



## 3.4. Les expressions et les instructions

### Expressions

La formation des expressions est définie par récurrence :

Les *constantes* et les *variables* sont des expressions.

Les expressions peuvent être combinées entre elles par des *opérateurs* et former ainsi des expressions plus complexes.

Les expressions peuvent contenir des appels de fonctions et elles peuvent apparaître comme paramètres dans des appels de *fonctions*.

### Exemples

```
i =0
i++
X=pow(A,4)
printf(" Bonjour !\n")
a=(5*x+10*y)*2
(a+b)>=100
position!=limite
```

### Instructions

Une expression comme `I=0` ou `I++` ou `printf(...)` devient une **instruction**, si elle est suivie d'un point-virgule.

### Exemples

```
i=0;
i++;
X=pow(A,4);
printf(" Bonjour !\n");
a=(5*x+10*y)*2;
```

### Évaluation et résultats

En C toutes les expressions sont évaluées et retournent une valeur comme résultat:

`(3+2==5)` retourne la valeur 1 (vrai)  
`A=5+3` retourne la valeur 8

Comme les affectations sont aussi interprétées comme des expressions, il est possible de profiter de la valeur rendue par l'affectation:

```
( (A=sin(X)) == 0.5 )
```

