

## Séance 2

Opérateurs:

du plus prioritaire au moins prioritaire:

```

- (unaire)
*      /      %
+      -
<      <=     >      >=     ==
=      !=
!
&&    ||
    
```

Incrémentation ++, décrémentation --

affectation avec opération

+=, -=, \*=, /=

affectation conditionnelle

**x = (condition) ? valeurOui : valeurNon ;**

Structures de contrôle

**if    switch    for    do    while**

## Exercices

1) Donner le résultat de

$-4 - 3 * 2 - 1$

$4-5-3 == 2*-5/2+1$

$1+3*8\%3$

`int n = 20, x; x = n++;`

`int n = 20, x; x = ++n;`

`32 && 2.3`

`!65.34`

`0 || !(32>12)`

`int X, A = 5, B = 10, C = 1; X = ((2*A+3)*B+4)*C;`

`int X = 3, Y = 4; X *= Y + 1;`

2) Ecrire un programme qui lit une date au clavier sous la forme JJ-MM-AAAA. Un jour est compris entre 1 et 31, et un mois entre 1 et 12. Le programme doit être robuste: il redemande la valeur attendue tant qu'elle n'est pas correcte.

3) Ecrire un programme qui affiche le résultat d'une équation du second degré définie par ses trois coefficients  $a$ ,  $b$ ,  $c$  rentrés au clavier par l'utilisateur. NB: on utilisera la fonction `double sqrt(double)` qui retourne la racine carrée d'un nombre. Pour cela, on fera un `#include <math.h>` et on compilera avec l'option `-lm`.

4) Ecrire un programme qui devine un nombre entier secret compris entre 0 et 100 choisi par l'utilisateur. Pour guider le programme vers la solution, l'utilisateur répond `>` ou `<` ou `=`, selon que le nombre proposé par le programme est supérieur, inférieur, ou égal au nombre secret.