

Interrogation du 29/09/2005

Corrigé

Exercice I

Le programme suivant répond aux questions 1, 2, 3 et 5.

```
#include <stdio.h>
#include <math.h>
#include <assert.h>

int DixPuissance (int N)
    /* Fonction retournant 10 puissance N
       ENTREE : l'exposant N
                lorsque int est limite a 2^32-1, il convient d'avoir N<=10
       SORTIE : 10^N
    */
{
    int i, r;

    r=1;
    for (i=1;i<=N;i++) r*=10;

    return(r);
}

int ISBNcorrect (int J, int C)
    /* Fonction verifiant l'integrite structurelle d'un ISBN
       ENTREE : J les 9 chiffres de gauche de l'ISBN et C le chiffre de droite
       SORTIE : 1 si l'ISBN est structurellement valide et 0 sinon
    */
{
    int n, an;
    int S=0;
    int gauche=J;

    for (n=0;n<=9;n++)
    {
        /* Pour les 9 premier chiffres en partant de la gauche */
        /* an s'obtient par division entiere de J par 10 puissance 9-n */
        /* Le 10e est chiffre en partant de la gauche est simplement C */
        if (n<=8) {
            an=gauche/DixPuissance(8-n);
            gauche%=DixPuissance(8-n);
        }
        else an=C;
    }
}
```

```

        /* Mise a jour de la somme */
        S+=(n+1)*an;
    }

    return(S%11==0);
}

void AfficherSur9Chiffres (int J)
    /* Fonction affichant J sur 9 chiffres, precede de 0 si necessaire
       ENTREE : J un entier ayant au plus 9 chiffres
       SORTIE : Affichage de J sur 9 chiffres
    */
{
    int n, nbchiffresJ;

    assert(J<1e9);

    /* Si J commence par 1 ou plusieurs zeros, il faut les afficher */
    /* E(log(J)/log(10))+1 donne le nombre de chiffres de J */
    /* On aurait egalement pu faire un boucle avec une division entiere par 10 */
    nbchiffresJ=floor(log(J)/log(10))+1;

    /* On affiche les 9-nbchiffres zeros manquant */
    n=9-nbchiffresJ;
    while (n>0) {
        printf("0");
        n--;
    }

    /* On affiche le reste de J */
    printf("%d",J);
}

void AfficherISBN13 (int J)
    /* Fonction affichant l'ISBN-13 a partir de de l'ISBN
       ENTREE : J les 9 chiffres de gauche de l'ISBN
       SORTIE : Affichage de l'ISBN-13
    */
{
    int n, an, chiffre, D;
    int moinsunpuissancen=1; /* initialisation avec n=0 */
    int S=0;
    int gauche=J;

    /* Calcul de la somme de controle jusqu'a l'avant dernier chiffre */
    for (n=0;n<=11;n++)
        {
            if (n==0) an=9;
            if (n==1) an=7;
            if (n==2) an=8;
            if ((n>=3)&&(n<=11)) {

```

```

        chiffre=n-3;
        an=gauche/DixPuissance(8-chiffre);
        gauche%=DixPuissance(8-chiffre);
    }
    S+=(2-moinsunpuissancen)*an;
    moinsunpuissancen*=-1;
}

/* D est le chiffre entre 0 et 9 tel que S+D soit divisible par 10 */
if (S%10==0) D=0;
else D=10-S%10;

/* Affichage du numero ISBN-13 */
printf("978");
AfficherSur9Chiffres(J);
printf("%d",D);
}

int main() {
    int J, C;

    printf("Entrer les 9 chiffres de gauche de l'ISBN : \n");
    scanf("%d",&J);
    printf("Entrer le chiffre a droite (10 au lieu X) de cet ISBN : \n");
    scanf("%d",&C);

    if (ISBNcorrect(J,C)) {
        printf("L'ISBN-13 est ");
        AfficherISBN13(J);
        printf("\n");
    }
    else printf("L'ISBN entre est incorrect\n");

    return(0);
}

```

4. La fonction AfficherISBN13 que nous avons écrite affiche le résultat mais ne retourne rien car elle est de type void. Une fonction qui retourne le résultat et n'affiche rien aurait été de type int, ce résultat aurait alors pu être affecté à une variable ou être utilisé comme argument d'une fonction (par exemple printf pour l'afficher).

En général il vaut mieux retourner une valeur qu'afficher une valeur puisque cela donne plus de flexibilité. Toutefois dans le cas précis de ce programme, une difficulté serait apparue : l'ISBN étant sur 13 chiffres, il dépasse la capacité du type int sur la plupart des machines 32 bits.

Exercice II

La première affectation qui concerne n et la deuxième qui concerne m ne posent pas de problème, ces variables reçoivent respectivement les valeurs 3 et 4.

A la ligne suivante, l'opérande à droite du signe égal vaut 1 puisqu'il s'agit de la division entière de 4 par 3. La variable x reçoit donc la valeur 1.

L'affectation suivante ne pose pas de problème, la variable p reçoit -1 .

La ligne suivante effectue les opérations suivantes, dans l'ordre

1. Incrémentation de n , cette variable vaut donc 4.
2. Somme de n et m et affectation à q , qui prend donc la valeur $4 + 4 = 8$.
3. Incrémentation de m , cette variable vaut donc 5.

On affiche alors les variables n , m , p , q et x , c'est-à-dire $n=4$, $m=5$, $p=-1$, $q=8$, $x=1.000000$.