

**Exercice 1 : Identificateurs**

Pour chacun des identificateurs suivants, indiquez s'il est correct ou incorrect, et donnez une justification.

Identificateur	Validité	Justification / Commentaire
2_pi	.....	.....
x_2	.....	.....
x__3	.....	.....
x 2	.....	.....
robotPosition	.....	.....
stored_values	.....	.....
-output_current	.....	.....
_alarm_	.....	.....
issue#2	.....	.....
int	.....	.....
défaillance	.....	.....
f'	.....	.....
INT	.....	.....
main	.....	.....
M_PI	.....	.....

**Exercice 2 : Affectation**

Pour chacune des variables suivantes, indiquer leurs valeurs. Les variables non initialisées sont notées avec ?. Cela signifie que leur contenu n'est pas prévisible.

Instruction	a	b	c	d
a = 5;	.....	.....	.....	.....
b = c;	.....	.....	.....	.....
c = a;	.....	.....	.....	.....
a = a + 1;	.....	.....	.....	.....
b = a * (c + 1);	.....	.....	.....	.....
d = b % 2;	.....	.....	.....	.....

**Exercice 3 : Échange de variables**

Écrire une série d'instructions permettant d'échanger les valeurs de a et b. À la fin de cette séquence d'instruction, la valeur initiale de a doit être dans b et réciproquement.

**Exercice 4 : Débogue**

Le programme suivant comporte 13 erreurs. Trouvez-les et corrigez-les.

```
/*
/* Programme exemple */
*/

#include <std_io.h>
#include <stdlib.h>

INT Main()
{
    int a, sum;
    printf("Addition de 2 entiers a et b.\n");

    printf("a:");
    scanf("%d", a);

    printf("b:");
    scanf("%d", &b);

    somme = a + b;
    Printf("%f + %d = %d\n", a, b, sum);

    return 1;
}
}
```

### Exercice 5 : Programmation

Écrire un programme demandant sur l'entrée standard 2 nombres réels : tension et résistance. Le programme indique ensuite la valeur du courant circulant dans le circuit en utilisant la loi d'Ohm. Un test doit être prévu si la résistance est nulle.