 00 00 00
-d 1000:10 19
1000:0010  00 00 00 00 00 00 00 00 00-00 00
-e 1000:10
1000:0010  00.0    00.1    00.2    00.1c

-e 1000:0 1 'a' 2 'b' 3 'c'
-d 1000:0 f
1000:0000  01 61 02 62 03 63 06 07-08 09 00 00 00 00 00 00 .a.b.c.....
```

用 E 命令向内存中写入机器码

```

● ○ ● DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-e 1000:0 1 "a+b" 2 "c++" 3 "IBM"
-d 1000:0 f
1000:0000 01 61 2B 62 02 63 2B 2B-03 49 42 4D 00 00 00 00 .a+b.c++.IBM....
-e 1000:0 b8 01 00 b9 02 00 01 c8
-e 1000:0 8
-d 1000:0 1f
1000:0000 08 01 00 B9 02 00 01 C8-03 49 42 4D 00 00 00 00 .....IBM....
1000:0010 00 01 02 1C 00 00 00 00-00 00 00 00 00 00 00 00 .....
-u 1000:0
1000:0000 0801 OR [BX+DI],AL
1000:0002 00B90200 ADD [BX+DI+0002],BH
1000:0006 01C8 ADD AX,CX
1000:0008 034942 ADD CX,[BX+DI+42]
1000:000B 4D DEC BP
1000:000C 0000 ADD [BX+SI],AL
1000:000E 0000 ADD [BX+SI],AL
1000:0010 0001 ADD [BX+DI],AL
1000:0012 021C ADD BL,[SI]
1000:0014 0000 ADD [BX+SI],AL
1000:0016 0000 ADD [BX+SI],AL
1000:0018 0000 ADD [BX+SI],AL
1000:001A 0000 ADD [BX+SI],AL
1000:001C 0000 ADD [BX+SI],AL
1000:001E 0000 ADD [BX+SI],AL

```

```

● ○ ● DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
1000:0018 0000 ADD [BX+SI],AL
1000:001A 0000 ADD [BX+SI],AL
1000:001C 0000 ADD [BX+SI],AL
1000:001E 0000 ADD [BX+SI],AL
-e 1000:0 b8 01 00 b9 02 00 01 c8
-r
AX=0100 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=FF00 IP=0200 NV UP EI PL NZ NA PO NC
FF00:0200 0000 ADD [BX+SI],AL DS:0000=CD
-rccs
CS FF00
:1000
-rip
IP 0200
:0
-r
AX=0100 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0000 NV UP EI PL NZ NA PO NC
1000:0000 B80100 MOV AX,0001
-t

```

```

AX=0001 BX=0000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0006 NV UP EI PL NZ NA PO NC
1000:0006 01C8 ADD AX,CX
-t

```

```

AX=0003 BX=0000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0008 NV UP EI PL NZ NA PE NC
1000:0008 034942 ADD CX,[BX+DI+42] DS:0042=0000

```

用 debug 的 A 命令以汇编形式在内存中写入机器指令

DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

```
AX=0003 BX=0000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0008 NV UP EI PL NZ NA PE NC
1000:0008 034942 ADD CX,[BX+DI+42] DS:0042=0000
-a 1000:0
1000:0000 mov ax,1
1000:0003 mov bx,2
1000:0006 mov cx,3
          ^ Error
1000:0006 mov cx,3
1000:0009 add ax,bx
1000:000B add ax,cx
1000:000D add ax,ax
1000:000F
-d 1000:0 f
1000:0000 B8 01 00 BB 02 00 B9 03-00 01 D8 01 C8 01 C0 00 ..... .
]--a
          ^ Error
-a
1000:000F mov ax,1
1000:0012 mov bx,2
1000:0015 mov cx,3
1000:0018 add ax,bx
1000:001A add ax,cx
1000:001C add ax,ax
1000:001E _
```

三. 实验任务

1. 使用 Debug 将下面程序写入内存，逐条执行，观察每条指令执行后 cpu 相关寄存器中内容的变化。

代码的输入过程：

```
-a 1000:0
1000:0000 mov ax,4e20
1000:0003 add ax,1416
1000:0006 mov bx,2000
1000:0009 add ax,bx
1000:000B mov bx,ax
1000:000D add ax,bx
1000:000F mov ax,001a
1000:0012 mov bx,0026
1000:0015 add al,bl
1000:0017 add ah,bl
1000:0019 add bh,al
1000:001B mov ah,0
1000:001D add al,bl
1000:001F add al,9c
1000:0021
```

执行过程：

```
● ○ ● DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

AX=4E20 BX=0000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0003 NV UP EI PL NZ NA PE NC
1000:0003 051614 ADD AX,1416
-t

AX=6236 BX=0000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0006 NV UP EI PL NZ NA PE NC
1000:0006 BB0020 MOV BX,2000
-t

AX=6236 BX=2000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0009 NV UP EI PL NZ NA PE NC
1000:0009 01D8 ADD AX,BX
-t

AX=8236 BX=2000 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=000B OV UP EI NG NZ NA PE NC
1000:000B 89C3 MOV BX,AX
-t

AX=8236 BX=8236 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=000D OV UP EI NG NZ NA PE NC
1000:000D 01D8 ADD AX,BX
-t
```

```
● ○ ● DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

AX=046C BX=8236 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=000F OV UP EI PL NZ NA PE CY
1000:000F B81A00 MOV AX,001A
-t

AX=001A BX=8236 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0012 OV UP EI PL NZ NA PE CY
1000:0012 BB2600 MOV BX,0026
-t

AX=001A BX=0026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0015 OV UP EI PL NZ NA PE CY
1000:0015 00D8 ADD AL,BL
-t

AX=0040 BX=0026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0017 NV UP EI PL NZ AC PO NC
1000:0017 00DC ADD AH,BL
-t

AX=2640 BX=0026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=1000 IP=0019 NV UP EI PL NZ NA PO NC
1000:0019 00C7 ADD BH,AL
-t
```

```

AX=2640  BX=4026  CX=0002  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=073F  ES=073F  SS=073F  CS=1000  IP=001B    NV UP EI PL NZ NA PO NC
1000:001B B400          MOV      AH,00
-t

AX=0040  BX=4026  CX=0002  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=073F  ES=073F  SS=073F  CS=1000  IP=001D    NV UP EI PL NZ NA PO NC
1000:001D 00D8        ADD      AL,BL
-t

AX=0066  BX=4026  CX=0002  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=073F  ES=073F  SS=073F  CS=1000  IP=001F    NV UP EI PL NZ NA PE NC
1000:001F 049C        ADD      AL,9C
-t

AX=0002  BX=4026  CX=0002  DX=0000  SP=00FD  BP=0000  SI=0000  DI=0000
DS=073F  ES=073F  SS=073F  CS=1000  IP=0021    NV UP EI PL NZ AC PO CY
1000:0021 0000        ADD      [BX+SI],AL
                                         DS:4026=00
-
```

2.将下面 3 条指令写入从 2000:0 开始的内存单元中，利用这三条指令计算 2 的 8 次方
指令输入过程：

```

t-r cs
CS 1000
:t:2000
t-r ip
IP 0021
:0
t-a
1000:0021 mov ax,1
1000:0024
-r ip
IP 0000
:0
-a 2000:0
2000:0000 mov ax,1
2000:0003 add ax,ax
2000:0005 jmp 2000:0003
2000:0007

```

运行过程：

```
AX=0001 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NU UP EI PL NZ AC PO CY
2000:0003 01C0 ADD AX,AX
-t

AX=0002 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NU UP EI PL NZ MA PO NC
2000:0005 EBFC JMP 0003
-t

AX=0002 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NU UP EI PL NZ MA PO NC
2000:0003 01C0 ADD AX,AX
-t

AX=0004 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NU UP EI PL NZ MA PO NC
2000:0005 EBFC JMP 0003
-t

AX=0004 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NU UP EI PL NZ MA PO NC
2000:0003 01C0 ADD AX,AX
```

```
AX=0008 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NU UP EI PL NZ MA PO NC
2000:0005 EBFC JMP 0003
-t

AX=0008 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NU UP EI PL NZ MA PO NC
2000:0003 01C0 ADD AX,AX
-t

AX=0010 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NU UP EI PL NZ AC PO NC
2000:0005 EBFC JMP 0003
-t

AX=0010 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NU UP EI PL NZ AC PO NC
2000:0003 01C0 ADD AX,AX
-t

AX=0020 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NU UP EI PL NZ MA PO NC
2000:0005 EBFC JMP 0003
```

```

DOSBox 0.74-3-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

AX=0040 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NV UP EI PL NZ NA PO NC
2000:0005 EBFC JMP 0003
-t

AX=0040 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NV UP EI PL NZ NA PO NC
2000:0003 01C0 ADD AX,AX
-t

AX=0080 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NV UP EI PL NZ NA PO NC
2000:0005 EBFC JMP 0003
-t

AX=0100 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0003 NV UP EI PL NZ NA PE NC
2000:0003 01C0 ADD AX,AX
-t

AX=0100 BX=4026 CX=0002 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=2000 IP=0005 NV UP EI PL NZ NA PE NC
2000:0005 EBFC JMP 0003
-t

```

3. 查看内存中的内容

PC 机主板的 ROM 中写有一个生产日期，在内存 FFF00H~FFFFFH 的某几个单元中，请找到这个生成日期并试图改变它。

```

-d ffff:0
FFF8:0000 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0010 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0020 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0030 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0040 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0050 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0060 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 ..... .
FFF8:0070 EA C0 12 00 F0 30 31 2F-30 31 2F 39 32 00 FC 55 ..... 01/01/92..U

```

4. 向内存从 B8100H 开始的单元中填写数据，如：

```

-e b810:0000 01 01 02 02 03 03 04 04
-d b810:0000
B810:0000 30 07 20 07 30 07 30 07-20 07 30 07 30 07 20 07 0. .0.0. .0.0. .
B810:0010 20 07 20 07 20 07 20 07-20 07 20 07 20 07 20 07 . . . . .
B810:0020 20 07 20 07 20 07 20 07-20 07 20 07 20 07 20 07 . . . . .
B810:0030 20 07 20 07 20 07 20 07-20 07 20 07 20 07 20 07 . . . . .
B810:0040 46 07 46 07 46 07 37 07-3A 07 30 07 30 07 31 07 F.F.F.7.:0.0.1.
B810:0050 30 07 20 07 20 07 30 07-30 07 20 07 30 07 30 07 0. . .0.0. .0.0.
B810:0060 20 07 30 07 30 07 20 07-30 07 30 07 20 07 30 07 .0.0. .0.0. .0.
B810:0070 30 07 20 07 30 07 30 07-20 07 30 07 30 07 20 07 0. .0.0. .0.0. .

```

B810:0 附近的内存区域是不可写入的，因为这是一个显存内存区域