LANGAGE C AVANCÉ

TD 1 : Exercices de rappel

Syntaxe

Objectif : le présent TD a pour but de faire un rappel du langage C. Nous y aborderons la syntaxe, les conditions les boucles ainsi que les fonctions.

1. Quels sont les identificateurs acceptés ?

```
y
_X_
_
5
fct-1
_SOMME_POINTS_
4e_jour
plus_grand_diviseur
p.g.c.d.
```

2. Quel type peut avoir une variable pouvant contenir chacune des valeurs d'une ligne :

```
1  12   4  0   -125
1  12  -4  0    250
1  12   4  0    250
1  12  -4  0.5  125
-220    32000   0
-3000005.000000001
410    50000    2
410    50000    -2
3.14159265      1015
2*107           10000001
2*10-7          10000001
1.05*1050      0.0001
305.122212  0   -12
```

3. Dans le code suivant, `type(fiche[i,j]->name)` est évalué au plus 4 fois.

```
if (type(fiche[i,j]->name)==0)
  f_0(data);
 else if (type(fiche[i,j]->name)==1)
   f_1(data);
 else if (type(fiche[i,j]->name)==2)
   f_2(data);
 else if (type(fiche[i,j]->name)==5)
   f_5(data);
 else
   f_all(data);
```

Réécrire ce code, de façon à éviter ces évaluations multiples, et pour le rendre plus lisible.

4. Évaluer les expressions suivantes en supposant

```
a=20   b=5   c=-10   d=2   x=12   y=15
```

```
(5*(x+2)*3)*(b+4)
a == (b=5)
a += (x+5)
a != (c *= (-d))
a *= c+(x-d)
a %= d++
a %= ++d
(x++)*(a+c)
a = x*(b<c)+y*!(b<c)
!(x-d+c)||d
a&&b||!0&&c&&!d
x - (y = 3, y + 1)
```

5. Quels sont les blocs acceptés ?

```
{i = 5;}    { }    { ; ; }    { ;; }    { ; }
{i = 5; k = 3;}    {i = 5; k = 3}
{i = 5; int k; k = 3;}
```

6. Écrire un programme qui calcule la factorielle de 5, et l'affiche.

7. Écrire un programme qui lit trois valeurs entières (A, B et C) au clavier et qui affiche la plus grande des trois valeurs.

8. Écrire un programme qui calcule par multiplications successives x^n de deux entiers naturels x et n entrés au clavier.

9. Calculer la somme des n premiers termes de la série harmonique : $1 + 1/2 + 1/3 + ... + 1/n$.

10. Écrire une fonction qui prend trois arguments (`a`, `b`, `c`) de type `float` et qui renvoie le nombre de solutions de l'équation $\mathtt{a}\,x^2 + \mathtt{b}\,x + \mathtt{c} = 0$.

11. Écrire un programme qui affiche la somme des chiffres d'un nombre positif entré au clavier.

12. Écrire un programme qui affiche en hexadécimal un nombre saisi en décimal. (Et exercice inverse)

13. Écrire une fonction qui renvoie le plus petit commun multiple des deux nombres entiers (`int`) en argument (0, si l'un des deux arguments est 0).

14. Écrire une fonction qui affiche un triangle rempli d'étoiles, s'étendant sur un nombre de lignes (`int`) passé en argument. Par exemple `triangle(4)` a pour effet d'afficher :

```
*
**
***
****
```

15. Écrire une fonction `fibonacci` à un seul argument `n` qui renvoie le nombre de Fibonacci F_n correspondant à `n`. Pour rappel

$$\begin{aligned} F_0 &= 0 \\ F_1 &= 1 \\ F_n &= F_{n-1} + F_{n-2} \text{ si } n > 2 \end{aligned}$$

Écrire une version récursive, et une version itérative.

16. Écrire un prédicat `prime` qui détermine si un entier strictement positif est premier ou non.

17. Écrire une fonction `nbsum` qui prend comme argument un nombre n et qui renvoie le nombre de façons d'écrire une somme égale à n (on comptera une seule fois les commutations).

Par exemple, il existe 5 façon d'écrire une somme égale à 5 :

$$\begin{aligned} 5 &= 5 \\ &= 4+1 \\ &= 3+2 \\ &= 3+1+1 \\ &= 2+2+1 \\ &= 2+1+1+1 \\ &= 1+1+1+1+1 \end{aligned}$$