

Chöông 5 : Nhaäp moän Assembly

Muïc tieâu

- Hieäu ngoän ngöõ maùy vaø ngoän ngöõ Assembly.
- Trình hoiù dòch Assembler.
- Lyù do nghiêän cöùu Assembly.
- Hieäu caùc thaønh phaàn cô baün cuüa Assembly
- Naém ñöôic caáu truùc cuüa 1 CT Assembly.
- Bieát vieát 1 chöông trình Assembly.
- Bieát caùch dòch, lieän keát vaø thöïc thi 1 chöông trình Assembly.

Giòuì thieäu ngoân ngöõ Assembly

- Giuùp khaùm phaù bí maät phaàn cöùng cuõng nhö phaàn meàm maùy tính.
- Naém ñöôic caùch phaàn cöùng MT laøm vieäc vòuì heä ñieàu haønh vaø hieäu ñöôic baèng caùch naøo 1 trình öùng duïng giao tieáp vòuì heä ñieàu haønh.
- Moät MT hay moät hoï MT söù duïng 1 taäp leänh maõ maùy rieâng cuõng nhö 1 ngoân ngöõ Assembly rieâng.

Assembler

- Một chương trình viết bằng ngôn ngữ Assembly muốn MT thực hiện thì ta phải chuyển thành ngôn ngữ máy.
- Chương trình dùng để dịch 1 file viết bằng Assembly → ngôn ngữ máy, gọi là Assembler.

Có 2 chương trình dịch:

MASM và TASM

Lyù do nghiênn cõu Assembly

- Nõu laø caùch toát nhaát ñeã hoïc phaàn cõng MT vaø heã ñieàu haønh.
- Vì caùc tieän ích cuõa nõu .
- Cõu theã nhuùng caùc chõng trình con vieát baèng ASM vaøo trong caùcchõng trình vieát baèng ngoân ngõõ caáp cao .

Leãnh maùy

- Laø 1 chuoãi nhò phaân coù yù nghóa ñaëc bieät – noù ra leãnh cho CPU thöïc hieän taùc vuiï.
- Taùc vuiï ñoù coù theå laø :
di chuyeån 1 soá töø vò trí nhôù naøy sang vò trí nhôù
khaùc.
Coäng 2 soá hay so saùng 2 soá.

0 0 0 0 0 1 0 0 Add a number to the AL register

1 0 0 0 0 1 0 1 Add a number to a variable

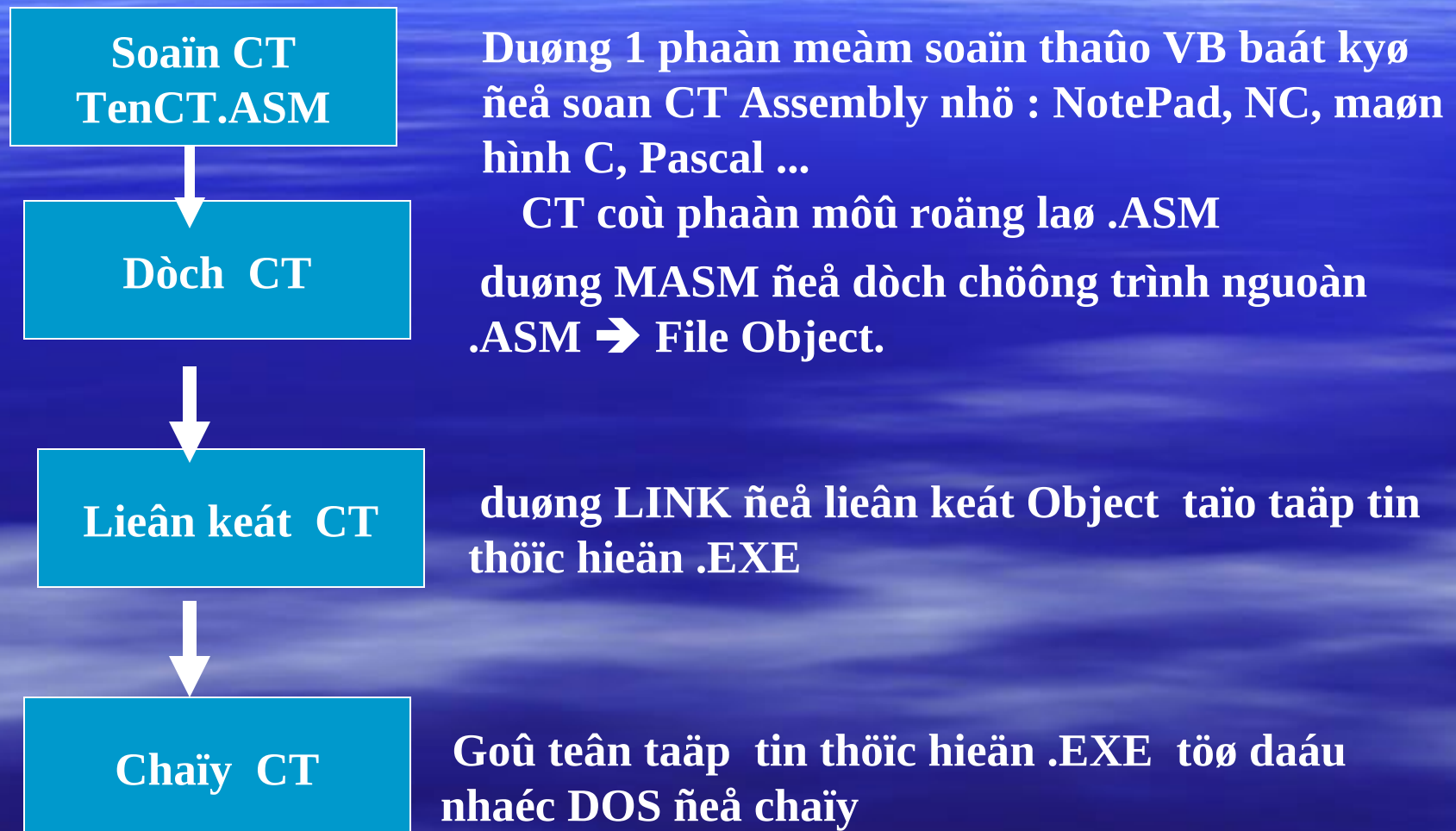
1 0 1 0 0 0 1 1 Move the AX reg to another reg

Leänh maùy (cont)

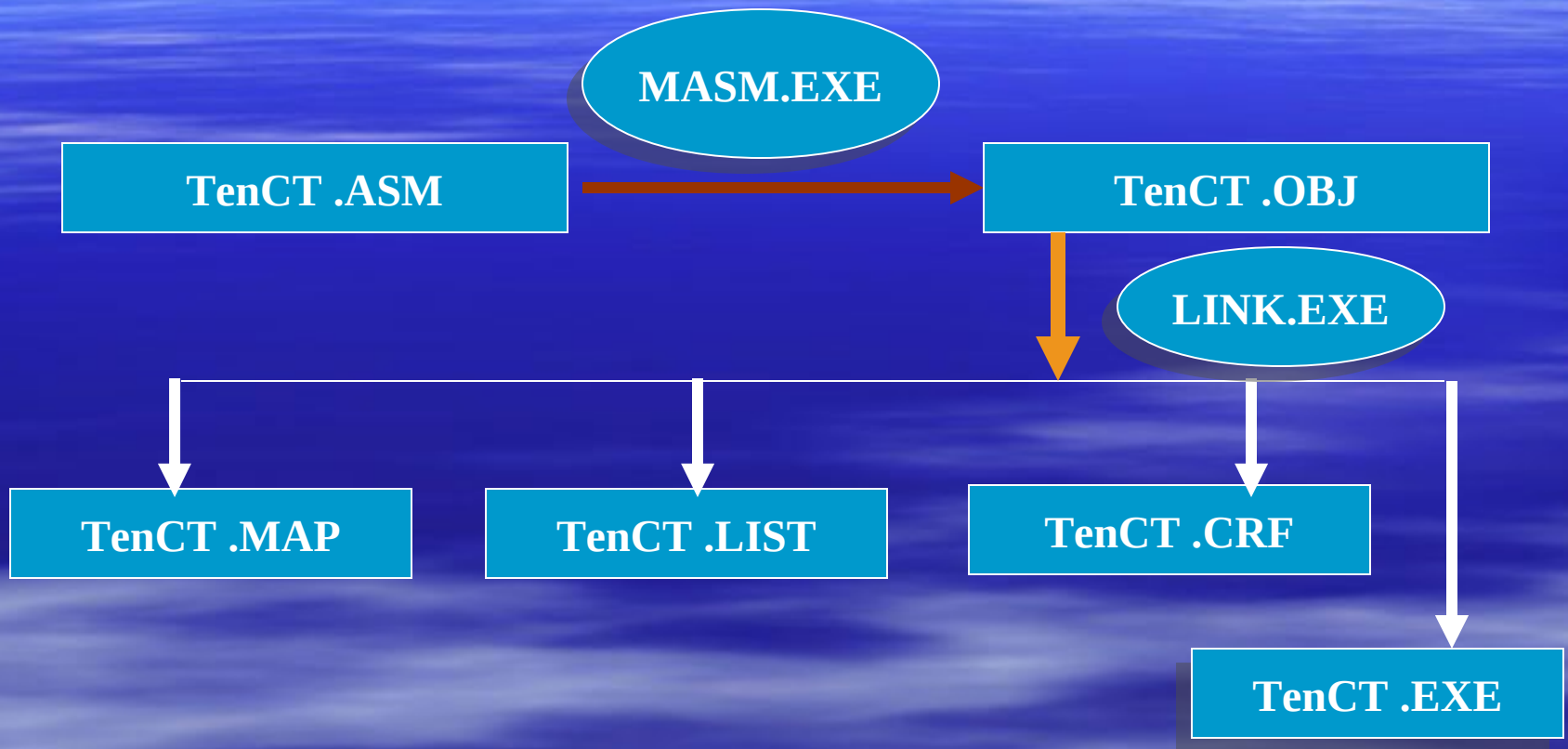
- Taäp leänh maùy ñöôïc ñònh nghóa tröôùc, khi CPU ñöôïc saün xuaát vaø nou ñaéc tröng cho kieäu CPU .
- Ex : B5 05 laø 1 leänh maùy vieát daïng soá hex, daøi 2 byte.
- Byte ñaàu B5 goïi laø Opcode
- Byte sau 05 goïi laø toaøn haïng Operand

YÙù nghóa cuüa leänh B5 05 : cheùp giaù trò 5 vaøo reg AL

Cách viết 1 chương trình Assembly



Dòch vaø noái keát chöông trình



Một chương trình minh họa

```
DOSSEG
.MODEL SMALL
.STACK 100h
.DATA
MES DB "HELLO WORD", '$'
.CODE
MAIN PROC
    MOV AX, @DATA
    MOV DS, AX
    MOV DX, OFFSET MES
    MOV AH, 9
    INT 21
    MOV AH, 4CH
    INT 21
MAIN ENDP
END MAIN
```

Caùc file ñhòic taõ

- Sau khi dòch thaønh công file nguoaøn.ASM, ta coù caùc file :
- File listing : file VB , caùc doøng coù ñaình soá thòu töi maõ.
- File Cross reference
- File Map
- File Obj
- File EXE

File Listing

Microsoft (R) Macro Assembler Version 5.10 10/11/4

Page 1-1

```
1          DOSSEG
2          .MODEL SMALL
3          .STACK 100H
4          .DATA
5 0000 48 45 4C 4C 4F 20      MES DB "HELLO WORD$"
6      57 4F 52 44 24
7          .CODE
8 0000          MAIN PROC
9 0000 B8 ---- R          MOV AX,@DATA
10 0003 8E D8          MOV DS, AX
11 0005 B4 09          MOV AH,9
12 0007 BA 0000 R          MOV DX, OFFSET MES
13 000A CD 21          INT 21H
14 000C B4 4C          MOV AH,4CH
15 000E CD 21          INT 21H
16 0010          MAIN ENDP
17          END MAIN
```

Microsoft (R) Macro Assembler Version 5.10 10/11/4

Map File

- **Start Stop Length Name Class**
- **00000H 0001FH 00020H _TEXT CODE**
- **00020H 0002AH 0000BH _DATA DATA**
- **00030H 0012FH 00100H STACK STACK**

- **Origin Group**
- **0002:0 DGROUP**

- **Program entry point at 0000:0010**

Giaûi thích

- .model small : duøng kieåu caáu truùc \leq 64 K boã nhòu cho mã , 64K cho döõ lieäu.
- .Stack 100h : daønh 256 bytes cho stack cuûa chöông trình .
- .Data : ñaøn hình daáu phaân ñoain döõ lieäu ôu ñòu caùc bieán ñöôic löu tröõ.
- .Code : ñaøn hình daáu phaân ñoain mã chöùa caùc leãnh phaûi thi haønh.
- Proc : khai baùo ñaàu 1 thuû tuïc, trong Ex naøy ta chæ coù 1 thuû tuïc Main.

Giaûi thích (cont)

- Cheùp ñòa chæ ñoain döõ lieäu vaøo thanh ghi AX.
- Sau ñoù cheùp vaøo thanh ghi DS
- Goïi haøm soá 9 cuûa Int 21h cuûa Dos ñeã xuaát chuoãi kyù töï ra maøn hình.
- Thoàùt khoûi CT .
- Main endp : ñaình daáu keát thuïc thuû tuïc
- End main : chaám döùt chöông trình



Caùc cheá ñoä boä nhôu

Kieâu	Moâ taû
SMALL	Maõ leänh trong 1 ñoäin. Döõ lieäu trong 1 ñoäin
MEDIUM	Maõ leänh nhieàu hôn 1 ñoäin. Döõ lieäu trong 1 ñoäin
COMPACT	Maõ leänh trong 1 ñoäin. Döõ lieäu nhieàu hôn 1 ñoäin
LARGE	Maõ leänh nhieàu hôn 1 ñoäin Döõ lieäu nhieàu hôn 1 ñoäin, khang coù maûng naøo > 64K
HUGE	Maõ leänh nhieàu hôn 1 ñoäin Döõ lieäu nhieàu hôn 1 ñoäin, maûng coù theä > 64K

Đãing leänh

Chuù thíc

■ [name] [operator] [operand] [comment]

Nhaõn, teân bieán
Teân thuû tuíc

Maõ leänh đãing
gõii nhòu

Register, oâ nhòu
Trò, haèng

Ex : MOV CX , 0

LAP : MOV CX, 4

LIST DB 1,2,3,4

Moãi dòng chæ chòu 1 leänh vaø mỗi leänh phaûi naèm trên 1 dòng

INT 21H

- Leãnh INT soá hieãu ngaét ñöôic duøng ñeã goïi chöông trình ngaét cuûa DOS vaø BIOS.

Ngaét 21h

Muoán söu duïng haøm naøo cuûa INT 21h ta ñaët function_number vaøo thanh ghi AH, sau ñoù goïi INT 21h

Function_number

chöùc naêng

1	nhaäp 1 kyù töi töø baøn phím
2	Xuaát 1 kyù töi ra maø hình.
9	Xuaát 1 chuoãi kyù töi ra maøn hình

INT 21h (cont)

Hàm 1 : Nhập 1 ký tự

Input : AH = 1

Output : AL = mã ASCII của phím ấn

= 0 nếu 1 phím nào khi ấn

Hàm 2 : Hiển thị 1 ký tự ra màn hình

Input : AH = 2

DL = Mã ASCII của ký tự hiển thị hay ký tự nào khi ấn

Thí dụ minh hoã

```
DOSSEG
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
    MOV AH , 2
    MOV DL , '?'
    INT 21H
    MOV AH , 1
    INT 21H
    MOV BL,AL
```

```
    MOV AH,2
    MOV DL, 0DH
    INT 21H
    MOV DL , 0AH
    INT 21H
    MOV DL , BL
    INT 21H
    MOV AX , 4C00H
    INT 21H
    MAIN ENDP
END MAIN
```

KEÁT QUAÛ

? N
N

Thí dụ minh họa cách hoạt động của INT 21

- In dấu ? ra màn hình :

```
MOV AH, 2
```

```
MOV DL, '?'
```

```
INT 21H
```

- Nhập 1 ký tự rồi chờ ấn phím :

```
MOV AH, 1
```

```
INT 21H
```

Biến

- Cuù phaùp : **[teân bieán] DB | DW |... [trò khôûi taïo]**
- Laø moät teân kyù hieäu daønh rieâng cho 1 vò trí trong boä nhòu nôï löu tröõ döõ lieäu.
- Offset cuûa bieán laø khoaûng caùch töø ñaàu phaân ñoain ñeán bieán ñoù.
- Ex : khai baùo 1 danh saùch aList ôû ñòa chæ 100 vòuï noãi dung sau :

.data

aList db “ABCD”

Bieán (cont)

Luèc ñoù :

Offset	Bieán
0000	A
0001	B
0002	C
0003	D

Khai báo biến

Tөр gөii nhөu	Moä taü	Soá byte	Thuoäc tính
DB	Ñöh nghóa byte	1	Byte
DW	Tөр	2	Word
DD	Tөр keùp	4	Doubleword
DQ	Tөр töu	8	Quardword
DT	10 bytes	10	tenbyte

Minh hoã khai baùo bieán

KIEAU BYTE

- **Char db 'A'**
- **Num db 41h**
- **Mes db "Hello Word", '\$'**
- **Array_1 db 10, 32, 41h, 00100101b**
- **Array_2 db 2,3,4,6,9**
- **Myvar db ? ; bieán khoâng khôûi taïo**
- **Btable db 1,2,3,4,5
db 6,7,8,9,10**

Minh hoĩa khai baùo bieán

KIEÁU WORD

DW 3 DUP (?)

DW 1000h, 'AB', 1024

DW ?

DW 5 DUP (1000h)

DW 256*2

DAĨNG LÖU TRÖÖ DÖÖ LIEÁU KIEÁU WORD :

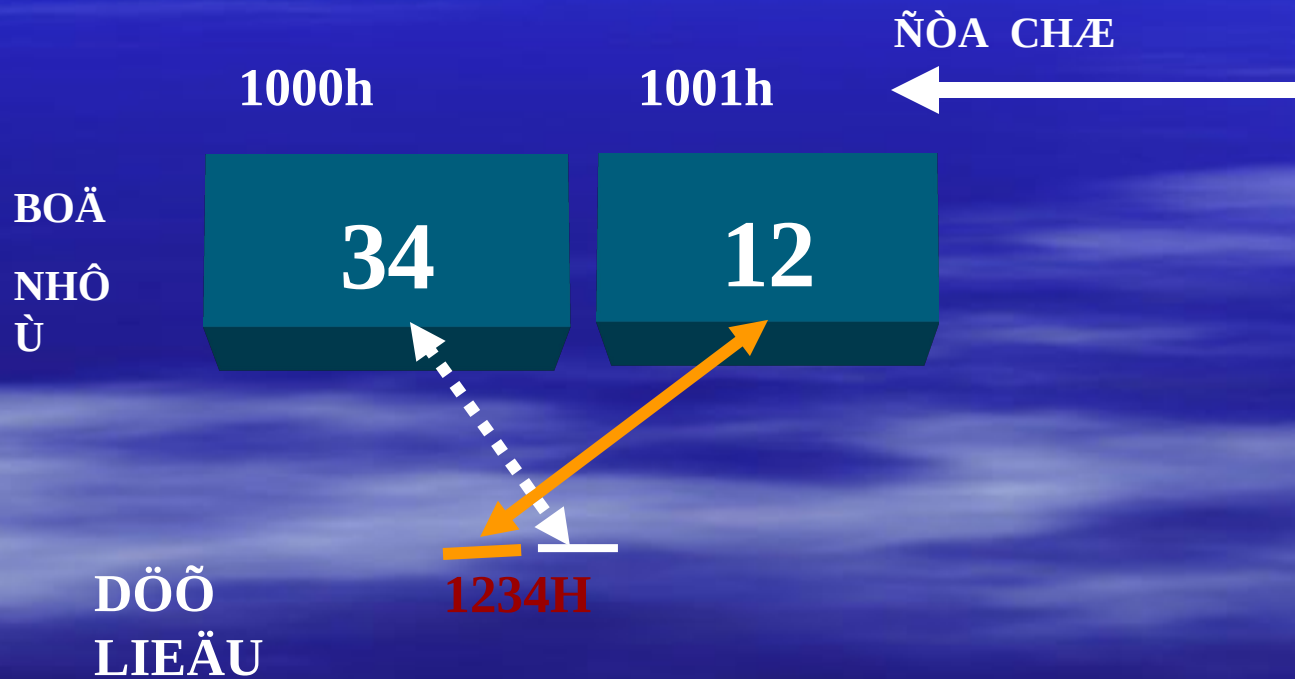
Trình hõip dòch ñaũo ngöõic cauc byte trong 1 giaù trò kieáu WORD khi löu tröõ trong boä nhòu :

Byte thaáp löu ôu ñòa chæ thaáp Byte cao löu ôu ñòa chæ cao

Minh hoã khai baùo bieán

KIEÅU WORD

Ex : 1234h ñöôic löu tröõ trong boã nhöù nhö sau :



Toaùn tồu DUP

- Laëp laii 1 hay nhieàu giaù trò khôù taïo.

- Ex :

```
Bmem DB 50 Dup(?)
```

```
; khai baùo vuøng nhôù goàm 50 bytes.
```

```
db 4 dup ("ABC")
```

```
;12 bytes "ABCABCABCABC"
```

```
db 4096 dup (0)
```

```
; Vuøng ñeãm 4096 bytes taát caù baèng 0
```


Khôûi taïo bieán

- Löu yù :

Khi khôûi taïo trò laø 1 soá hex thì giaù trò soá luoân luoân baét ñaàu baèng 1 kyù soá töø 0 ñeán 9. Neáu kyù soá baét ñaàu laø A.. F thì phaûi theâm soá 0 ôû ñaàu.

- Ex :

Db A6H ; sai

Db 0A6h ; ñuùng

Toaùn töô DUP (cont)

Amtrix dw 3 dup (4 dup (0))

Taïo 1 ma traän 3x4

Atable db 4 dup (3 dup (0), 2 dup ('X'))

Taïo 1 vường nhòu chöua 000XX 000XX 000XX 000XX

Toàn tồ DUP

- Chæ xuaát hiễn sau 1 chæ thò DB hay DW
- Vôùi DUP ta còu theả laép laiì 1 hay nhieàu trò cho vương nhòu.
- Raát còu ích khi laøm vieäc vôùi maûng hay chuoải.

Toàùn töü ?

- Muốn khai baø 1 bieán hay 1 maúng maø khoâng caàn khôûi taío trò ta duøng toàùn töü ?

Ex : MEM8 DB ? ; khai baø 1 byte troáng trong boä nhòu

MEM16 DW ? ; khai baø 2 byte troáng trong boä nhòu

BMEM DB 50 DUP(?)

; khai baø 50 byte troáng trong boä nhòu

Chöông trình daing .COM

CODE SEGMENT

ASSUME CS:CODE , DS:CODE, SS:CODE

; toaøn boã chöông trình chæ naèm trong 1 segment

Org 100h ;; chæ thò naip thanh ghi leãnh IP=100h khi
CT ñöôic naip

Main proc

mov ax,bx

.....

Main endp

Count db 10

.....

Code ends

End main

SUMMARY

- chương trình Assembly gồm nhiều dòng lệnh.
- Mỗi lệnh phải viết trên 1 dòng
- Lệnh có thể gồm [tên] [toán tử] [toán hạng]
- Các ký tự phải đặt trong dấu ‘ ‘ hay “ ”
- DB dùng để định nghĩa biến kiểu BYTE
- DW dùng để định nghĩa biến kiểu WORD.

Caâu hoûi ôn taäp

- Trong maõ maùy döõuì ñaây ñöôïc laáy töø taäp tin lieät keâ, haõy neâu yù nghóa cuûa R
- Neâu yù nghóa cuûa kyù hieäu ñoà chæ cuûa bieán döõuì ñaây trong 1 taäp tin lieät keâ.

```
5B 0021  R          ADD BX, VAL1
```

Caâu hoûi ôn tập

- Chöông trình sau cöù loãi. Haõy tìm caâu leãnh naøo gaây ra loãi, giaûi thích vaø söûa laïi cho ñuùng.

```
.MODEL SMALL
```

```
.STACK 100H
```

```
.DATA
```

```
MOV AX, VALUE1
```

```
MOV BX, VALUE2
```

```
INC BX, 1
```

```
INT 21H
```

```
MOV 4C00H, AX
```

```
MAIN ENDP
```

```
VALUE1    0AH
```

```
VALUE2    1000H
```

```
END MAIN
```


- Chương trình sau có lỗi. Hãy tìm câu lệnh nào gây ra lỗi, giải thích và sửa lỗi cho đúng.

Chương hỏi ôn tập

```
.MODEL SMALL
.STACK 100H
.CODE
MAIN PROC
    MOV AX, @DATA
    MOV DS, AX
    MOV AX, VALUE1
    MOV AX, VALUE2
    MOV AX, 4C00H
INT 21H
MAIN ENDP

VALUE1 DB 0AH
VALUE2 DB 1000H

END MAIN
```

Bài tập lập trình

Bài 1 : Viết chương trình nhập 1 ký tự rồi in ra ký tự hoa tương ứng.

Bài 2 : Viết chương trình hoàn và 2 biến kiểu byte rồi gán số 1 và 2.

Bài 3 : Viết chương trình tạo 1 array có các phần tử 31h,32h,33h,34h.

Đưa tổng phần tử vào thanh ghi DL và xuất nó ra màn hình. Giải thích tại sao kết quả xuất trên màn hình là 1234.