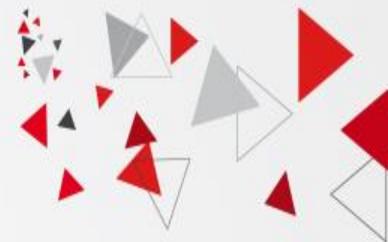
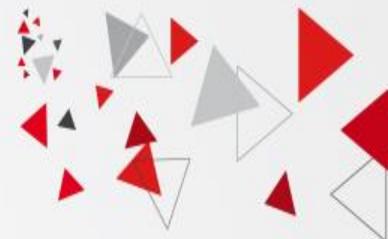




Fonctions de manipulation de chaînes



- Il existe de nombreuses fonctions permettant de manipuler les chaînes de caractères.
- Les fonctions de manipulation des chaînes se trouvent dans la bibliothèque **<string.h>**.
- Exemples de fonctions de la bibliothèque **<string.h>**
 - `strlen`
 - `strcpy`
 - `strncpy`
 - `strcat`
 - `strncat`
 - `strcmp`

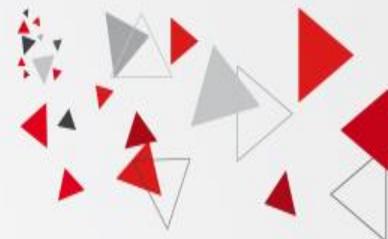


► Fonctions de manipulation de chaînes

- **strlen**: une fonction de la bibliothèque <string.h> qui retourne la taille d'une chaîne sans compter le caractère '\0'.

Exemple

```
int lg;  
Lg=strlen("TOTO");           /*lg=4*/
```



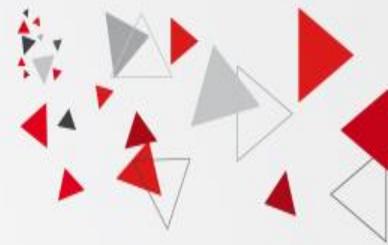
Fonctions de manipulation de chaînes

- **strcpy** : permet de recopier une chaîne de caractères dans une variable de type chaîne de caractères.

```
strcpy ( chaine1, chaine2); /* copie chaine2 dans chaine1 */
```

Exemple

```
char chaine1[5]= "";           //chaine vide
strcpy (chaine1, "Toto");    /*chaine1 contient la chaine Toto*/
```



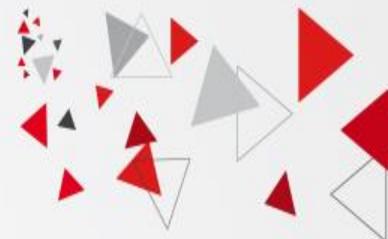
► Fonctions de manipulation de chaînes

- **strncpy** : permet de recopier « n » caractères de chaine2 dans chaine1.

**strncpy (chaine1, chaine2, n); /* copie n caractères de
chaine2 dans chaine1 */**

Exemple

```
char chaine1[15]= "";           //chaine vide
strcpy (chaine1, "Programmation", 4);
/* chaine1 contient la chaine : "Prog " */
```



Fonctions de manipulation de chaînes

- **strcat** : permet de concaténer deux chaînes de caractères.

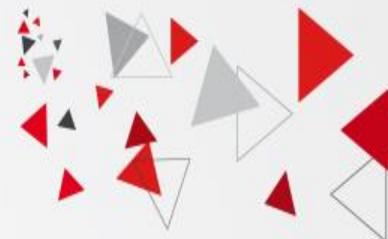
strcat (chaine1,chaine2); /* ajoute chaine2 à la fin de chaine1 */

Exemple

```
strcpy (chaine1, "Bon");
```

```
strcpy (chaine2, "jour")
```

```
strcat (chaine1, chaine2); /* (chaine1 contient Bonjour)*/
```

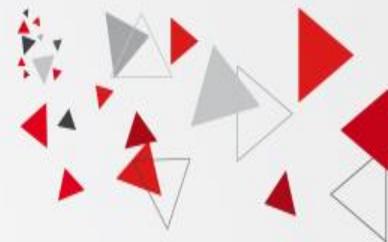


► Fonctions de manipulation de chaînes

- **strncat** : elle permet de concaténer seulement les « n » premiers caractères de la chaîne de caractères «chaine2» à la fin de la chaîne de caractères «chaine1».

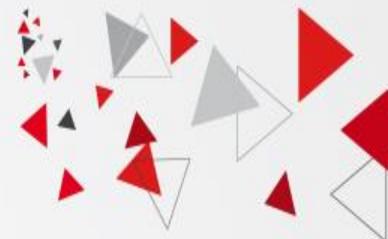
Exemple

```
strcpy (chaine1, "Cours de ") ;  
strcpy (chaine2, "Programmation") ;  
strncat ( chaine1, chaine2, 4); /* (chaine1 contient  
                                la chaîne : "Cours de Prog") */
```



Fonctions de manipulation de chaînes

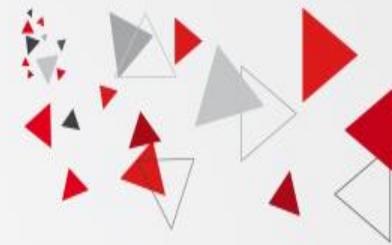
- **strcmp** : permet de comparer deux chaînes de caractères selon l'ordre lexicographique.
- strcmp (chaine1,chaine2);
compare chaine1 et chaine2 et rend la valeur :
 - 0 : si elles sont identiques.
 - une valeur négative : si chaine1 précède chaine2.
 - une valeur positive : si chaine1 suit chaine2.



► Fonctions de manipulation de chaînes

Exemple

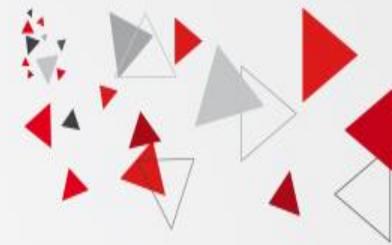
```
strcpy (chaine1, "Bon");
strcpy (chaine2, "Ben");
if (!strcmp (chaine1, chaine2)) // if ( strcmp (chaine1, chaine 2) ==
0)
    printf ("Les 2 chaînes sont égales");
else
    printf ("Les 2 chaînes sont différentes");
```



▶ Fonctions de manipulation de chaînes

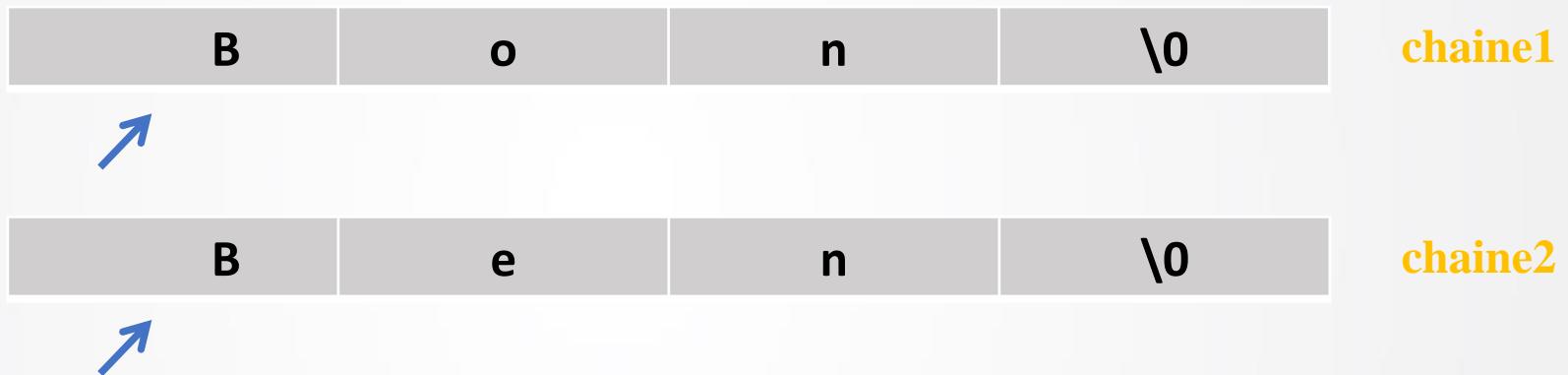
Exemple

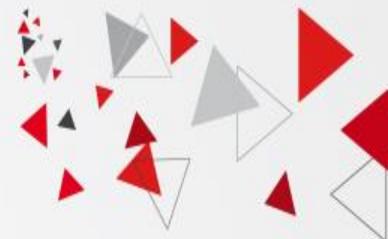
B	o	n	\0	chaine1
B	e	n	\0	chaine2



Fonctions de manipulation de chaînes

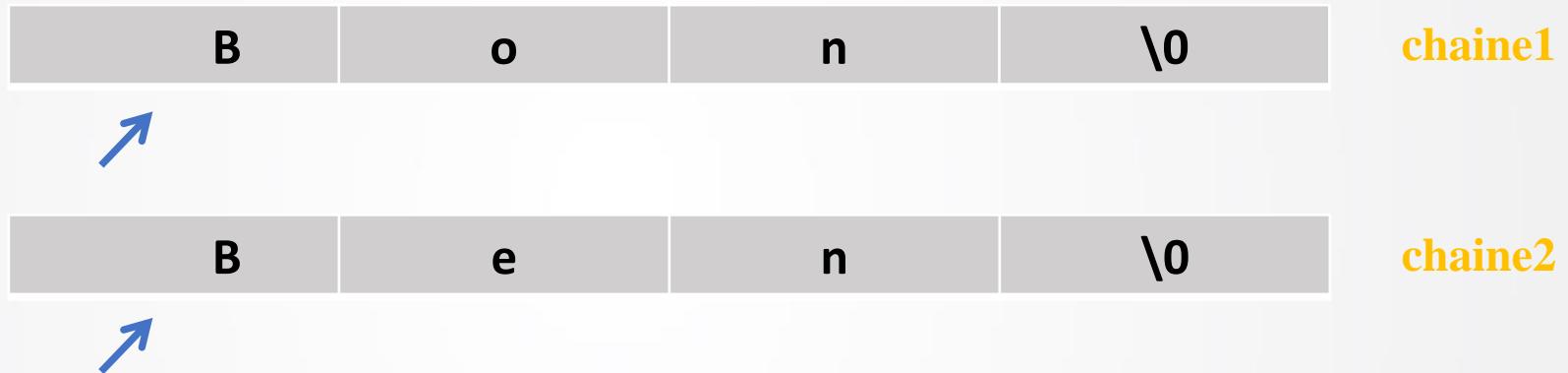
Exemple



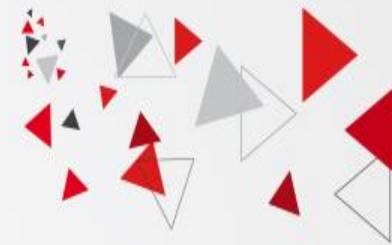


Fonctions de manipulation de chaînes

Exemple

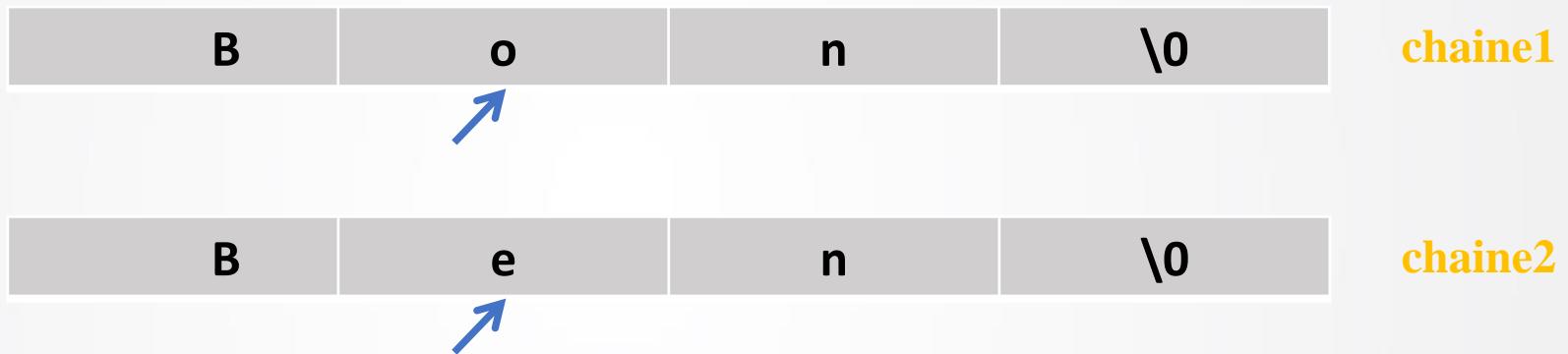


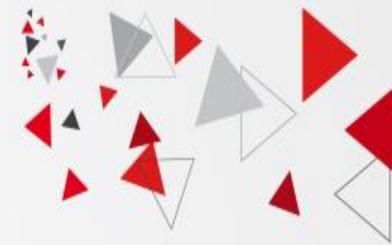
Le caractère 'B' de chaine1 == caractère 'B' de chaine2



Fonctions de manipulation de chaînes

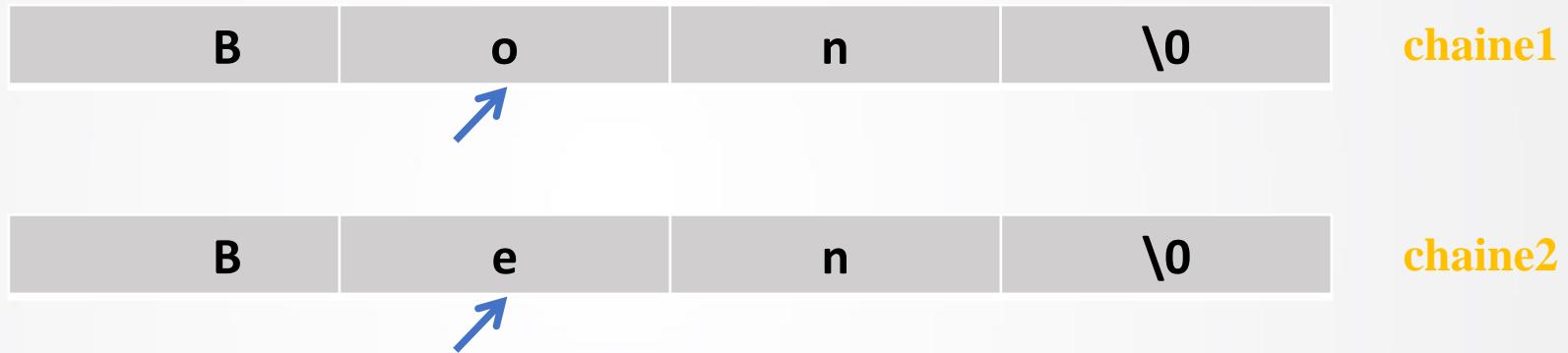
Exemple



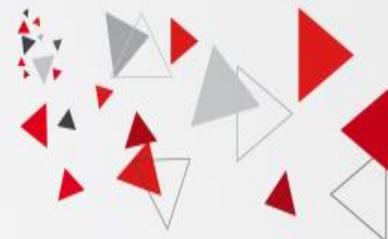


Fonctions de manipulation de chaînes

Exemple

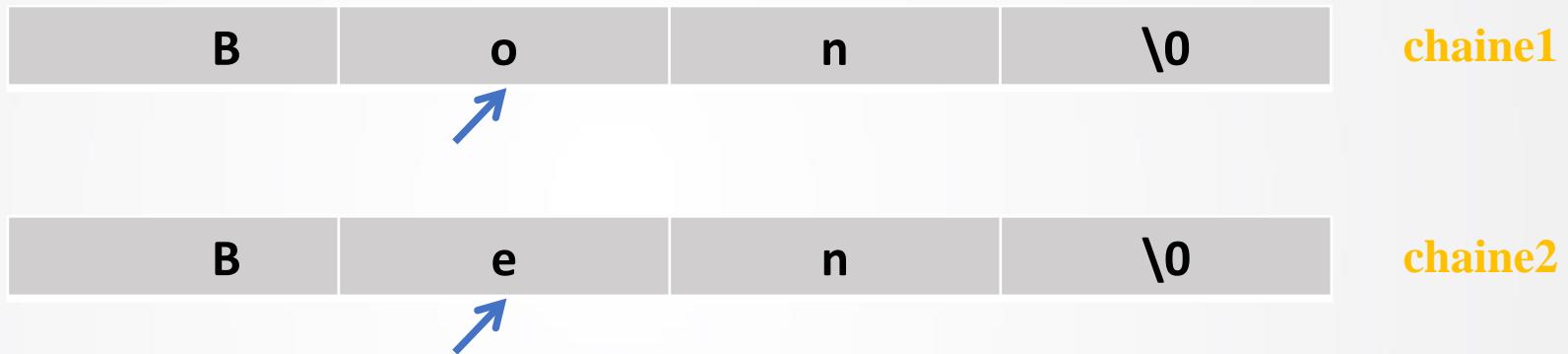


Le caractère 'e' de chaine2 < (précède lexicographiquement) le caractère 'o' de chaine1



Fonctions de manipulation de chaînes

Exemple



Le caractère 'e' de chaine2 < (précède lexicographiquement) le caractère 'o' de chaine1

--> Le résultat de la comparaison avec la fonction : strcmp (chaine1, chaine2)

- 1). Affichage du message :« Les 2 chaînes sont différentes»
- 2). une valeur > 0