

## TD 2 - Syntaxe et Expressions

### 1. Utilisation des diagrammes de Conway

Quels sont les littéraux corrects dans la série:

432	7.4	3,475	.1475	'\020'
6E3	"123"	0.0001	'\458'	0671
1.0e-4	0xffff	0.1E999	432.	0XA67F
'!'	'1'	-6.0	-12.3e-4	11.e4.5

Pour chaque cas favorable, préciser le type de cette constante.

### 1. Vérifier la validité de :

1.1 `int **tab ;` (déclaration variable)

1.2 `for ( ; ; )` (instructions)

1.3 `for(i=0; i<MAX; i++) ;` (instructions)

1.4 `struct {  
    char *nom ;  
    int age ;  
} *18 ;` (déclaration variable)

### 3. Calculer les expressions C suivantes :

<code>4 * 18 - 198 % 10</code>	<code>7 + 3 &lt;= 10</code>	<code>(8+4) &gt; 3 &amp;&amp; 3 &gt; 2</code>
<code>(36 / 18 * 4) / 3</code>	<code>7 &lt; 3 &lt; 1</code>	<code>8.1 == 9.8</code>
<code>10 / 4</code>	<code>10.0 / 4.0</code>	<code>'c' &lt; 'f'</code>
<code>16 &lt; x &lt; 24</code>		

### 4. Que vaut x après la suite d'instructions :

```
y = 3 ;
x = 2 ;
y++ ;
x += 2 ;
x *= 2 + y;
x -= 2 ;
```

#### Complément de cours

#### Opérateurs d'affectations (OPA):

en algo                    `variable ← expression`  
en C                        **`variable = expression ;`**

exemple                    `i=3 ;`  
                              `j= i * 4 ;`  
                              `z = z+1 ;`

autres opérateurs d'affectation

**`variable ++`** ; équivalent à `variable = variable + 1 ;`  
**`variable --`** ; équivalent à `variable = variable - 1 ;`

**`variable operateur= expression ;`** ; équivalent à `variable = variable operateur (expression) ;`

exemple :                    `x++ ;`    équivalent à `x=x+1 ;`  
                              `x+=1 ;`    équivalent à `x=x+ (1) ;`