

## ECUE «Introduction à la programmation »

Contrôle continu n°1 - 7 novembre 2011

sans document - durée 1 heure

Bruno Bouzy

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

**Exercice 1 (2 points)**

Corriger le programme suivant (8 erreurs à trouver).

```
#include <stdio.c>
int main() {
    int m, n;
    printf("m ? );
    scanf("%d", m);
    printf("m = %d\n", &m);
    printf("n ? ");
    scanf("%n", &n);
    printf("n = %d\n", n);
    printf("%d + %d = %d\n", m+n);
    printf("%d - %d = %d\n", m, n, difference);
    return (0).
}
```

```
// #include <stdio.h> // 1 (.h pas .c)
int main() {
    int m, n;
    printf("m ? "); // 2 (fermer les ")
    scanf("%d", &m); // 5 (mettre le &)
    printf("m = %d\n", m); // 6 (pas de &)
    printf("n ? ");
    scanf("%d", &n); // 7 (un d, pas un n)
    printf("n = %d\n", n);
    printf("%d + %d = %d\n", m, n, m+n); // 8 (mettre m et n)
    printf("%d - %d = %d\n", m, n, m-n); // 3 (var. non déclarée)
    return (0); // 4 (un ; pas un .)
}
```

**Exercice 2 (3 points)**

Ecrire un programme `exo2.c` permettant à l'utilisateur d'entrer un nombre d'étudiants inscrits et un nombre d'étudiants présents et affichant le pourcentage d'étudiants présents. La sortie du programme doit correspondre à l'exécution ci-dessous. On suppose que l'utilisateur entre des valeurs strictement positives. On affichera le pourcentage avec un chiffre après la virgule.

```
nombre d'etudiants inscrits ? 400
nombre d'etudiants inscrits = 400
nombre d'etudiants presents ? 250
```

```

nombre d'etudiants presents = 250
pourcentage de presences = 62.5

```

```

// exo2.c
#include <stdio.h>
int main() {
    float i, p;
    printf("nombre d'etudiants inscrits ? ");
    scanf("%f", &i);
    printf("nombre d'etudiants inscrits = %.0f\n", i);
    printf("nombre d'etudiants presents ? ");
    scanf("%f", &p);
    printf("nombre d'etudiants presents = %.0f\n", p);
    printf("pourcentage de presences = %.1f\n", 100*p/i);
    return (0);
}

```

### Exercice 3 (5 points)

Ecrire un programme `exo3.c` permettant à l'utilisateur d'entrer 3 nombres entiers `a`, `b`, `c` et les affichant en ordre croissant, séparés par un `<` ou un `=` selon qu'ils sont différents ou égaux deux à deux. La sortie du programme doit correspondre aux exécutions ci-dessous.

a ? 1	a ? 1	a ? 3	a ? 3	a ? 2
b ? 2	b ? 1	b ? 2	b ? 2	b ? 2
c ? 3	c ? 2	c ? 1	c ? 3	c ? 2
1<2<3	1=1<2	1<2<3	2<3=3	2=2=2

```

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

```

## Exercice 4 (4 points)

Donner la sortie du programme ci-dessous.

```
#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("(%d+%d)*%d=%d\n", a, b, c, (a+b)*c);
    printf("%d/%d=%d\n", c, b, c/b);
    printf("%d=%d (%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);
    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);
    return (0);
}
```

```
a=2, b=3, c=5
2+3*5=17
(2+3)*5=25
5/3=1
5=2 (3)
a=16, b=3, c=5
a=16, b=14, c=5
a=16, b=14, c=3
a=4, b=14, c=3
a=4, b=7, c=2
a=4, b=8, c=11
a=18, b=7, c=11
a=19, b=30, c=11
```

## Exercice 5 (6 points)

Ecrire un programme `exo5.c` permettant à l'utilisateur de rentrer 4 valeurs booléennes (0 pour faux, 1 pour vrai) indiquant si un plat est sucré ou pas, salé ou pas, chaud ou froid, cher ou pas. Le programme affichera:

un smiley :-D si le plat est sucré, pas salé, froid et pas cher,  
 un smiley :-) si le plat est salé, pas sucré, chaud,  
 un smiley :-/ si le plat est pas salé, pas sucré et froid,  
 un smiley :- ( si le plat est sucré, salé, froid,  
 un smiley :-| sinon.

Entre autres, la sortie du programme correspondra aux exécutions ci-dessous. On utilisera l'instruction `if` et pas l'instruction `switch`. On utilisera des variables booléennes `sale`, `sucre`, `chaud` et `cher` valant 0 ou 1.

sale ? (1/0) <b>0</b>	sale ? (1/0) <b>0</b>	sale ? (1/0) <b>1</b>
sucre ? (1/0) <b>1</b>	sucre ? (1/0) <b>0</b>	sucre ? (1/0) <b>1</b>
chaud ? (1/0) <b>0</b>	chaud ? (1/0) <b>0</b>	chaud ? (1/0) <b>1</b>
cher ? (1/0) <b>0</b>	cher ? (1/0) <b>1</b>	cher ? (1/0) <b>0</b>
:-D	:-/	:-

```
// exo5.c
#include <stdio.h>
int main() {
    int sale, sucre, chaud, cher;
    printf("sale ? (1/0) "); scanf("%d", &sale);
    printf("sucre ? (1/0) "); scanf("%d", &sucre);
    printf("chaud ? (1/0) "); scanf("%d", &chaud);
    printf("cher ? (1/0) "); scanf("%d", &cher);
    if (sucre) {
        if (sale) {
            if (!chaud) printf(":-(");
            else printf(":-|");
        }
        else {
            if (!cher) printf(":-D");
            else printf(":-|");
        }
    }
    else {
        if (sale) {
            if (chaud) printf(":-)");
            else printf(":-|");
        }
        else {
            if (!chaud) printf(":-/");
            else printf(":-|");
        }
    }
    printf("\n");
    return (0);
}
```