

PC-t módos;  $L_1$  utas;  $L_2$  sok  
feltételos ugró

$\mathcal{F}_{IF} C$

NC

$Z, E$

NZ NE

O

No

M és P

NM és NP

Flötenköl

A

--

B

AE

BE

NBE

NAE

NB

NA

EL<sup>1</sup>ides

G

NLE

L

NGE

GE

NL

LE

NG

PE

NPO

PO

NPE

---

LOOP

LOOP E

LOOP Z

LOOP NE

LOOP NZ

72 LACI

⋮

LACI:

Mov cx, 8

CIKLUS:

⋮

Loop

CIKLUS

CIKIMAJ.

MOV CX, 5

Loop CIKIMAJ

OSLOR: MOV BX, OSLOR

MOV CX, SOR

SORCIK:

⋮

LOOP SORCIK

DEC

INC

BX  
OSLOR



OSZCIK: MOV CX, OSZLOP  
PUSH CX

SORCIK: MOV CX, SORSZA

⋮

LOOP SORCIK

POP CX

LOOP OSZCIK

JC XZ

JMP

SHORT LACI

72

MESSZ E

72 KÖZEL

JMP MESSZ

KÖZEL:

MESSZ:

# Aritmetikai utasítások

ADD

AX, BX

ADC

[BX], CX

H

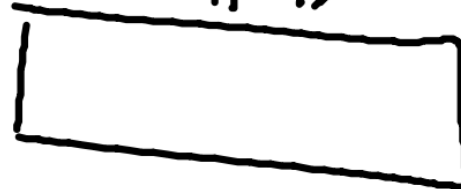
L



+

ADC

ADD



SUB SI, 8

SBB

INC DH

DEC

word ptr [BX]

NEG

AX

CMP AL, 10'

SUB AX, AX

-----

MUL

<sup>8</sup>  
16 bites érték

$AX \leftarrow AL * 8\text{bit}$

$DX:AX \leftarrow AX * 16\text{bit}$

IMUL

DIV

<sup>8</sup>  
16 bites érték

DX:AX

/ 32 bit

H AX  
L DX

AX

/ 16 bit

H AL  
L AH

DUP LABEL WORD  
DB ?

P0 DB 38 DUP (?)

20 DW 5, -1, 65535, 'KZ'

MOV AL, [P0]



DD

3.2, 2.4E8

DQ

7.17, -8, 5

DT

LOGIKAI

AND

AX, 00FFH

OR

CX, BP

XOR

AL, 11110000B

TEST

AL, 0FH

NOT

AH

AH	AX	AL
$  \begin{array}{r}  01000010 \quad 11110101 \\  00000000 \quad 11111111 \\  \hline  00000000 \quad 11110101  \end{array}  $		

XOR      AX, AX

$$\begin{array}{r}
 11110101 \\
 11110101 \\
 \hline
 00000000
 \end{array}$$

304 → '0'

314 → '1'

⋮

384 → 'z'

AND AX, 0FH

MOV SI, AX

MOV AX, 10

MUL CX

<sup>7C</sup>  
ADD AX, SI

MOV CX, AX

HIBA

RET

OR AX, AX

AND AX, AX

①

ADD AL, 30H

②

OR AL, 30H

③

OR AL, 48

OR AL, 10

VARAKO:

NOP

NOP

LOOP

VARAKO

CLC

STC

CMC

CLZ

STZ

CLI

STI

CLI

IDE.

JMP

IDE

CLD

STD

# HLT

---

String kernel

CHK:      MOV CX, 312  
            MOV BX, OFFSET PO  
            MOV BP, OFFSET CH  
            MOV AL, [BX]  
            MOV [BP], AL  
            INC BX  
            INC BP



LOOP CLK

DS: [SI] forwards

ES: [DI] cell

MOVSB

MOVSW

MOV CX, 312

MOV SI, OFFSET FO

MOV DI, OFFSET CE

CIKLUS:

MOVSB

✓ LOOP CIKLUS

REP

MOVSB

LODS<sup>B</sup><sub>W</sub>     $AL \leftarrow DS:[SI]$   
                   $AX_1 \leftarrow DS:[SI]$   
STOS<sup>B</sup><sub>W</sub>     $ES:[DI] \leftarrow AL$   
                   $ES:[DI] \leftarrow AX$   
MOV CX, 250  
MOV DI, OFFSET TER  
XOR AX, AX  
REP STOSW

SCAS<sup>B</sup><sub>W</sub>

AL OH ES:[DI]  
AX

REPZ

REP<sub>E</sub>

REPZ

REP<sub>E</sub>

CMPS<sup>B</sup><sub>W</sub>

LEA AX, [BX]

LEA AX, [BX+8]

LEA EAX, [EBX\*4]  
[EBX\*4+EBX]

LDS DX, [ADAT]

ADAT

DD

12345678H  
0 S

E BBBB BBBB  
1 00 00 00 00

1 000  
1 1 1 1 1 000

XOR AX, AX  
CWD

$XLAT \Rightarrow MOV AL, [BX+AL]$

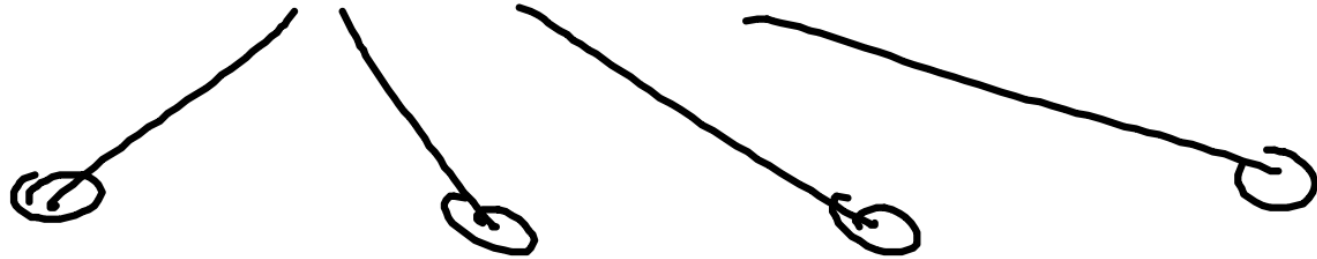
$INC \quad [LA-1]$

$LACI \quad DW \quad 3$



ADD AL, 99H

ADD AL, BL

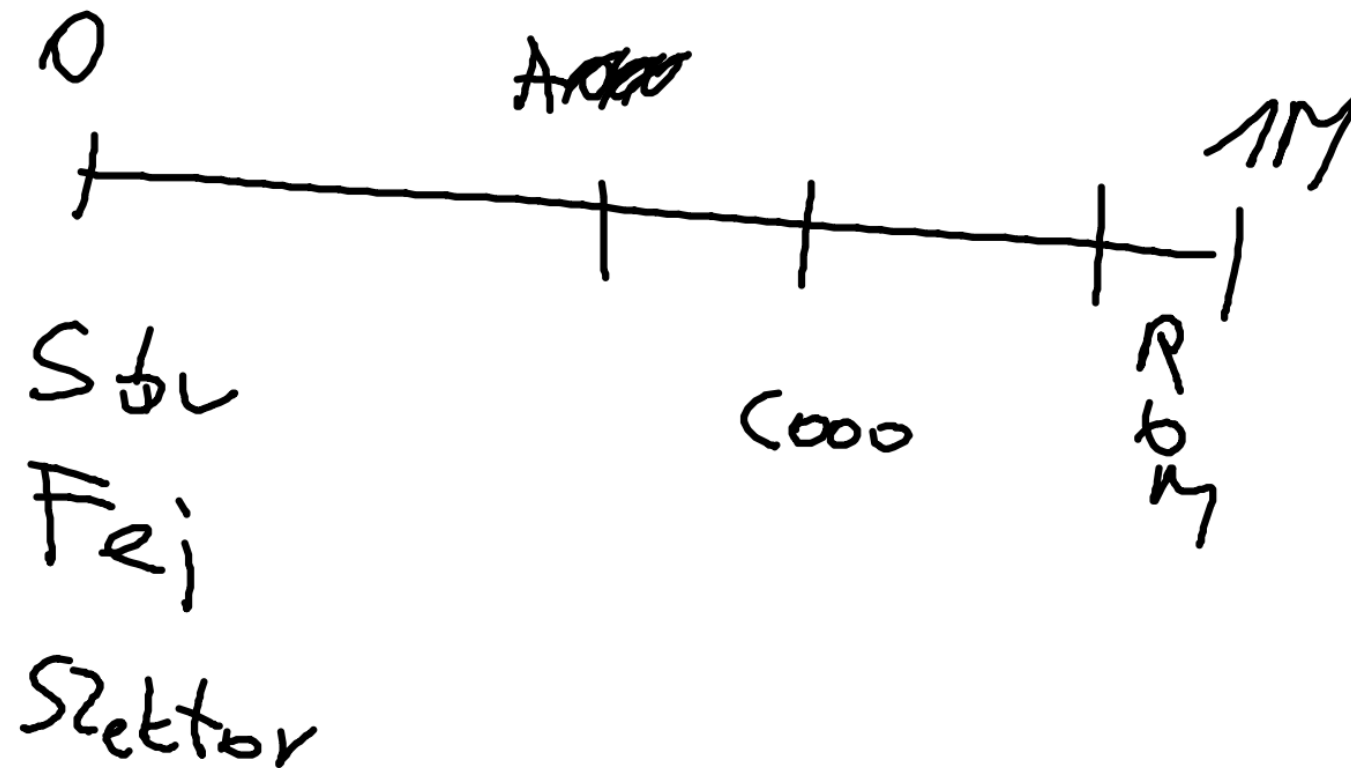


ADD AL, KOPR

0000:0000

XT FFFF:0000

F000:FFFF



XOR AX, AX

MOV ES, AX

CLI

MOV SI, ES: [21H \* 4]

MOV DI, ES: [21H \* 4 + 2]

MOV ES: [21H \* 4], CS

MOV ES: [21H \* 4 + 2], OFFSET  
ITRUT

MOV WORD PTR [TCIM], SI  
MOV WORD PTR [TCIM+2], DI  
STI

?

TCIM DD ?

ITRUT: PUSH ..

POP...

JMP CS:[ITC1M]

ITRUT:

push r  
CALL CS:[ITRUT]  
:  
IRET

push f  
push cs  
push ip

f — — cli

f — — tv

CS:IP ← új érték

XOR AX, AX

INT 33H

OR AX, AX

JZ WINCSF

MOV AX, 1

INT 33H



⋮

MOV AX, 2

INT 33H

MOV AH, 4CH

INT 21H

MOV AX, 3

INT 33h

; CX ← X posició

; DX ← Y posició

; BX ← ayomógom bok állapota

$AH \leftarrow \emptyset$  leütésvárás

$\hookrightarrow AL \leftarrow$  Leütés kódja

$AH \leftarrow 1$  Leütés vizsgálata

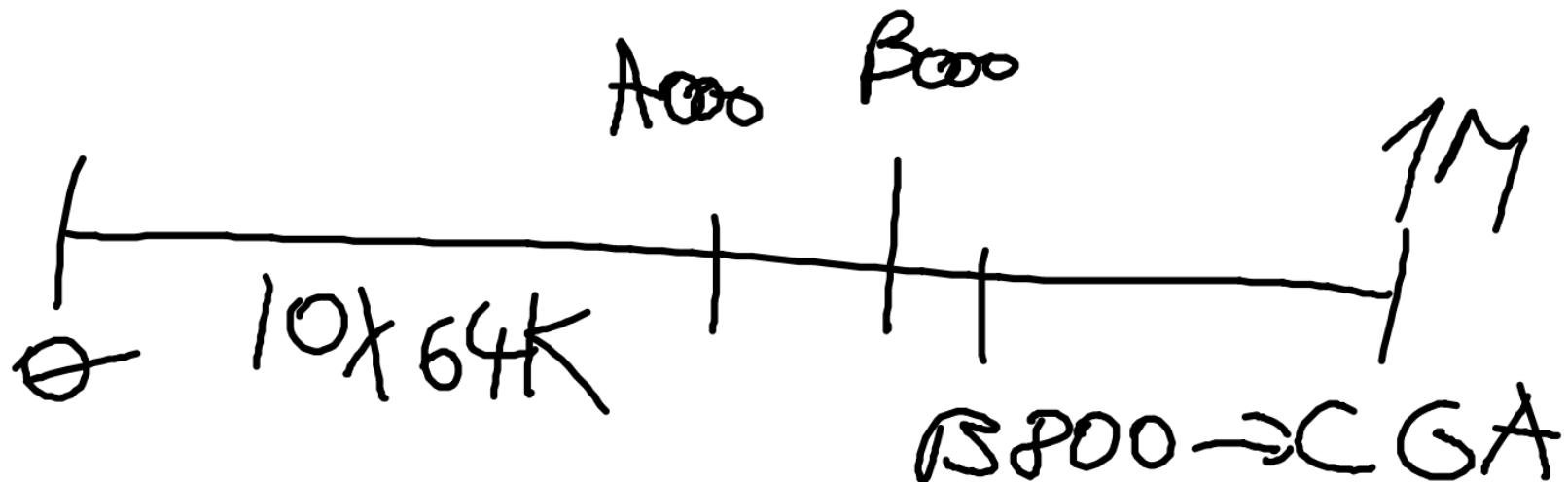
$\hookrightarrow Z \rightarrow$  nincs leütés

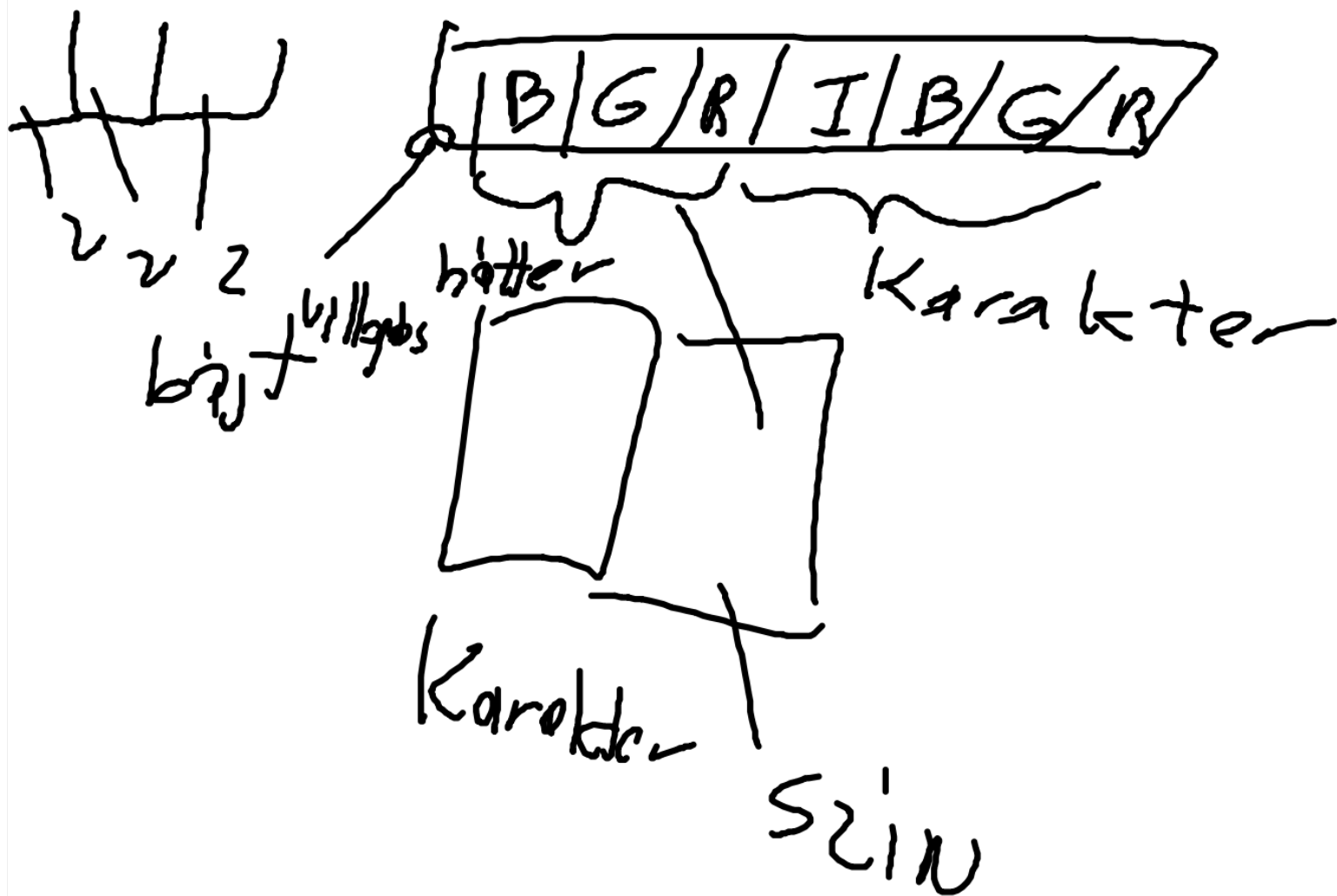
$\hookrightarrow NZ \rightarrow$  van leütés  
 $AL \rightarrow$  kód

B800

MOV AX, 0B800H

MUL ES, AX





MOV AX, 3

INT 33H

AND CX, 0FFFFH

AND DX, 0FFFFH

DIR<sub>L</sub> / P

DIR<sub>tab</sub> / P

DIR / P

ORG 100H

$16^2$   $16^1$   $16^0$

100

PARHOS  
PARHAT  
PARCHR

OR G 20H

D B 2

DIB 126 DUP (?)



ORG 10014

MOV SI, OFFSET PARH0

CMP byte ptr [SI], 0

CLD → MOV SI, 80H  
LODS B  
CMP AL, 3  
JNZ H1B A  
INC SI  
LODS B  
CMP AL, '0'  
JB H1B A  
CMP AL, '9'  
JA H1B A

MOV BL, AL

LODSB

CMP AL, 'H'

JNZ H1BA

LODSB

CMP AL, '0'

JB H1BA

CMP AL, '9'

JA H1BA

MOV AL, [ESI]  
INC SI

MOV BH, AL

MOV AH, 2

MOV DL, BL

INT 21H

MOV DL, 7

INT 21H

MOV DL, BH

INT 21H

MOV BL, DL

INT 21H

MOS

SUB BH, '0'  
SUB BL, '0'  
ADD BL, BH

Mov DL, BL  
ADD DL, '0'

INT 21H

MOV AL, BL

AAM

OR AH, AH

JZ EG

MOV BL, AL

MOV AH, 2

MOV DL, AH

OR DL, 10

INT

MOV

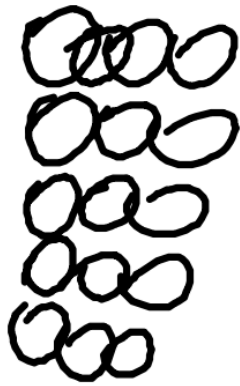
AL, BL

EGY:

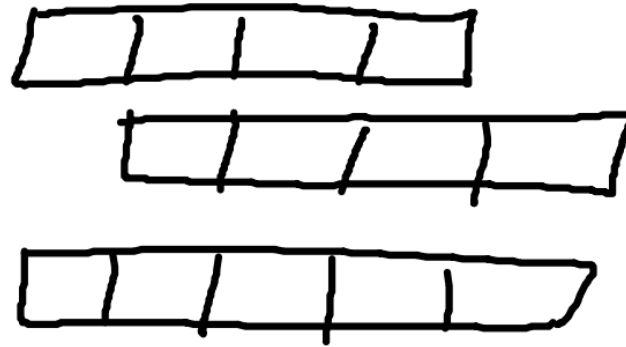
MOV D4, AL  
MOV AH, 2  
OR DL, '0'  
INT 21H

```
MOV SI, 02H
Ciklus: XOR DI, DI
        LODSB
        CMP AL, 13
        JZ RESZ
        AND AX, 11111B
        XCHG AX, DI
        MOV CX, 10
        MUL CX
        ADD DI, AX
        JMP Ciklus
```



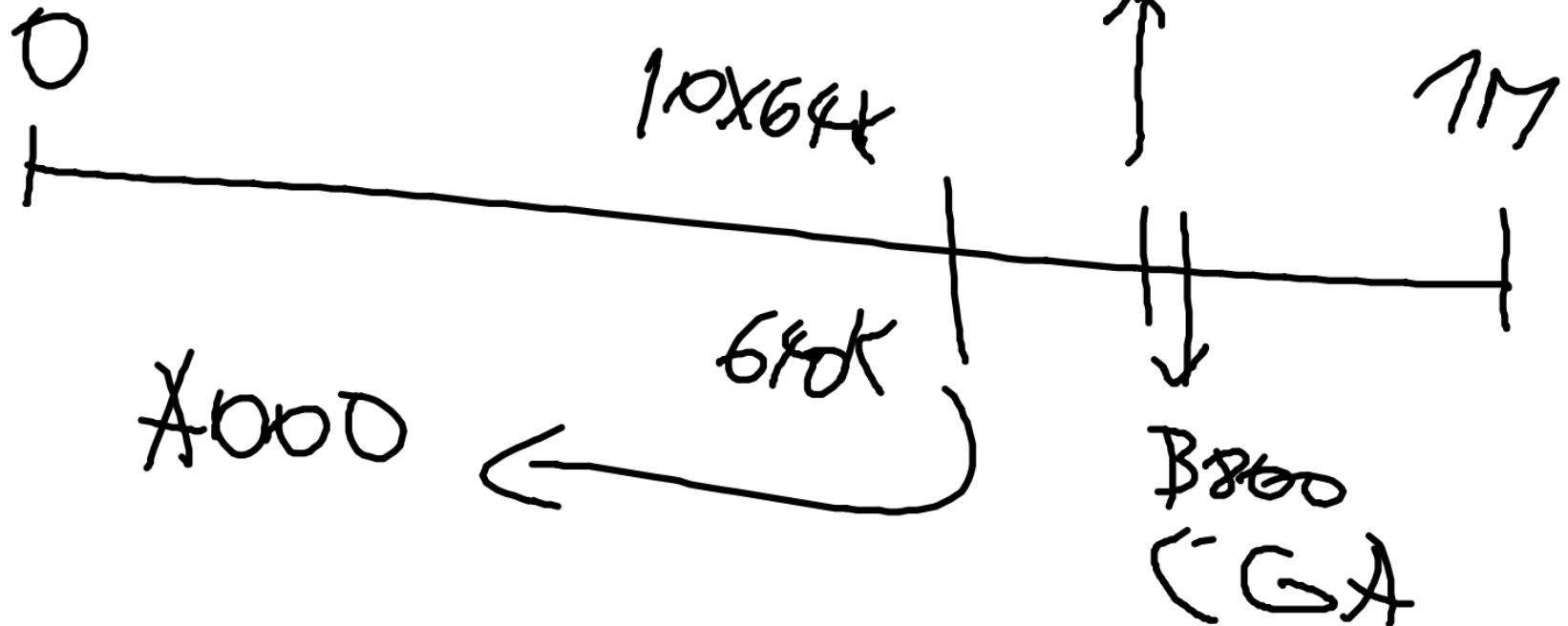


+



SEG  
OFFSET

Bobo  
MONOCROME



MC GA

320x200 256 színű

BH üzemmód

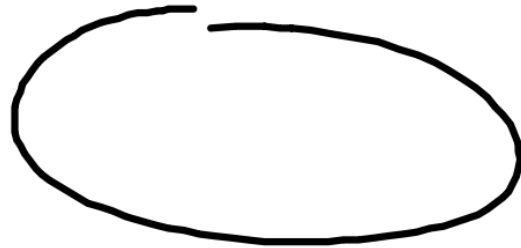
1/0 port színszámhoz: 3C8  
1/0 port színösszetevőkhöz: 3C9

MOV AX, 13H

INT 10H

MOV AX, 0A000H

MOV ES, AX

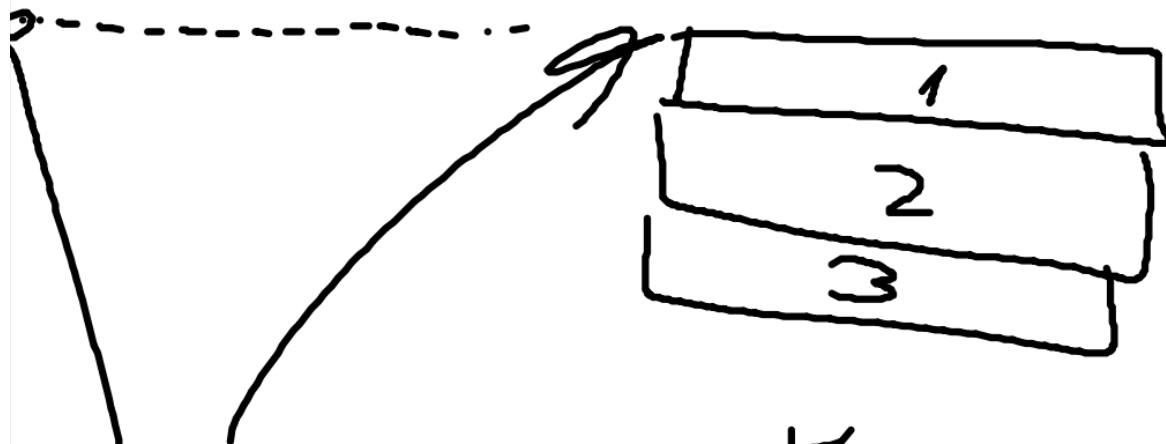


MOV AX, 3

INT 10H

SZELFS EQU 160

MAGAS EQU 33



KEZDOS EQU

$$((200 - (3 * MAGAS)) / 2) * 320$$
  
$$KEZDOS = ((200 - (3 * MAGAS)) / 2) * 320$$

SORRUT:

MOV CX, MAGAS

SORK1:

PUSH CX

MOV CX, SIZELES

REP STOSB

ADD DI, 320-SIZELES

POP CX

loop SORK1

MOV DI, KE2DOS + KE2D00

MOV AL, 1

SZINEX:

CALL SORTPUT

INC AL

CMP AL, 4

JNZ SZINEX

MOV DX, 3C8H

MOV AL, 1

OUT DX, AL

INC DX

MOV AL, 63

OUT DX, AL

MOV AL, 0

OUT DX, AL

OUT DX, AL

DEC DX

MOV AL, 2

OUT DX, AL

INC DX

MOV AL, 63

OUT DX, AL

OUT DX, AL

OUT DX, AL



```
DEC DX  
MOV AL, 3  
OUT DX, AL  
INC DX  
MOV AL, 6  
OUT DX, AL  
MOV AL, 63  
OUT DX, AL  
MOV AL, 0  
OUT DX, AL
```

```
MOV DX, OFFSET zz  
PUSH DX  
RET
```

zz:

MOV BX, OFFSET F12V

SHL word ptr [BX], 1

RCR word ptr [BX], 1

F12V:

DB 4 DUP(0)

MOV BX, OFFSET SASTON

MOV CX, 100

NULLA2:

MOV byte ptr [BX], 0

INC BX

LOOP NULLA2

SASTON

DB 100 DUP(?)