

# *Chapitre 9*

## **Les chaînes de caractères**

# *I. Introduction*

- Il n'existe pas de type spécial *chaîne* ou *string* en C.
- Une chaîne de caractères est traitée comme un *tableau à une dimension de caractères* (vecteur de caractères).
- Il existe des notations particulières et beaucoup de fonctions spéciales pour le traitement de tableaux de caractères.

## *II. Déclaration*

- `char <NomVariable> [<Longueur>];`
- *Exemples :*
  - `char NOM [20];`
  - `char PRENOM [20];`
  - `char ADRESSE [300];`
- La représentation interne d'une chaîne de caractères est terminée par le symbole `'\0'`.
- Pour une chaîne de `n` caractères, il faut prévoir `n+1` octets.

## *II. Déclaration*

- Le nom d'une chaîne est le représentant de *l'adresse du premier caractère* de la chaîne.
- Pour mémoriser une variable qui doit être capable de contenir un texte de N caractères, nous avons besoin de N+1 octets en mémoire:
- *Exemple:*  
`char TXT[10] = "BONJOUR !";`

## *III. Utilisation*

- L'accès à un élément d'une chaîne de caractères peut se faire de la même façon que l'accès à un élément d'un tableau.
- En déclarant une chaîne par :  
**char A[6];**  
nous avons défini un tableau A avec six éléments
- On peut y accéder par : **A[0], A[1], ... , A[5]**

## *III. Utilisation*

- L'ordre sur les chaînes est l'ordre lexicographique.

"ABC" précède "B"

"Abc" précède "abc"

"ab" précède "abcd"

" ab" précède "ab"

(le code ASCII de ' ' est 32, et le code ASCII de 'a' est 97)

## *III. Utilisation*

### ■ *Exemples :*

```
if (C>='0' && C<='9') printf("Chiffre\n");  
if (C>='A' && C<='Z') printf("Majuscule\n");  
if (C>='a' && C<='z') printf("Minuscule\n");
```

Pour convertir des lettres majuscules en des lettres minuscules:

```
if (C>='A' && C<='Z') C = C-'A'+'a';
```

# *IV. Les fonctions prédéfinies*

## *A. La bibliothèque stdio.h*

- **printf** avec le spécificateur de format **%s** permet d'intégrer une chaîne de caractères dans une phrase.

*Exemple:*

```
char NOM[] = "hello, world";  
printf(":%s:", NOM);
```

:hello, world:

# *IV. Les fonctions prédéfinies*

## *A. La bibliothèque `stdio.h`*

- `puts` permet d'écrire une chaîne constante ou le contenu d'une variable dans une ligne isolée.

Syntaxe: `puts( <Chaîne> )`

- `puts` écrit la chaîne de caractères désignée par `<Chaîne>` sur *stdout* et provoque un retour à la ligne.
- En pratique, `puts(TXT);` est équivalent à `printf("%s\n",TXT);`

# *IV. Les fonctions prédéfinies*

## *A. La bibliothèque stdio.h*

- **scanf** avec le spécificateur **%s** lit un *mot* du fichier d'entrée standard *stdin* et le mémorise à l'adresse qui est associée à **%s**.
- *Exemple :*

```
char LIEU[25];  
int JOUR, MOIS, ANNEE;  
printf("Entrez lieu et date de naissance : \n");  
scanf("%s %d %d %d", LIEU, &JOUR, &MOIS,  
&ANNEE);
```



# *IV. Les fonctions prédéfinies*

## *A. La bibliothèque stdio.h*

- **gets** permet de lire une ou plusieurs lignes de texte (p.ex. des phrases) terminées par un retour à la ligne. Syntaxe: **gets( <Chaîne> )**
- Effet: **gets** lit une chaîne de de caractères de *stdin* et la copie à l'adresse indiquée par **<Chaîne>**.
- Le retour à la ligne final est remplacé par le symbole de fin de chaîne **'\0'**.
- *Exemple:*  

```
int MAXI = 1000;  
char LIGNE[MAXI];  
gets(LIGNE);
```

# *IV. Les fonctions prédéfinies*

## *A. La bibliothèque stdio.h*



- **gets et scanf(“%s”,... sont des fonctions dangereuses car vous risquez le buffer overflow si l'utilisateur entre une chaîne plus longue que prévue.**
- **Risque de piratage selon les types de programmes...**

# *IV. Les fonctions prédéfinies*

## *B. La bibliothèque strings.h*

- **strlen(<s>)** fournit la longueur de la chaîne *sans* compter le '\0' final
- **strcpy(<s>, <t>)** copie <t> vers <s>
- **strcat(<s>, <t>)** ajoute <t> à la fin de <s>
- **strcmp(<s>, <t>)** compare <s> et <t> lexicographiquement et fournit un résultat:
  - Négatif si <s> précède <t>
  - Zéro si <s> est égal à <t>
  - Positif si <s> suit <t>
- **strncpy(<s>, <t>, <n>)** copie au plus <n> caractères de <t> vers <s>
- **strncat(<s>, <t>, <n>)** ajoute au plus <n> caractères de <t> à la fin de <s>



# *IV. Les fonctions prédéfinies*

## *B. La bibliothèque strings.h*

- Comme le nom d'une chaîne de caractères représente une adresse fixe en mémoire, on ne peut pas 'affecter' une autre chaîne au nom d'un tableau.
- Il faut copier la chaîne caractère par caractère ou utiliser la fonction **strcpy** respectivement **strncpy**:
- **strcpy(A, "Hello");**

# *IV. Les fonctions prédéfinies*

## *C. La bibliothèque stdlib.h*

- **atoi(<s>)** retourne la valeur numérique représentée par <s> comme int
- **atol(<s>)** retourne la valeur numérique représentée par <s> comme long
- **atof(<s>)** retourne la valeur numérique représentée par <s> comme double
- **Pour une chaîne non convertible, ces fonctions retournent zéro**

# *IV. Les fonctions prédéfinies*

## *D. La bibliothèque ctype.h*

- La fonction ... retourne 1 ...
  - ... isupper(<c>) ... si <c> est une majuscule ('A'...'Z')
  - ... islower(<c>) ... si <c> est une minuscule ('a'...'z')
  - ... isdigit(<c>) ... si <c> est un chiffre décimal ('0'...'9')
  - ... isalpha(<c>) ... si islower(<c>) ou isupper(<c>)
  - ... isalnum(<c>) ... si isalpha(<c>) ou isdigit(<c>)
  - ... isxdigit(<c>) ... si <c> est un chiffre hexadécimal
  - ... isspace(<c>) ... si <c> est un signe d'espacement
  
- tolower(<c>) retourne <c> converti en minuscule  
toupper(<c>) retourne <c> converti en majuscule

