

# Mise à niveau en langage C

Infres 1A

Enseignant: P. BERTIN-JOHANNET

# Les tableaux

- Un tableau est une liste d'éléments du même type
- Pour déclarer un tableau on précise
  - Le type des éléments qu'il contient
  - Le nom de la variable
  - Le nombre d'éléments

```
type nom_tableau[nombre_elements];
```

- On accède ensuite à l'élément `n` d'un tableau ainsi :  
`nom_tableau[n - 1];`

# Exemple :

Programme:

```
char tab [4];
```

```
tab[0] = 1;
```

```
tab[1] = 2;
```

```
tab[2] = tab[0] + 2;
```

Mémoire :



# Exemple :

Programme:

```
> char tab [4];  
tab[0] = 1;  
tab[1] = 2;  
tab[2] = tab[0] + 2;
```

Mémoire :



# Exemple :

Programme:

```
char tab [4];
```

```
> tab[0] = 1;
```

```
tab[1] = 2;
```

```
tab[2] = tab[0] + 2;
```

Mémoire :



# Exemple :

Programme:

```
char tab [4];
```

```
tab[0] = 1;
```

```
> tab[1] = 2;
```

```
tab[2] = tab[0] + 2;
```

Mémoire :



# Exemple :

Programme:

```
char tab [4];
```

```
tab[0] = 1;
```

```
tab[1] = 2;
```

```
> tab[2] = tab[0] + 2;
```

Mémoire :

1
2
3
tab[3]
Octet 5
Octet 6
Octet 7

# Les chaines de caractères

- En C, une chaine de caractère est un tableau de `char` se terminant par la valeur zero
- **/!\ Attention** il ne s'agit pas du caractère `'0'` (qui vaut en ASCII `48`) mais de `'\0'` (qui est en ASCII `0`).
- Il faut donc toujours réserver une place de plus pour une chaines de caractères.
- Le langage C permet d'écrire une chaine de caractères entre guillemets : `"Bonjour"`



# Exemple :

Programme:

```
char tab [4] = "oui";  
tab[0]++;
```

Mémoire :



# Exemple :

Programme:

```
> char tab [4] = "oui";  
tab[0]++;
```

Mémoire :

111 (o)
117 (u)
105 (i)
0
Octet 5
Octet 6
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
> tab[0]++;
```

Mémoire :

112 (p)
117 (u)
105 (i)
0
Octet 5
Octet 6
Octet 7

# Les chaines de caractères

- On peut utiliser `printf` et `scanf` avec les chaines de caractères
- Le format à utiliser est `%s`
- **/!\ Attention** pour scanner une chaine de caractères avec `scanf`, il ne faut pas utiliser de `&`

# Exemple :

Programme:

```
char tab [4];  
scanf("%s", tab);  
tab[0] -= 32;  
printf("%s\n", tab);
```

# Exemple :

Programme:

```
char tab [4];  
scanf("%s", tab);  
tab[0] -= 32;  
printf("%s\n", tab);
```

Entrée utilisateur :

oui

# Exemple :

Programme:

```
char tab [4];  
scanf("%s", tab);  
tab[0] -= 32;  
printf("%s\n", tab);
```

Entrée utilisateur :

oui

Affiché :

Oui

# La boucle while

- L'instruction while (en anglais : "tant que") permet d'exécuter du code tant qu'une condition est vraie.

```
while(condition){  
    code  
}
```

- Les lignes de `code` entre les accolades seront répétées tant que la `condition` est vraie



# Exemple :

Programme:

```
char tab [4] = "oui";  
int i = 0;  
while(tab[i] != 0){  
    tab[i]++;  
    i++;  
}
```

Mémoire :



# Exemple :

Programme:

```
> char tab [4] = "oui";  
int i = 0;  
while(tab[i] != 0){  
    tab[i]++;  
    i++;  
}
```

Mémoire :

111 (o)
117 (u)
105 (i)
0
Octet 5
Octet 6
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
> int i = 0;
```

```
while(tab[i] != 0){
```

```
    tab[i]++;
```

```
    i++;
```

```
}
```

Mémoire :

111 (o)
117 (u)
105 (i)
0
0
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
> while(tab[i] != 0){  
    tab[i]++;  
    i++;  
}
```

Mémoire :

111 (o)
117 (u)
105 (i)
0
0
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
while(tab[i] != 0){
```

```
>   tab[i]++;
```

```
   i++;
```

```
}
```

Mémoire :

112 (p)
117 (u)
105 (i)
0
0
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
while(tab[i] != 0){
```

```
    tab[i]++;
```

```
    i++;
```

```
}
```

>

Mémoire :

112 (p)
117 (u)
105 (i)
0
1
Octet 7

# Exemple :

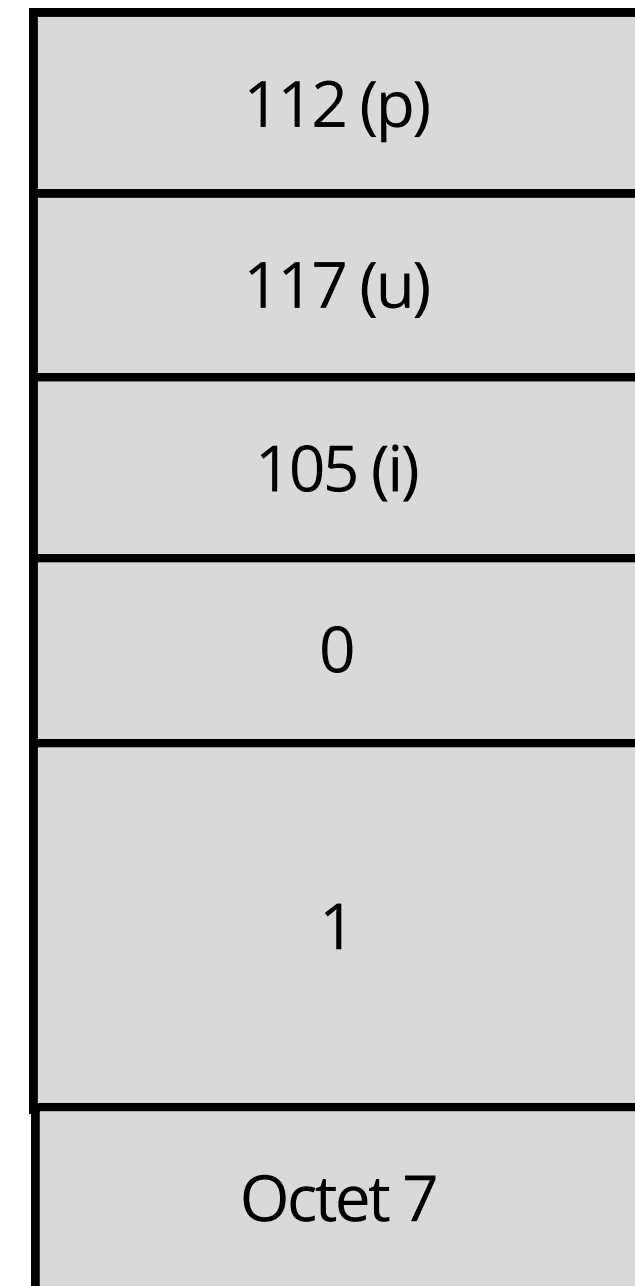
Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
> while(tab[i] != 0){  
    tab[i]++;  
    i++;  
}
```

Mémoire :



# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
while(tab[i] != 0){
```

```
>   tab[i]++;
```

```
   i++;
```

```
}
```

Mémoire :

112 (p)
118 (v)
105 (i)
0
1
Octet 7



# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
while(tab[i] != 0){
```

```
    tab[i]++;
```

```
    i++;
```

```
}
```

Mémoire :

112 (p)
118 (v)
105 (i)
0
2
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
> while(tab[i] != 0){  
    tab[i]++;  
    i++;  
}
```

Mémoire :

112 (p)
118 (v)
105 (i)
0
2
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
while(tab[i] != 0){
```

```
>   tab[i]++;
```

```
   i++;
```

```
}
```

Mémoire :

112 (p)
118 (v)
106 (j)
0
2
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
while(tab[i] != 0){
```

```
    tab[i]++;
```

```
    i++;
```

```
}
```

>

Mémoire :

112 (p)
118 (v)
106 (j)
0
3
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

```
> while(tab[i] != 0){  
    tab[i]++;  
    i++;  
}
```

Mémoire :

112 (p)
118 (v)
105 (j)
0
3
Octet 7

# Exemple :

Programme:

```
char tab [4] = "oui";
```

```
int i = 0;
```

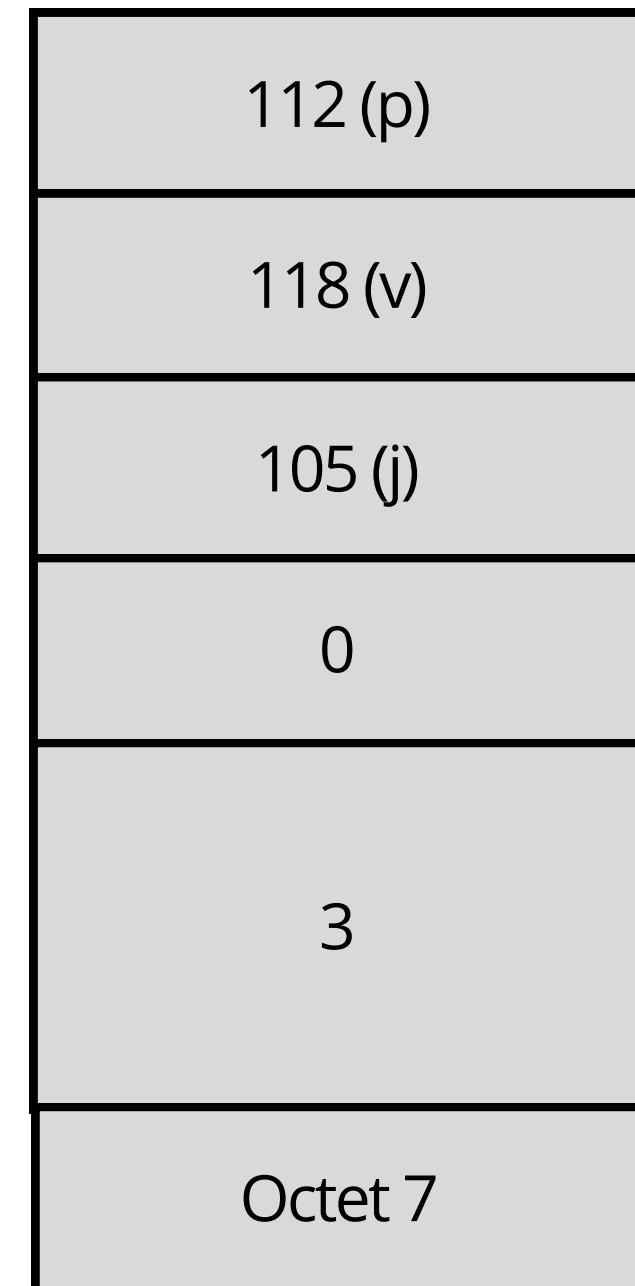
```
while(tab[i] != 0){
```

```
    tab[i]++;
```

```
    i++;
```

```
> }
```

Mémoire :



# La boucle for

- L'instruction for (en anglais : "pour") s'écrit ainsi :  

```
for(depart; condition; instruction){  
    code  
}
```
- L'instruction `depart` sera exécutée une fois au début
- Tant que `condition` sera vraie, `code` et `instruction` seront exécutées en boucle.

# Exemple :

On aurait pu écrire le programme précédent ainsi :

```
char tab [4] = "oui";  
for(int i = 0; tab[i] != 0; i++){  
    tab[i]++;  
}
```



# Exemple :

Ou ainsi :

```
char tab [4] = "oui";  
for(int i = 0; tab[i] != 0; tab[i++]++);
```

# Durée de validité d'une variable

- Une variable déclarée entre deux accolades existe uniquement entre ces deux accolades
- Une variable déclarée entre les parenthèses d'une boucle for, n'existe que pour cette boucle for

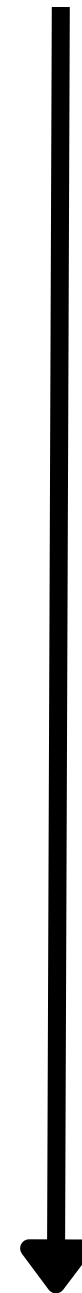
# Example :

```
int a = 0;  
for (int b = 0; a < 4; b++){  
    if (a == b){  
        int i = 5;  
        a = i;  
    }  
}  
a ++;
```

A

B

I



# Buffer overflow

- **/!\ Attention** : Il est possible d'écrire en dehors des limites d'un tableau, cela peut causer des failles de sécurité appelées buffer overflow
- Pour éviter ce problème, pensez à toujours contrôler la taille des chaînes de caractères que vous enregistrez

# Mise en pratique

Si vous n'avez pas d'environnement linux vous pouvez utiliser :

[https://www.onlinegdb.com/online\\_c\\_compiler](https://www.onlinegdb.com/online_c_compiler)