



Initialisation (1/5)

Pour initialiser une chaîne de caractères, il suffit de remplir les cases du tableau avec des caractères, sachant que celui-ci devra obligatoirement contenir le caractère de fin de chaîne ‘\0’.



Initialisation (2/5)

1- On peut remplir la chaîne caractère par caractère:

```
Char Chaine[3+1];
```

```
Chaine[0]='B'
```

```
Chaine[1]='O'
```

```
Chaine[2]='N'
```

```
Chaine[3]='\0'
```



Initialisation (3/5)

2- Les chaînes de caractères peuvent aussi être initialisées par l'indication de la liste des éléments du tableau entre accolades:

```
char Chaine[]={ 'H', 'E', 'l', 'l', 'o', '\0' };
```

--> Réserve de 6 cases en mémoire

Equivalent d'écrire :

```
char Chaine [6] = { 'H', 'E', 'l', 'l', 'o', '\0' };
```

Initialisation (4/5)



- Il est possible aussi d'initialiser une chaîne de caractères directement par une chaîne constante.

```
char Chaine1[] = " HELLO ";  
char Chaine2[] = "";           //chaîne vide
```

- Lors de l'initialisation, le compilateur réserve automatiquement le nombre d'octets nécessaires pour la chaîne, c.à.d.: le nombre de caractère +1.
- C'est le compilateur qui rajoutera le caractère '\0' à la fin de la chaîne.

--> On réserve 6 octets pour Chaine1 et un octet pour Chaine2. 4



Initialisation (5/5)

```
char Chaine1[6]= "HELLO ";    // OK
```

Par contre:

```
char Chaine1[5]= "HELLO "; // donnera une erreur d'exécution
```

```
char Chaine1[4]= "HELLO "; // donnera une erreur d'exécution.
```