

Interrogation du 27/09/2006

Corrigé

Exercice I

Le programme suivant répond aux quatre questions.

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

void AfficherSurNChiffres (int X, int N)
/* Afficher X sur N chiffres, precede de zeros si necessaire
   ENTREE : X le nombre a afficher, N le nombre de chiffres
   SORTIE : X sur N chiffres
   REMARQUE : N doit etre superieur ou egal au nombre de chiffres de X
*/
{
    int NombreDeChiffres, Y, i;

    /* On compte le nombre de chiffres de X
       On aurait pu \’egalement se baser sur le logarithme en base 10 de X */

    NombreDeChiffres=0;
    Y=X;

    while (Y>0) {
        Y /= 10;
        NombreDeChiffres++;
    }

    /* Si le nombre de chiffres de X est strictement superieur a N
       la fonction n’aurait pas du etre appelee avec ces arguments */

    assert (NombreDeChiffres<=N);

    /* Affichage des 0
       Il faut en afficher N-NombreDeChiffres */

    for (i=1;i<=N-NombreDeChiffres;i++)
        printf("0");

    printf("%d",X);
}

void PresenterSIREN (int SIREN)
/* Affiche le numero SIREN sous la forme xxx xxx xxx
```

```

    ENTREE : SIREN le numero a afficher
    SORTIE : Pas de valeur de retour mais affichage a l'ecran du numero formate
*/
{
    int TroisPremiersChiffres, TroisChiffresDuMilieu, TroisDerniersChiffres,
        SixPremiersChiffres;

    /* Determination des trois groupes */

    TroisDerniersChiffres = SIREN % 1000;
    SixPremiersChiffres = SIREN / 1000;
    TroisChiffresDuMilieu = SixPremiersChiffres % 1000;
    TroisPremiersChiffres = SixPremiersChiffres / 1000;

    /* Affichage des trois groupes */

    AfficherSurNChiffres(TroisPremiersChiffres,3); printf(" ");
    AfficherSurNChiffres(TroisChiffresDuMilieu,3); printf(" ");
    AfficherSurNChiffres(TroisDerniersChiffres,3);
}

int EstCorrectSIREN (int SIREN)
{
    int X, i, chiffre, p, somme;

    X = SIREN;
    somme = 0;

    for (i=1;i<=9;i++){

        /* Determination du chiffre courant */
        chiffre = X % 10;

        if (i%2==0) {

            /* Chiffre en position paire */

            p = chiffre * 2;

            /* Si le produit p >= 10 on fait la somme de ses chiffres
            Cela revient a retrancher 9 */

            if (p>=10) p-=9;
        }

        else {

            /* Chiffre en position impaire, la multiplication par 1
            donne ce chiffre */

            p = chiffre ;
        }
    }
}

```

```

        /* On met a jour la somme et X */
        somme += p;
        X /= 10;
    }

    /* Le SIREN est valide si somme est un multiple de 10 */

    return (somme%10==0);
}

int main() {
    int SIREN;

    printf("Entrer le numero SIREN\n");
    scanf("%d",&SIREN);

    printf("Le numero SIREN ");
    PresenterSIREN(SIREN);
    printf(" est ");
    if (EstCorrectSIREN(SIREN))
        printf("correct");
    else
        printf("incorrect");
    printf("\n");
}

```

Exercice II

Le programme suivant répond aux deux questions.

```

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

double f(double x, double y)
/* Fonction a integrer */
{
    return (2*x*x+y*y*y);
}

double integralef (double a, double b, double c, double d, int nx, int ny)
/* Integration de la fonction f sur la pave [a,b]x[c,d]
   ENTREE : Les reels a, b, c, d et le nombre de subdivision nx et ny
            nx et ny correspondent aux subdivisions en x et y respectivement
   SORTIE : Une approximation de l'integrale de f sur [a,b]x[c,d]
*/
{
    double hx, hy, somme;
    int i, j;

```

```

/* Le pave doit etre non vide et non trivial */
assert((b>a)&&(d>c));

/* Le nombre de subdivisions doit etre strictement positifs */
assert((nx>0)&&(ny>0));

/* Calcul des pas hx et hy */

hx = (b-a)/nx;
hy = (d-c)/ny;

/* Approximation du volume par la somme de parallelepipedes rectangles

La hauteur du parallepipede (i,j) est f(a+(i+0.5)*hx,c+(j+0.5)*hy)
Sa base a pour aire hx*hy.
Son volume est donc f(a+(i+0.5)*hx,c+(j+0.5)*hy)*hx*hy

Pour etre plus efficace, le produit par hx*hy est factorise */

somme = 0;

for (i=0;i<nx;i++)
  for (j=0;j<ny;j++)
    somme += f(a+(i+0.5)*hx,c+(j+0.5)*hy);

somme *= hx*hy;

/* La variable somme contient une approximation de l'integrale */

return (somme);
}

int main() {

double a=1, b=2, c=4, d=8;
int nx=1000, ny=2000;

printf("Approximation avec nx = %d et ny = %d :\t %f\n",
      nx,ny,integralef(a,b,c,d,nx,ny));
printf("Valeur de l'integrale calculee a la main :\t %f\n",2936.0/3);

return 0;
}

```