

## ECUE «Introduction à la programmation »

Contrôle continu n°1 – 21 octobre 2013  
sans document - durée 1 heure 30

## CORRIGE

Dans tous les exercices, les entrées clavier sont indiquées en caractères gras.

**Exercice 1 (2 points)**

Dans le programme `exo1.c`, trouver:

- les 6 erreurs de compilation,
- les 2 erreurs d'exécution.

```

|| exo1.c
#include <stdio.h>
int main() (
    printf("Bonjour.\n").
    int a, b=7, c;
    printf("a = %d\n", a);
    printf("b = %d\n", b);
    c := a + b;
    printf("La somme est c\n", c);
    printf("Au revoir.\n");
    return 1;
}

```

<code>// exo1.c</code>	<b>erreur gcc 1: // au lieu de   </b>
<code>#include &lt;stdio.h&gt;</code>	<b>erreur gcc 2: include au lieu de inclure</b>
<code>int main() {</code>	<b>erreur gcc 3: { au lieu de (</b>
<code>    printf("Bonjour.\n");</code>	<b>erreur gcc 4: ; au lieu de .</b>
<code>    int a=3, b=7, c;</code>	<b>erreur logique 1: a non initialisee</b>
<code>    printf("a = %d\n", a);</code>	
<code>    printf("b = %d\n", b);</code>	<b>erreur gcc 5: manque "</b>
<code>    c = a + b;</code>	<b>erreur gcc 6: : en trop</b>
<code>    printf("La somme est %d\n", c);</code>	<b>erreur logique 2: manque %d</b>
<code>    printf("Au revoir.\n");</code>	
<code>    return 1;</code>	
<code>}</code>	

**(0.25 point par erreur trouvée)**

**Exercice 2 (6 points)**

Ecrire un programme `exo2.c` permettant à l'utilisateur d'entrer un salaire et un loyer en euros, calculant un impôt de 10% sur le salaire et affichant le montant restant en euros. Si l'utilisateur entre un salaire négatif, le programme s'arrête en affichant une erreur. Si l'utilisateur entre un loyer négatif ou supérieur au tiers du salaire, le programme s'arrête en affichant une erreur. Sinon le programme affiche l'impôt et le montant restant en euros avec deux chiffres après la virgule. La sortie du programme doit correspondre aux trois cas d'utilisation suivants.

Cas n°1:

salaire ? **-500**  
erreur: salaire < 0

Cas n°2:

salaire ? **3000**  
loyer ? **1200**  
erreur: loyer > 1000

Cas n°3:

salaire ? **3000**  
loyer ? **900**  
impot = 300.00  
reste = 1800.00**(6 points)**

**traitement du salaire: 1 point**  
**traitement du loyer: 1 point**  
**traitement de l'impot: 1 point**  
**traitement du reste: 1 point**  
**include + main + bonjour + au revoir + return 0: 1 point**  
**propreté, soin, indentation, lisibilité: 1 point**

```

// exo2.c
#include <stdio.h>
int main() {
    printf("Bonjour.\n");
    float salaire;
    printf("salaire ? ");
    scanf("%f", &salaire);
    if (salaire < 0) {
        printf("erreur: salaire < 0.\n");
        return -1;
    }
    float loyer;
    printf("loyer ? ");
    scanf("%f", &loyer);
    if (loyer < 0) {
        printf("erreur: loyer < 0.\n");
        return -1;
    }
    if (loyer > salaire/3) {
        printf("erreur: loyer > %.2f.\n", salaire/3);
        // printf("erreur: loyer > salaire/3.\n");
        return -1;
    }
    float impot = salaire/10;
    printf("impot = %.2f.\n", impot);
    float reste = salaire - loyer - impot;
    printf("reste = %.2f.\n", reste);
    printf("Au revoir.\n");
    return 0;
}

```

### Exercice 3 (7 points)

```

// exo3.c
#include <stdio.h>
int main() { int a, b, c, d, e;
    printf("a ? "); scanf("%d", &a); printf("b ? "); scanf("%d", &b);
    printf("c ? "); scanf("%d", &c); printf("d ? "); scanf("%d", &d);
    if (a>0) { if (b>1) {
        if (c>2) { if (d>3) e = a+b+c+d; else e = a+b+c-d; }
        else { if (d>3) e = a+b-c+d; else e = a+b-c-d; }
    } else {
        if (c>2) {if (d>3) e = a-b+c+d; else e = a-b+c-d; }
        else {if (d>3) e = a-b-c+d; else e = a-b-c-d; }
    }
}

```

```
} } else e = -1; printf("e = %d\n", e); return 0; }
```

1) Re-écrire le programme `exo3.c` avec une indentation correcte.

**(1.5 point)**

```
// exo3.c
#include <stdio.h>
int main() {
    int a, b, c, d, e;
    printf("a ? "); scanf("%d", &a);
    printf("b ? "); scanf("%d", &b);
    printf("c ? "); scanf("%d", &c);
    printf("d ? "); scanf("%d", &d);
    if (a>0) {
        if (b>1) {
            if (c>2) {
                if (d>3) e = a+b+c+d;
                else e = a+b+c-d;
            }
            else {
                if (d>3) e = a+b-c+d;
                else e = a+b-c-d;
            }
        }
        else {
            if (c>2) {
                if (d>3) e = a-b+c+d;
                else e = a-b+c-d;
            }
            else {
                if (d>3) e = a-b-c+d;
                else e = a-b-c-d;
            }
        }
    }
    else e = -1;
    printf("e = %d\n", e);
    return 0;
}
```

2) Donner la sortie du programme `exo3.c`, pour chacun des six cas d'entrées suivants:

a ? 0	a ? 1	a ? 1	a ? 1	a ? 1	a ? 1000
b ? 1	b ? 1	b ? 2	b ? 2	b ? 2	b ? 100
c ? 2	c ? 2	c ? 2	c ? 3	c ? 3	c ? 10
d ? 3	d ? 3	d ? 3	d ? 3	d ? 4	d ? 1

Pour chacun des six cas, les valeurs de `e` sont:

-1	-5	-2	3	10	1109
----	----	----	---	----	------

**(1.5 point: 0.25 point par sortie correcte)**

3) Re-écrire le traitement valorisant `e` en utilisant seulement quatre `if` et quatre `else`.

```

if (a>0) {
    e = a;
    if (b>1) e +=b;
    else e -= b;
    if (c>2) e +=c;
    else e -= c;
    if (d>3) e +=d;
    else e -= d;
}
else e = -1;

```

**(2 points)**

4) Même question sans utiliser ni if, ni else, ni switch mais une seule instruction.

```
e = (a>0)*((a+1) + (b>1)*2*b-b + (c>2)*2*c-c + (d>3)*2*d-d)-1;
```

**(2 points)**

## Exercice 4 (5 points)

Donner la sortie du programme `exo4.c`.

```

// exo4.c
#include <stdio.h>
int main() {
    int a=2, b=3, c=5;
    printf("a=%d, b=%d, c=%d\n", a, b, c);
    printf("%d+%d*d=%d\n", a, b, c, a+b*c);
    printf("(a+b)*c=%d\n", a, b, c, (a+b)*c);
    printf("c/b=%d\n", c, b, c/b);
    printf("c modulo b=%d\n", c, c%b, b);
    a+=b*c;    printf("a=%d, b=%d, c=%d\n", a, b, c);
    b-=a/c;    printf("a=%d, b=%d, c=%d\n", a, b, c);
    c*=a-b;    printf("a=%d, b=%d, c=%d\n", a, b, c);
    a=(-b)+(c++);    printf("a=%d, b=%d, c=%d\n", a, b, c);
    b=(c--)-(++a);    printf("a=%d, b=%d, c=%d\n", a, b, c);
    c=(-a)+(b++);    printf("a=%d, b=%d, c=%d\n", a, b, c);
    return 0;
}

```

```

a=2, b=3, c=5
2+3*5=17
(2+3)*5=25
5/3=1
5=2 modulo 3
a=17, b=3, c=5
a=17, b=0, c=5
a=17, b=0, c=85
a=84, b=-1, c=86
a=85, b=1, c=85
a=84, b=2, c=85

```

pour chaque ligne a partir de la 2e ligne: 0.5 point par ligne correcte  
si ce pré-total fait 0 et que la ligne 1 est correcte alors total = 0.5  
sinon total = pré-total.