

Langage C

TD1

Itheri Yahiaoui && Stéphane Cormier

Exercice 1 : « Les commentaires et la structure du code »

Observez le code C suivant et indiquez tous les problèmes ne lui permettant pas d'être compilé correctement.

```
#include <stdlib.h>
int g ;

int main
# define y 30
{ /* Déclarations de variables
  int a = 125,
  short s = 4043;      /*
  float x = 2.13459
  double dx ;          /***** dx est un réel à double précision *****/
  char c = 'W';
  char ch = "Hello" ;
  define z 21 ;

  *****/ affichage de la somme *****/
  printf("a + c = %d\n", a + c); /*affichage d'un entier «%d»*/
  printf("x + c = %f\n", x + c) // affichage d'un float «%f»
  dx = 1.1415927 ;
  printf("dx + x = %f\n", dx + x);

  #include <math.h>
  /******/ Exemple de transtypage /******/
  /* affichage d'un long « %ld » */
  printf("((int) dx) + ax = %ld\n", ((int) dx) + ax);
  int b = 12345;
  printf("a + x = %f\n", a + x);
  printf("s + b = %d\n", s + b)
  long ax = 1234567890;
  printf("ax + b = %ld\n", ax + b);
  printf("s + c = %hd\n", s + c)
  printf("ax + c = %ld\n", ax + c);
  ;
  ;
  unsigned long ux = 2541567890;
  // affichage d'un long non signé « %lu »
  printf("ax + ux = %lu\n", ax + ux);
  printf("\n")
  return 0;
}
```

Exercice 2 : « Les noms de variables »

Est-ce que les noms de variables suivants sont corrects ? Sinon expliquez d'où vient l'erreur.

- 1_reel
- un-reel
- un_réel
- un_reel
- int
- ab_35
- A65
- x'
- 7on
- nom_de_variable
- cOmPteurR
- .var
- case
- CE\$

Exercice 3 : « Initialisation de variables »

Complétez le tableau suivant.

```
int main()
{
    long x, y , compt, lim;
    double X, dep, var;
    char symbole, lettre;
    .....
    .....
    .....
    .....
    return 0 ;
}
```

	x	y	compt	lim	X	dep	var	symbole	lettre
x = -10 ;									
symbole = 'x'+x ;									
y= 3 - x ;									
compt = 0 ;									
lim= y + y									
y = y - 1 ;									
X = 6.022E+23									
var = X * 1.0E-17 ;									
dep = var * (y - x) ;									
compt = compt % lim ;									

Exercice 4: « Les types de variables »

Complétez le tableau suivant.

```
int main()
{
    long v1, v2;
    double y1, y2, w1;
    .....
    .....
    .....
    .....
    return 0 ;
}
```

	v1	v2	y1	y2	w1
v1 = 11 ;					
y1 = 4.0E+2 ;					
v2 = v1 -6 ;					
w1 = 3 * y1 - 200.0 ;					
y2 = v1 * v2 ;					
y1 = v1 / v2 ;					
w1 = y2 - v1					
y2 = v2 / v1					
v2 = y1 ;					

Exercice 5 : « Affichage simple »

- Ecrire un programme qui affiche le message 'Hello World !'
- Ecrire un programme qui affiche votre nom, date de naissance et numéro de portable
- Ecrire un programme qui affiche la forme de la lettre F en utilisant le symbole (#).

```
#####
#
#
#####
#
#
#
```

Exercice 6: « Cas simples d'utilisation de variables »

- Ecrire un programme qui calcule le produit de deux entiers initialisés par vous même.
- Ecrire un programme qui calcule la somme de trois nombres entiers initialisés par vous même.
- Ecrire un programme pour calculer le périmètre et l'aire, en mètre et mètre carré, d'un rectangle d'une hauteur de 70 cm et d'une largeur de 50 cm.
- Ecrire un programme qui fait l'échange du contenu de deux variables entre elles.
- Ecrire un programme proposant une technique de rotation des valeurs sur une série de 6 variables.

- Nous avons un échantillon composé de N fois la valeur réelle 2.34 et M fois la valeur réelle 4.52. Ecrivez un programme qui initialise les valeurs de N et M et calcule la valeur moyenne de l'échantillon.
- Ecrire un programme qui calcule le maximum de trois entiers sans utiliser la structure « if...else... »

Exercice 6 : « Priorité des opérateurs »

Soit le programme C suivant, donnez la trace d'exécution de ce code en expliquant ce qui se passe ligne par ligne.

```
int main()
{
    /*** Déclaration de variables ***/

    int x, y, z;

    /***/

    x = 2;
    x += 3 + 2;
    printf(" x = %d \n", x);
    x -= y = z = 4;
    printf(" x = %d y = %d z = %d \n", x, y, z);
    x = y == z;
    printf(" x = %d y = %d z = %d \n", x, y, z);
    x == (y = z);
    printf(" x = %d y = %d z = %d \n", x, y, z);
    x = 3; y = 2; z = 1;
    x = x && y || z;
    printf(" x = %d \n", x);
    printf(" L'évaluation de l'expression donne %d \n", x || ! y && z);
    x = y = 0;
    z = x ++ -1;
    printf(" x = %d, z = %d \n", x, z);
    z += -x ++ + ++ y;
    printf(" x = %d, z = %d \n", x, z);

    x = 1; y = 1;
    printf(" L'évaluation de l'expression donne %d \n", ! x | x);
    printf(" L'évaluation de l'expression donne %d \n", ~ x | x);
    printf(" L'évaluation de l'expression donne %d \n", x ^ x);
    x <= 3;
    printf(" x = %d \n", x);
    y <= 3;
    printf(" y = %d \n", y);
    y >= 3;
    printf(" y = %d \n", y);
    return 0;
}
```