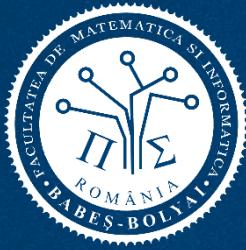


Arhitectura Sistemelor de Calcul

Lect. Dr. Șotropa Diana
diana.sotropa@ubbcluj.ro



Facultatea de Matematică și Informatică
Universitatea Babeș-Bolyai





Instructiuni pe siruri

Caracteristici ale șirurilor

- Byte (B)
- Word (W)
- Doubleword (D)

Tipul elementelor

- DS:ESI – pentru șirul sursă
- ES:EDI – pentru șirul destinație

Adresa elementelor

- CLD \Leftrightarrow DF=0 \Leftrightarrow De la adrese mici la adrese mari (de la stânga la dreapta)
- STD \Leftrightarrow DF=1 \Leftrightarrow De la adrese mari la adrese mici (de la dreapta la stânga)

Direcția de parcurgere

Caracteristici ale șirurilor

LODS (Load from string) – încarcă B / W / D din memorie de la adresa DS:ESI în registrul AL, AX sau EAX

STOS (Store string) – stochează conținutul registrului AL, AX sau EAX în memorie la adresa ES:EDI

MOVS (Move string data) – copiază B / W / D din memorie de la adresa DS:ESI în B / W / D din memorie de la adresa ES: EDI

CMPS (Compare strings) – compară conținutul B / W / D din memorie de la adresa DS:ESI cu B / W / D din memorie de la adresa ES: EDI

SCAS (Scan string) – compară registrul AL, AX sau EAX cu B / W / D din memorie de la adresa ES: EDI

Caracteristici ale șirurilor

LODS (Load from string) – încarcă B / W / D din memorie de la adresa DS:ESI în registrul AL, AX sau EAX

- **LODSB** – octetul de la adresa **DS:ESI** este încărcat în **AL**
 - Dacă DF = 0 atunci ESI = ESI + 1 altfel ESI = ESI - 1
- **LODSW** – cuvântul de la adresa **DS:ESI** este încărcat în **AX**
 - Dacă DF = 0 atunci ESI = ESI + 2 altfel ESI = ESI - 2
- **LODSD** – dublucuvântul de la adresa **DS:ESI** este încărcat în **EAX**
 - Dacă DF = 0 atunci ESI = ESI + 4 altfel ESI = ESI - 4

Caracteristici ale șirurilor

STOS (Store string) – stochează conținutul registrului AL, AX sau EAX în memorie la adresa ES:EDI

- **STOSB** – octetul din registrul **AL** este stocat în octetul de la adresa **ES:EDI**
 - Dacă DF = 0 atunci EDI = EDI + 1 altfel EDI = EDI – 1
- **STOSW** – cuvântul din registrul **AX** este stocat în cuvântul de la adresa **ES:EDI**
 - Dacă DF = 0 atunci EDI = EDI + 2 altfel EDI = EDI – 2
- **STOSD** – dublucuvântul din registrul **EAX** este stocat în dublucuvântul de la adresa **ES:EDI**
 - Dacă DF = 0 atunci EDI = EDI + 4 altfel EDI = EDI – 4

Caracteristici ale șirurilor

MOVS (Move string data) – copiază B / W / D din memorie de la adresa DS:ESI în B / W / D din memorie de la adresa ES: EDI

- **MOVSB** – copiază **octetul** din memorie de la adresa **DS:ESI** în **octetul** din memorie de la adresa **ES:EDI**
 - Dacă DF = 0 atunci ESI = ESI + 1 și EDI = EDI + 1,
altfel ESI = ESI – 1 și EDI = EDI – 1
- **MOVSW** – copiază **cuvântul** din memorie de la adresa **DS:ESI** în **cuvântul** din memorie de la adresa **ES:EDI**
 - Dacă DF = 0 atunci ESI = ESI + 2 și EDI = EDI + 2,
altfel ESI = ESI – 2 și EDI = EDI – 2
- **MOVSD** – copiază **dublucuvântul** din memorie de la adresa **DS:ESI** în **dublucuvântul** din memorie de la adresa **ES:EDI**
 - Dacă DF = 0 atunci ESI = ESI + 4 și EDI = EDI + 4,
altfel ESI = ESI – 4 și EDI = EDI – 4

Caracteristici ale șirurilor

CMPS (Compare strings) – compară conținutul B / W / D din memorie de la adresa DS:ESI cu B / W / D din memorie de la adresa ES: EDI

- **CMPSB** – compară **octetul** din memorie de la adresa **DS:ESI** cu **octetul** din memorie de la adresa **ES:EDI**
 - Dacă DF = 0 atunci ESI = ESI + 1 și EDI = EDI + 1,
altfel ESI = ESI – 1 și EDI = EDI – 1
- **CMPSW** – compară **cuvântul** din memorie de la adresa **DS:ESI** cu **cuvântul** din memorie de la adresa **ES:EDI**
 - Dacă DF = 0 atunci ESI = ESI + 2 și EDI = EDI + 2,
altfel ESI = ESI – 2 și EDI = EDI – 2
- **CMPSD** – compară **dublucuvântul** din memorie de la adresa **DS:ESI** cu **dublucuvântul** din memorie de la adresa **ES:EDI**
 - Dacă DF = 0 atunci ESI = ESI + 4 și EDI = EDI + 4,
altfel ESI = ESI – 4 și EDI = EDI – 4

Caracteristici ale șirurilor

SCAS (Scan string) – compară registrul AL, AX sau EAX cu B / W / D din memorie de la adresa ES: EDI

- **SCASB** – compară **octetul** din registrul **AL** cu **octetul** de la adresa **ES:EDI**
 - Dacă DF = 0 atunci EDI = EDI + 1 altfel EDI = EDI – 1
- **SCASW** – compară **cuvântul** din registrul **AX** cu **cuvântul** de la adresa **ES:EDI**
 - Dacă DF = 0 atunci EDI = EDI + 2 altfel EDI = EDI – 2
- **SCASD** – compară **dublucuvântul** din registrul **EAX** cu **dublucuvântul** de la adresa **ES:EDI**
 - Dacă DF = 0 atunci EDI = EDI + 4 altfel EDI = EDI – 4

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0 ⇔ directia de parcurgere de la stanga la dreapta
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir; ESI = offset sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB; AL = 88h; ESI = ESI + 1
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW; AX = 6677h, ESI = ESI + 2
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD; EAX = 22334455h, ESI = ESI + 4

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r; EDI = offset r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah; AL = 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB; byte [EDI]=[r]=1Ah, EDI = EDI + 1
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h; AX = 1234h
STOSW
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW; word [EDI] = [r+1] = 1234h, EDI = EDI + 2
MOV EAX, 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh; EAX = 567890cdh
STOSD
```

Caracteristici ale șirurilor

```
segment data ...
sir dq 1122334455667788h, 1a2b3c4d5e6faabbh
lung_sir equ ($-sir)/8
r times lun_sir dq 0
segment code...
CLD; DF = 0
MOV ESI, sir
LODSB
LODSW
LODSD

MOV EDI, r
MOV AL, 1Ah
STOSB
MOV AX, 1234h
STOSW
MOV EAX, 567890cdh
STOSD; dword [EDI] = [r+3] = 567890cdh
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1
MOV EDI, b
ADD EDI, lb - 1
STD
CMPSB
JE equal
JNE nonEqual
```

```
MOV AX, 0FFFFh
SCASW
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1; ESI = offset a + 2
MOV EDI, b
ADD EDI, lb
STD
CMPSB
JE equal
JNE nonEqual
```

```
MOV AX, 0FFFFh
SCASW
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1
MOV EDI, b
ADD EDI, lb - 1; EDI = offset b + 2
STD
CMPSB
JE equal
JNE nonEqual
```

```
MOV AX, 0FFFFh
SCASW
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1
MOV EDI, b
ADD EDI, lb - 1
STD; DF = 1
CMPSB
JE equal
JNE nonEqual
```

```
MOV AX, 0FFFFh
SCASW
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1
MOV EDI, b
ADD EDI, lb - 1
STD
CMPSB; "CMP 11h, 56h", ESI = ESI - 1, EDI = EDI-1
JE equal
JNE nonEqual
```

```
MOV AX, 0FFFFh
SCASW
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1
MOV EDI, b
ADD EDI, lb - 1
STD
CMPSB
JE equal
JNE nonEqual
```

MOV AX, 0FFFFh; AX = FFFFh

```
SCASW
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Caracteristici ale șirurilor

```
segment data ...
a db 10h, 12h, 11h
la equ $-a
b db 98h, 34h, 56h
lb equ $-b
segment code...
MOV ESI, a
ADD ESI, la - 1
MOV EDI, b
ADD EDI, lb - 1
STD
CMPSB
JE equal
JNE nonEqual
```

```
MOV AX, 0FFFFh
SCASW; "CMP AX, 5634h", EDI =
EDI - 2
JG maiMare
JLE maiMic
equal:
...
nonEqual:
...
maiMare:
...
maiMic:
...
```

Prefixe de instrucțiuni repetitive

REP – Repeat while ECX > 0

REPZ, REPE – Repeat while
ZF = 1 and ECX > 0

REPNZ, REPNE – Repeat
while ZF = 0 and ECX > 0

Exemplu: copierea unui sir de cuvinte într-un alt sir

```
segment data ...
s dw 1234h, 5678h
ls equ ($-s)/2
d times ls dw 0
segment code ...
MOV ECX, ls
MOV ESI, 0
MOV EDI, 0
JECXZ final
repeta:
    MOV AX, [s+ESI]
    MOV [d+EDI], AX
    add ESI, 2
    add EDI, 2
LOOP repeta
final:
```

Varianta 1

Exemplu: copierea unui sir de cuvinte într-un alt sir

```
segment data ...
s dw 1234h, 5678h
ls equ ($-s)/2
d times ls dw 0
segment code ...
MOV ECX, ls
MOV ESI, s
MOV EDI, d
JECXZ final
CLD
repeta:
    LODSW
    STOSW
LOOP repeta
final:
```

Varianta 2

Exemplu: copierea unui sir de cuvinte într-un alt sir

```
segment data ...
s dw 1234h, 5678h
ls equ ($-s)/2
d times ls dw 0
segment code ...
MOV ECX, ls
MOV ESI, s
MOV EDI, d
JECXZ final
CLD
repeta:
    MOVSW
LOOP repeta
final:
```

Varianta 3

Exemplu: copierea unui sir de cuvinte într-un alt sir

```
segment data ...
s dw 1234h, 5678h
ls equ ($-s)/2
d times ls dw 0
segment code ...
MOV ECX, ls
MOV ESI, s
MOV EDI, d
JECXZ final
CLD
REP MOVSW
final:
```

Varianta 4



FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
UNIVERSITATEA BABEŞ-BOLYAI

Str. Mihail Kogălniceanu nr. 1
Cluj-Napoca, Cluj, România

www.cs.ubbcluj.ro