

Licence 1 - IF3

Examen Programmation C/C++

8 janvier 2004

Consignes : le sujet de cet examen consiste à écrire les différentes parties d'un seul programme, lisez donc toutes les questions pour vous faire une idée globale de ce que vous allez devoir faire.

Ce sujet va vous permettre de faire l'analyse d'un problème simple : comment rendre la monnaie automatiquement. Ce problème est découpé en plusieurs parties, les plus indépendantes possible, qu'il est nettement préférable de traiter dans l'ordre proposé.

On suppose que le monnayeur :

- dispose d'une somme S en Euros pour rendre la monnaie ;
- et qu'il peut utiliser des pièces de 1, 2, 5, 10, 20 et 50 centimes d'euro, de 1 et 2 euros, et enfin des billets de 5, 10 et 20 euros. Pièces et billets de toute valeur seront par la suite appelées « unités ».

La somme S est décidée au départ et indique à tout moment la quantité d'argent disponible pour rendre la monnaie. Le monnayeur doit également savoir combien d'unités sont disponibles pour chaque catégorie afin de pouvoir rendre la monnaie en utilisant le moins d'unités possible.

Exemple : le monnayeur dispose de 10 pièces de 50 cents, de 8 pièces de 1 euro, de 4 pièces de 2 euros, etc.

1 Organisation des données

Proposez une représentation des unités (pièces et billets) et une autre pour représenter le stock de monnaie disponible.

Comment connaître le nombre d'unités disponibles pour une catégorie décrite ci-dessus ? Rédigez la fonction C/C++ qui réalise cette opération, vous préciserez également les entrées et les sorties de cette fonction.

2 Entrées

Proposez une fonction C/C++ qui permet de demander quelle somme doit être encaissée, quelle est la somme due, et qui retourne la somme qui doit être rendue (sans donner plus de détail). Cette fonction n'a qu'un seul « retour », la somme à rendre.

3 Fonction max

Écrire une fonction C/C++ qui prend en entrée une somme à rendre et retourne la plus grosse unité de monnaie (autorisée et disponible dans le « stock ») inférieure ou égale à cette valeur.

4 Fonction principale

Écrire l'algorithme du monnayeur qui détermine comment rendre la monnaie sur une somme. Cet algorithme prend en paramètres la somme à rendre et le stock d'unités disponibles. L'algorithme doit déterminer la solution qui utilise le moins d'unités afin de continuer à fonctionner le plus longtemps possible sans réapprovisionnement. L'idée est de rendre la monnaie en utilisant l'unité de plus grande valeur.

Écrire une fonction C/C++ qui met en œuvre l'algorithme précédent.

Exemple et aide (non notée)

Imaginons que le monnayeur dispose de 10 pièces de 1, 2, 5, 10, 20 et 50 cents, et qu'on veut rendre 45 cents. Quelle est la meilleure solution ? 2 pièces de 20 cents et 1 pièce de 5 cents semble une solution correcte. Comment déterminer ce résultat ? Pour y parvenir (et donc pour résoudre le problème général posé), essayez de répondre aux questions suivantes :

1. Quelle est la plus grande catégorie inférieure à la somme à rendre ?
2. Combien d'unités sont disponibles dans cette catégorie ? Combien d'unités peut-on utiliser dans cette catégorie ?
3. Combien reste-t-il à rendre ?
4. Comment finir de rendre la monnaie ?
5. A quoi sert la fonction *max* (cf. 3) ?