

---

# Exercices C (3a1)

---

Laurent Signac – CC-BY-SA – 11-10-23 0952 06930fa7dd42b5ff5a00

Même s'ils contiennent du code, les exercices suivants doivent être faits **sur feuille**.

## 1 Conversions / cast

### 1.1 Conversions affectation, calcul, cast

```
int n=5,d=10;
float f1,f2,f3,f4;
f1=n/d;
f2=(float) n/d;
f3=(float) (n/d);
f4=(float) n / (float) d;
printf("%f %f %f %f\n",f1,f2,f3,f4);
```

Qu'est ce qui va s'afficher ? Expliquez.

### 1.2 Conversions dans les calculs

```
int c = 46;
int f;

f = 32 + c * 1.8
```

Que contiendra f ?

### 1.3 Perte d'information

Les conversions implicites provoquent parfois une perte d'information.

```
int i;
float f;
i = 2147483100;
f = 1.0 * i;
printf("%i %f\n", i, f);
```

Le programme précédent affiche :

```
2147483100 2147483136.000000
```

Expliquez.

## 2 Opérateurs

### 2.1 Expression booléenne

Écrivez une expression booléenne indiquant si  $n$  est une année bissextile. Les années bissextiles sont les années multiples de 4, excepté si elles sont aussi multiples de 100 sans l'être de 400.

### 2.2 Manipulations de bits

Calculez :

- $186 \& 203$
- $186 | 203$
- $186 \wedge 203$
- $186 \ll 2$

- 203 >> 3

## 2.3 Priorité des opérateurs

Que vaut :  $a = 5 * 3 + 1 > 6 + 2 \&\& 5 \& 4$  ?

## 3 Leds et masques

Nous écrivons un programme en C qui permet de piloter une série de 8 leds, par le biais d'une variable nommée OUTLEDS. Chacune des 8 leds correspond à un bit de la variable. La led 0 au bit 0 (bit de poids faible) etc... Si le bit est à 1, la led est allumée. Écrivez les instructions qui permettent de réaliser les opérations suivantes :

1. éteindre toute les leds ;
2. allumer la led 4 et éteindre les autres ;
3. allumer la led 5 sans toucher aux autres ;
4. éteindre la led 5 sans touche aux autres ;

## 4 Instructions / Expressions / Opérateurs / Priorités

### 4.1 Que va afficher le code suivant ?

```
int i=4, b, a;
b = (a=i++, 2*i);
printf("%d %d %d\n", i, b, a);
```

### 4.2 Inutilement compliqué...

Noter et expliquer les différences.

```
int i, j, k;
i = j = 0;
k = ++i, --j;
printf("%d %d %d\n", i, j, k);
```

```
int i, j, k;
i = j = 0;
k = (++i, --j);
printf("%d %d %d\n", i, j, k);
```

## 5 Boucles

```
for (i=0,k=1; i<10 ; i++,k*=2)
    printf("2**%d=%d\n", i, k);
```

Qu'est ce qui va s'afficher ?