

Décomposition de la monnaie [ss05] - Exercice

Karine Zampieri, Stéphane Rivière

Unisciel  algoprogram  Version 16 mai 2018

Table des matières

1	Décomposition de la monnaie / pgmonnaie	2
1.1	Décomposition du problème	2
1.2	Procédures et Programme	3
2	Références générales	4

C - Décomposition de la monnaie (Solution)



Mots-Clés Algorithmes paramétrés ■

Requis Structures de base, Structures conditionnelles ■

Difficulté ●○○ (30 min) ■



Objectif

Cet exercice reprend l'exercice @[Décomposition de la monnaie] en réalisant une décomposition en procédures et fonctions et traite le cas des coupures nulles.

(image : google/images)



1 Décomposition de la monnaie / pgmonnaie

1.1 Décomposition du problème

L'exercice @[Décomposition de la monnaie] décompose une somme d'argent (en euros) en son équivalent minimal en billets de 100 €, 50 €, 10 €, et de pièces de 2 € et 1 €.



Programme initial

```
#include <stdio.h>
int main()
{
    int somme;
    printf("Somme a decomposer? ");
    scanf("%d",&somme);
    int b100 = somme/100;
    int r100 = somme%100;
    printf("==> %4d billet(s) de 100 euros, reste %d\n",b100,r100);
    int b50 = r100/50;
    int r50 = r100%50;
    printf("==> %4d billet(s) de 50 euros, reste %d\n",b50,r50);
    int b10 = r50/10;
    int r10 = r50%10;
    printf("==> %4d billet(s) de 10 euros, reste %d\n",b10,r10);
    int p2 = r10/2;
    int r2 = r10%2;
    printf("==> %4d pieces(s) de 2 euros, reste %d\n",p2,r2);
    int p1 = r2;
    printf("==> %4d pieces(s) de 1 euro\n",p1);
}
```

Exemple d'exécution

(Avec cet algorithme)

```
Somme à décomposer? 1254
==> 12 billet(s) de 100 euros, reste 54
==> 1 billet(s) de 50 euros, reste 4
==> 0 billet(s) de 10 euros, reste 4
==> 2 piece(s) de 2 euros, reste 0
==> 0 piece(s) de 1 euro
```



Proposez une décomposition en procédures et fonctions en écrivant les profils de ces dernières.

Aide simple

Constatez que trois lignes se reproduisent.

Solution simple

Les trois lignes qui se reproduisent ont la forme suivante : Une procédure `calcAfficher(somme, valeur, nc,` permettra de faire les calculs et l’affichage.

1.2 Procédures et Programme

Écrivez une procédure `calcAfficher(somme, valeur, nc, rt, typec)` qui, pour une somme d’argent `somme` (entier) et une valeur de coupures `valeur` (entier), calcule le nombre de coupures dans `c` (entier), le reste à décomposer dans `rt` (entier) et affiche le résultat suivant (où `[x]` désigne le contenu de `x`) :

```
==> [nc] [typec] de [valeur] euros, reste [rt]
```

Le type de la coupure est défini par le caractère `typec` qui vaut ‘b’ (billet) ou ‘p’ (pièce). Dans le cas où `nc` est nul, n’affichez pas le texte.

Solution Paramètres

Entrants : `somme, valeur, typec`

Sortants : `nc, rt`



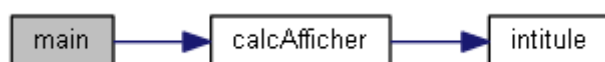
Validez votre procédure avec la solution.

Solution C @[pgmonnaie.c]

```
void calcAfficher(int somme, int valeur, int* nc, int* rt, char typec)
{
    *nc = somme/valeur;
    *rt = somme%valeur;
    char intitule[10];
    if (typec=='b') strcpy(intitule, "billet");
    else strcpy(intitule, "piece");
    if (*nc != 0)
    {
        printf("==> %4d %s(s) de %d euros, reste %d\n", *nc, intitule, valeur, *rt);
    }
}
```



Ré-écrivez un programme en utilisant la procédure `calcAfficher`.

**Aide simple**

Pour calculer le nombre de pièces de 1 euro, ajoutez une variable `r1` pour l’appel.



Testez. Exemple d'exécution :

```
Somme à décomposer? 1254
==> 12 billet(s) de 100 euros, reste 54
==> 1 billet(s) de 50 euros, reste 4
==> 2 piece(s) de 2 euros, reste 0
```



Validez votre programme avec la solution.

Solution C @[pgmonnaie.c]

```
int main()
{
    int somme;
    printf("Somme a decomposer? ");
    scanf("%d",&somme);
    int b100;
    int r100;
    calcAfficher(somme,100,&b100,&r100,'b');
    int b50;
    int r50;
    calcAfficher(r100,50,&b50,&r50,'b');
    int b10;
    int r10;
    calcAfficher(r50,10,&b10,&r10,'b');
    int p2;
    int r2;
    calcAfficher(r10,2,&p2,&r2,'p');
    int p1;
    int r1;
    calcAfficher(r2,1,&p1,&r1,'p');
    int rs = 100*b100+50*b50+10*b10+2*p2+1*r1;
    printf("==> %d\n",rs);
}
```

2 Références générales

Comprend [Rohaut-JV1 :c4 :xm] ■