

Corso di Architettura degli Elaboratori
Modulo di Assembly

ASSEMBLATORE

Bruno Iafelice

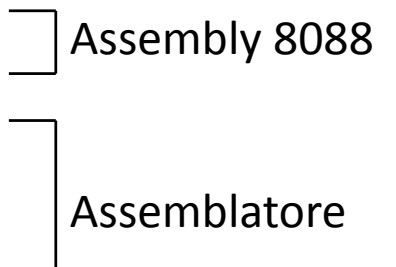
University of Bologna

iafelice at cs(dot)unibo(dot)it

ASSEMBLATORE

- Disponibile presso:
 - CD-ROM allegato al libro di testo del corso
 - INTERNET `ftp://ftp.cs.vu.nl/pub/evert`
- Il tool comprende:
 - Emulatore-Interprete dell'architettura 8088 (s88)
 - Invocazione da linea di comando: `s88 Nomeprogetto`
 - Programma assembler (as88)
 - Invocazione da linea di comando : `as88 Nomeprogetto(.s)`
 - Programma tracer per il debugging (t88)
 - Invocazione da linea di comando : `t88 Nomeprogetto`

Un programma assembly è una sequenza di:

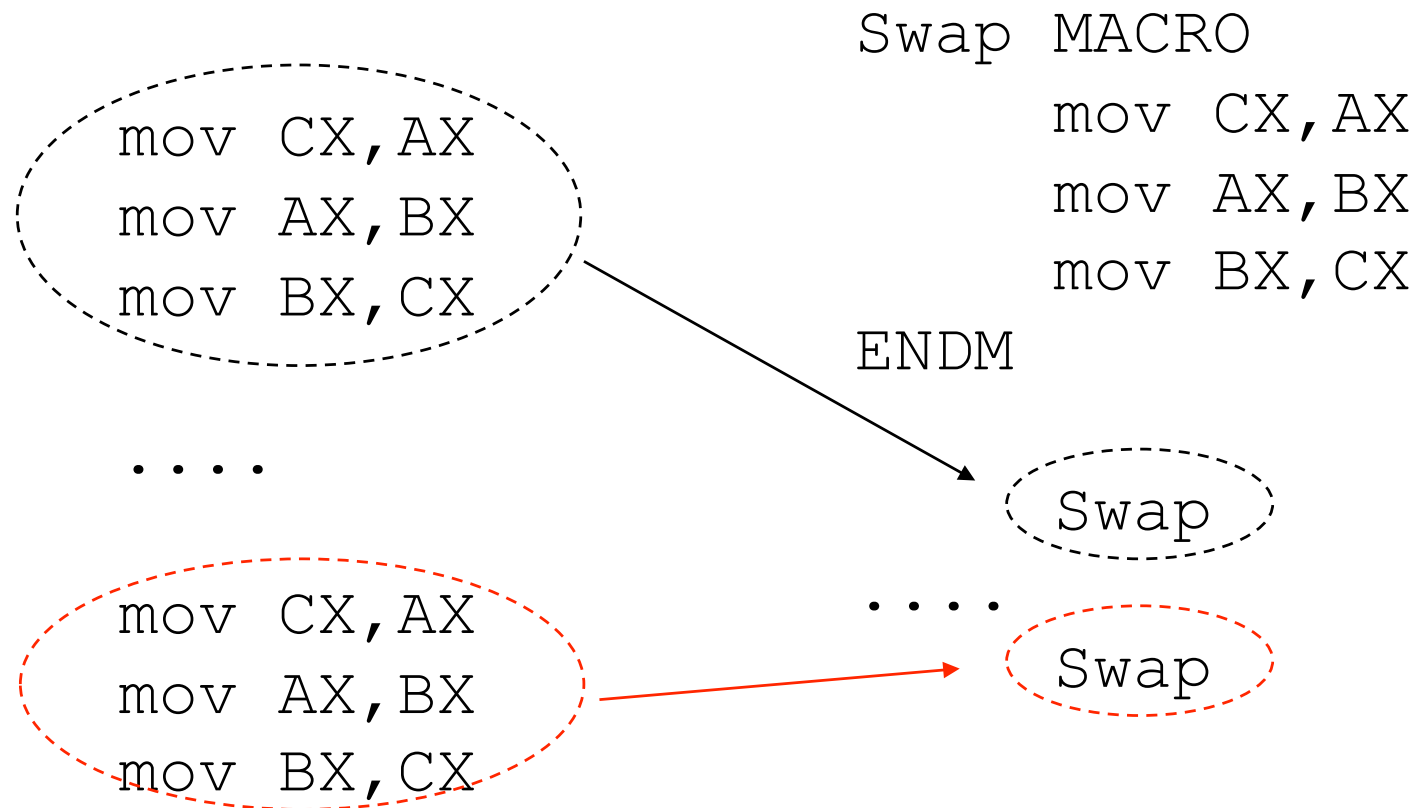
- Istruzioni dell'Architettura di riferimento
 - **Macro dell'assemblatore di riferimento**
 - **Direttive dell'assemblatore di riferimento**
- 

Direttiva: Controlla il comportamento dell'assemblatore in fase di traduzione.
Non necessariamente e' tradotta in codice macchina.

Macro non presenti con as88

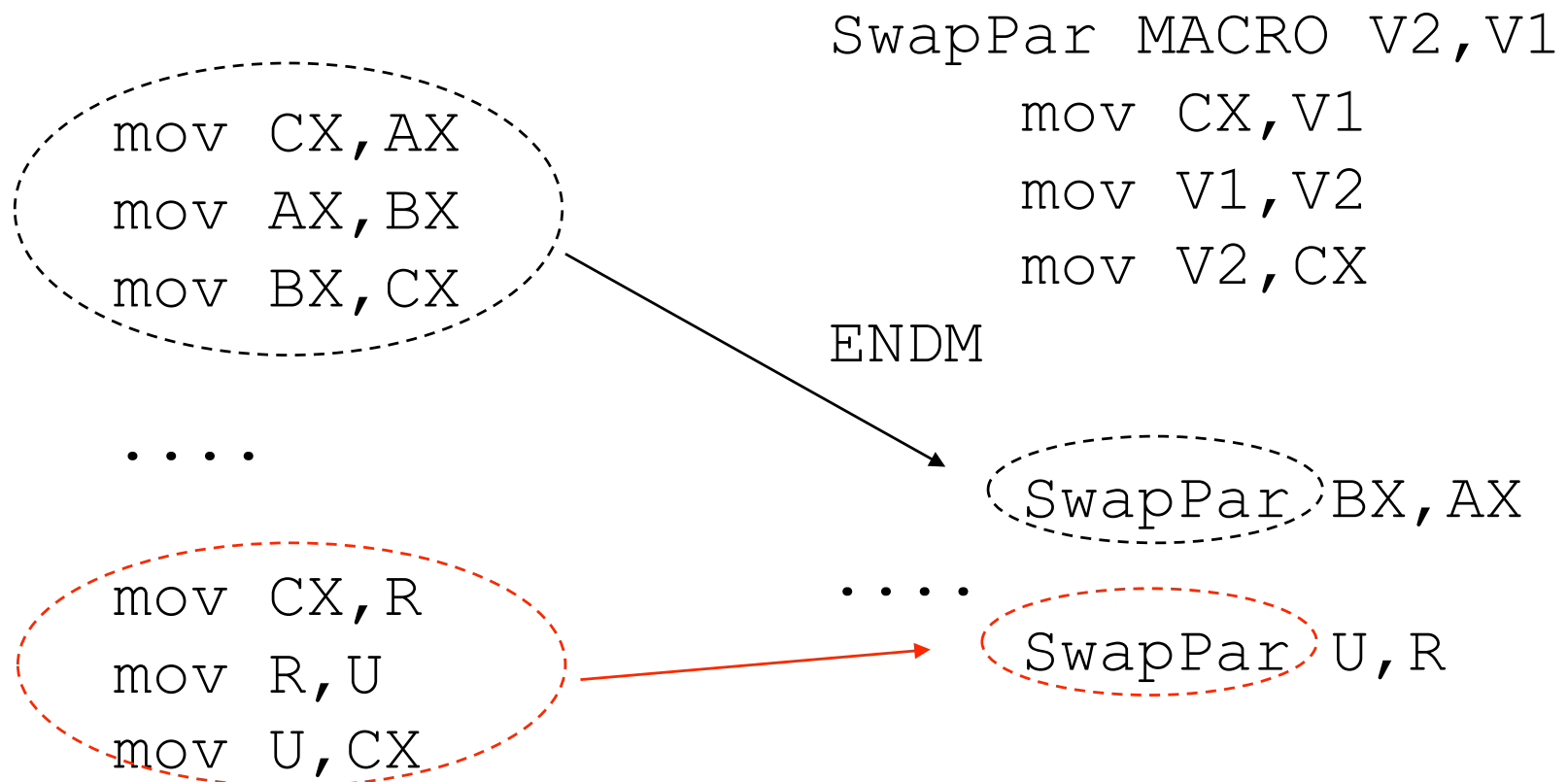
MACRO

- Per facilitare la ripetizione di codice



Macro parametriche

- Aggiungiamo la possibilità di passare parametri alla macro (Sottoprocedura vs. Macro ????)



DIRETTIVE

- Istruzioni di tipo dichiarativo che forniscono informazioni agli strumenti di sviluppo:
Assemblatore e Linker
- Direttive di Segmento
- Direttive per la Definizione di Procedure
- Direttive per la Definizione dei Dati

Istruzione	Descrizione
SEC .TEXT	Assembla le linee seguenti nella sezione TESTO
SEC .DATA	Assembla le linee seguenti nella sezione DATI
SEC .BSS	Assembla le linee seguenti nella sezione BSS
.BYTE	Assembla gli argomenti come una sequenza di byte
.WORD	Assembla gli argomenti come una sequenza di parole
.LONG	Assembla gli argomenti come una sequenza di long
.ASCII "str"	Salva la stringa str in ASCII senza farla terminare per zero
.ASCIIZ "str"	Salva la stringa str in ASCII facendola terminare per zero
SPACE n	Incrementa il contatore di locazioni di n posizioni ←
ALIGN n	Allinea il contatore alla locazione che dista al più n byte
EXTERN	L'identificatore è un nome esterno

BYTE

EXTERN indica al LINKER le variabili e procedure esterne.

Direttive di Segmento

`SECT segment`

- sezione di TESTO: (direttiva: `SECT .TEXT`) contiene le istruzioni del programma
- sezione DATI: (direttiva `SECT .DATA`) alloca spazio nel segmento DATI per i dati (inizializzati)
- sezione BSS: (direttiva `SECT .BSS`) alloca spazio nel segmento DATI per i dati (non inizializzati)

Etichette

- E' possibile definire etichette di due tipi:
 - **Globali** identificatori alfanumerici seguiti dal carattere ":"
Es. ELABORA:
 - **Locali** utilizzabili solo nel segmento TESTO corrente e nella procedura corrente, costituite da una sola cifra seguita da ":"
Es. L:

Direttive per la definizione di dati

- Nella forma:

```
nome: TIPO contenuto
```

- Tipi:

- .WORD word
- .BYTE byte
- .ASCII char

- Esempi .DATA:

```
contatore: .WORD 0
```

```
array: .WORD 1,2,3,4
```

- Esempio .BSS:

```
buffer: .SPACE 80 !buffer di 80 byte
```

Stringhe

```
MOVB AL, format(SI)
```

- Carica in AL il carattere della stringa *“format”* alla posizione SI.
- Esempio:
 - SI = 5
 - Carica in AL il **quinto** carattere della stringa *“format”*

TRACER

- Debugging
- Esecuzione “passo-passo”
- Visione di cosa accade a livello macchina

Processore e registri	Stack	Testo del programma File Sorgente
Stack per le chiamate di subroutine	Area dei messaggi d'errore Area di input Area di output	
Interprete dei comandi		
Valore delle variabili globali Segmento dati		

- E' possibile interagire con il tracer in modalità batch (fornendo in input un file con i comandi del tracer)
- oppure in modalità interattiva (inserendo comandi da tastiera uno alla volta)

Indirizzo	Comando	Esempio	Descrizione
			Esegue una istruzione
#	, !, X	24	Esegue # istruzioni
/T+#	g, !,	/start+5g	Continua fino alla riga # dopo etichetta T
/T+#	b	/start+5b	Inserisce breakpoint alla riga # dopo etichetta T
/T+#	c	/start+5c	Rimuove breakpoint alla riga # dopo etichetta T
#	g	108g	Esegue il programma fino alla riga #
	g	g	Esegue il programma finché non torna alla riga corrente
	b	b	Inserisce breakpoint alla riga corrente
	c	c	Rimuove breakpoint dalla riga corrente