

## Interrogation du 25/09/2002

*Correction*

### Exercice I

1. Le programme contient les erreurs de syntaxe suivantes :
  - la fonction test renvoie un int (la variable Resultat) alors que son en-tête annonce un void.
  - présence d'une virgule après l'instruction d'initialisation de la boucle for : il faut un ; .
  - la variable i de la fonction test n'est pas déclarée.
  - la variable N n'est pas déclarée.
  - la variable N n'est pas initialisée (il faut appeler scanf avec l'adresse de N)
  - présence du caractère \ avant le second " dans la fonction printf.
2. La fonction test compte les carrés parfaits entre 0 et N.

### Exercice II

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

void PiecesARendre(int MontantARendre, int TypeDePiece)
    /* Affiche le nombre de pieces (de type TypeDePiece) a rendre
    pour le montant MontantARendre
    ENTREE : MontantARendre est le montant a rendre en centimes d'euros
    TypeDePiece est la valeur de la piece en centimes d'euros
    SORTIE : Aucune mais le programme affiche les messages a l'ecran
    */
{
    int ARendre;
    ARendre=MontantARendre/TypeDePiece;
    printf("Rendre %d piece(s) de %.2f euro(s)\n",ARendre,(float)TypeDePiece/100);
}

int ResteARendre(int MontantARendre, int TypeDePiece)
    /* Indique ce qui reste a rendre une fois que les pieces TypeDePiece ont
    ete rendues.
    ENTREE : MontantARendre est le montant a rendre en centimes d'euros
    TypeDePiece est la valeur de la piece en centimes d'euros
    SORTIE : Montant restant a rendre
    */
{
    return (MontantARendre%TypeDePiece);
}
```

```

}

void RenduMonnaie (int PrixArticle, int MontantPaye)
    /* Affiche le nombre de pieces a rendre afin de minimiser le nombre de
    pieces rendues.
    ENTREE : PrixArticle est le prix de l'article achete
    MontantPaye est le montant paye par l'acheteur
    SORTIE : Aucune mais le programme affiche les messages a l'ecran
    */
{
    int MontantARendre, Reste;

    MontantARendre=MontantPaye-PrixArticle;
    assert(MontantARendre>=0);

    PiecesARendre(MontantARendre,200);
    Reste=ResteARendre(MontantARendre,200);
    PiecesARendre(Reste,100);
    Reste=ResteARendre(MontantARendre,100);
    PiecesARendre(Reste,50);
    Reste=ResteARendre(MontantARendre,50);
    PiecesARendre(Reste,20);
    Reste=ResteARendre(MontantARendre,20);
    PiecesARendre(Reste,10);
    assert(ResteARendre(MontantARendre,10)==0);
}

int main()
    /* On verifie sur l'exemple
    Cette partie n'etait pas demandee en interrogation
    */
{
    RenduMonnaie(430,570);
}

```