

Assembleur

Chapitre 3

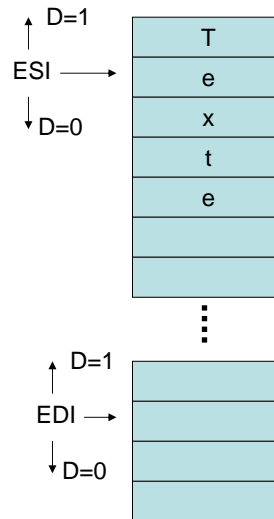
Paul Ghobril

La manipulation des chaînes de caractères

- On a souvent besoin de copier une chaîne ou une partie d'une chaîne dans une autre, d'effectuer une concaténation de chaînes, d'insérer une chaîne dans une autre, d'en extraire une suite de caractères, etc...
- Toutes ces opérations effectuées sur des blocs de données sont directement programmables en langage machine par des instructions spéciales. Ce sont les instructions de manipulation de chaînes de caractères.

Les instructions de manipulation de chaînes de caractères

- Ces instructions sont normalement des opérations de transfert de données. Il suffit de pointer sur le caractère de la chaîne source (par le registre ESI) à copier, de pointer sur le caractère de la chaîne destination (par le registre EDI) d'indiquer la direction de déplacement des pointeurs suite à l'exécution de l'instruction (par le bit D du registre EFLAGS, **CLD** pour mettre D à 0 et incrémenter ESI et EDI et **STD** pour mettre D à 1 et décrémenter ESI et EDI) puis d'exécuter l'instruction.



Le préfixe REP

- Le préfixe **rep** permet de répéter l'instruction qui le suit en décrémentant ECX jusqu'à ECX=0. Donc il exécute l'instruction x fois si l'on commence avec ECX=x.
- Les préfixes **repb** (identique à **repz**) et **repne** (identique à **repnz**) effectuent une exécution conditionnelle toute en répétant si la condition est vérifiée et si ECX est >0. Donc pour repne on sort de l'exécution si ECX=0 ou si l'instruction exécutée met l'indicateur Z à 1.

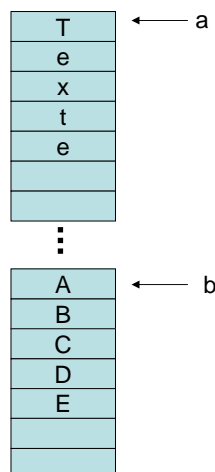
Les instructions de manipulation des chaînes de caractères

- CMPS: pour comparer [ESI] à [EDI]. $ESI += (-1)^D$ et $EDI += (-1)^D$ (ou $2 * (-1)^D$ ou $4 * (-1)^D$)
 - CMPSB
 - CMPSW
 - CMPSD
- LODS: pour charger AL, AX ou EAX de [ESI]. $ESI += (-1)^D$ (ou $2 * (-1)^D$ ou $4 * (-1)^D$)
 - LODSB
 - LODSW
 - LODSD
- MOVS: pour copier [ESI] dans EDI. $ESI += (-1)^D$ et $EDI += (-1)^D$ (ou $2 * (-1)^D$ ou $4 * (-1)^D$)
 - MOVSB/MOVSW/MOVSDB
- SCAS: pour comparer [EDI] à AL, AX ou EAX. $EDI += (-1)^D$ (ou $2 * (-1)^D$ ou $4 * (-1)^D$)
 - SCASB / SCASW / SCASD
- STOS: pour stocker AL, AX ou EAX dans [EDI]. $EDI += (-1)^D$ (ou $2 * (-1)^D$ ou $4 * (-1)^D$)
 - STOSB / STOSW / STOSD

exemple

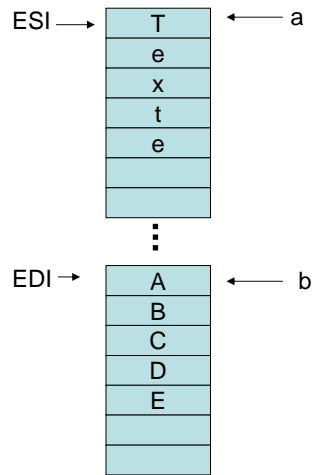
```

mov esi,a;
mov edi,b;
mov ecx,5;
cld;
rep movsb;
  
```



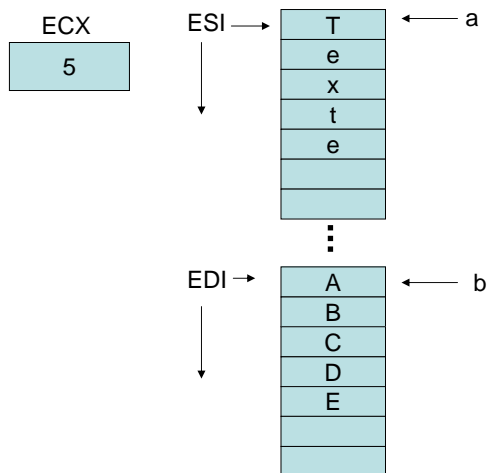
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



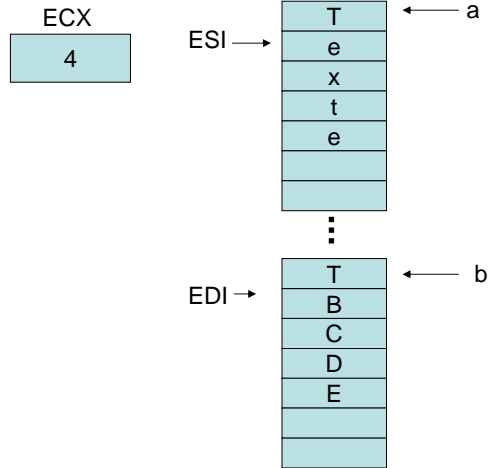
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



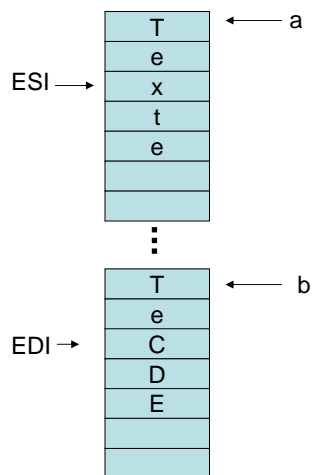
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



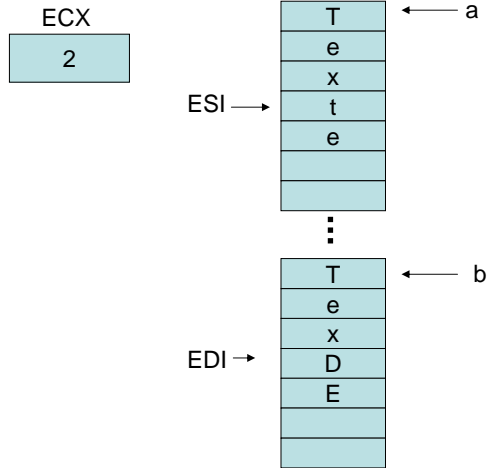
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



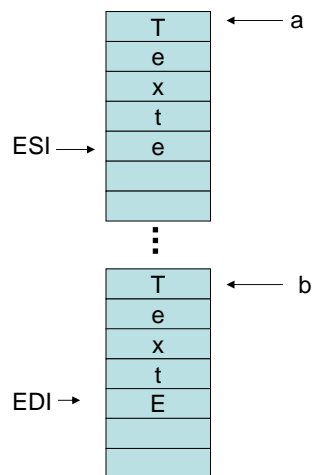
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



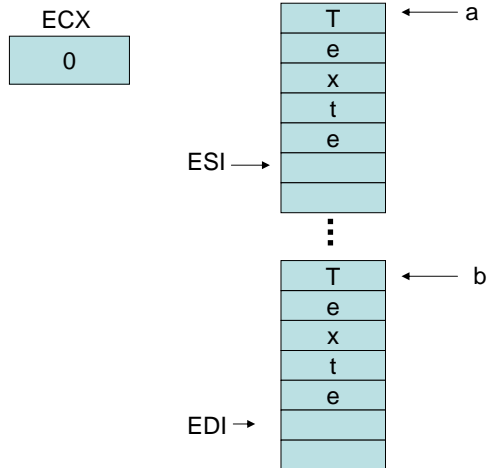
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



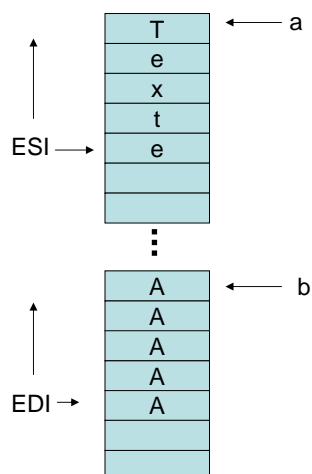
exemple

```
mov esi,a;  
mov edi,b;  
mov ecx,5;  
cld;  
rep movsb;
```



Exemple avec STD

```
mov esi,a+5;  
mov edi,b+5;  
mov ecx,5;  
std;  
rep movsb;
```



Exemple avec STD

```
mov esi,a+5;  
mov edi,b+5;  
mov ecx,5;  
std;  
rep movsb;
```

