

TP2 : Expressions, if-else

Dans ce TP et pour les suivants, on prendra soin de la présentation des programmes :

- Un commentaire d'entête décrira l'objectif du programme
- Les variables seront commentées dans leur déclaration pour préciser leur rôle.
- Les blocs d'instructions seront indentés pour visualiser leur hiérarchie
- Les } seront à l'aplomb des {

1. si-alors-sinon

Il est parfois nécessaire de vérifier que des données saisies sont correctes ou, tout au moins, ont de grandes chances d'être correctes. Le numéro de sécurité sociale (nss) est composé de 13 chiffres et d'une clé à 2 chiffres qui offre cette possibilité en vérifiant l'égalité suivante :

$$\text{clé} = 97 - (\text{nss} \bmod 97)$$

Ecrivez le programme qui saisie un n° de sécurité sociale, sa clé, et valide ou non ce n° en affichant *ok* ou *non valide* suivant le cas.

Attention, pour stocker un entier à 13 chiffres, il faut utiliser le type `long` (64 bits).

2. Calcul sur les entiers et gestion de l'affichage

Ecrivez le programme qui lit une somme d'argent et décompose cette somme en billets de 50, 20, 10 et pièces de 1 €. On fera en sorte que les sommes nulles n'apparaissent pas à l'affichage.

Exemple, la saisie de 95 doit provoquer l'affichage de *1x50€ 2x20€ 5x1€*

Algorithme à affiner :

- lire la somme
- calculer le nombre de billets de 50 et l'afficher si non nul
- calculer le reste de la somme à décomposer
- calculer le nombre de billets de 20 et l'afficher si non nul
- compléter...*

3. si imbriqués

Une horloge digitale indique un temps t en *heure, minute, seconde* (3 entiers). Ecrire un programme qui lit un temps t sous la forme de 3 entiers (h, m, s) et calcule le temps t+1s en respectant un affichage heures-minutes-secondes.

Exemple : pour la saisie de 12 35 59, on obtiendra l'affichage *12h 36mn 0s*

Algorithme à affiner :

- définir 3 variables entières h, m et s
- lire h, m et s au clavier (on supposera ces données valides)
- ajouter 1 seconde à s
- si s atteint 60 s alors
 - remettre le compteur s à 0
 - ajouter 1 mn à m
- ... complétez l'algorithme avant de le traduire en C*
- fin-si
- écrire la nouvelle heure

On testera ce programme sur les temps suivants : 10 30 00

10 30 59

23 59 59