

Université Paris XII
IUT de Sénart-Fontainebleau
Département Informatique
Algorithmique
2004/2005
Première année

PARTIEL 1

2 heures

Seuls les notes manuscrites et les listings portant le nom de l'étudiant (dans le programme et sur chaque page) sont permis à titre de documentation.

Exercice 1.- Donner une description précise de chacune des sept expressions suivantes en langage C :

2, 2.0, 2L, 2.0f, '2', "2", '\2'.

Exercice 2.- (**Monnaie**)

Écrire un programme C d'aide à une caissière pour rendre la monnaie. On entre la somme due puis la somme remise par le client. Le programme calcule la différence et affiche la monnaie que doit rendre la caissière : le nombre d'Euros, le nombre de pièces de 50 centimes, de pièces de 20 centimes, de pièces de 10 centimes, de pièces de 5 centimes, de pièces de 2 centimes et enfin de pièces de 1 centime.

[*Un exemple de session est la suivante :*

Montant : 25.31

Somme remise : 30

Rendre 4 Euros

1 pièces de 50 centimes

0 pièces de 20 centimes

1 pièces de 10 centimes

1 pièces de 5 centimes

2 pièces de 2 centimes

0 pièces de 1 centime.

On utilisera une alternative pour marquer le pluriel ou le singulier de 'Euro'. Par contre 'pièces' sera invariant. On utilisera la division euclidienne pour déterminer le nombre de pièces de chaque sorte.]

Exercice 3.- (Suite bitone)

Une **suite monotone** d'entiers naturels est une suite croissante ou décroissante. Une **suite bitone** est une suite monotone, suivie d'une autre suite monotone (éventuellement vide). Par exemple les trois suites suivantes (1, 3, 5, 2, 4, 8), (2, 2, 3, 5, 3, 2) et (4, 3, 1, 2, 7) sont bitones.

Écrire un programme C qui demande un certain nombre d'entiers naturels, la valeur sentinelle étant -1 pour terminer la saisie, qui affiche si la suite ainsi saisie est bitone ou non.

[On remarquera qu'il y a changement du sens de croissance si on a au moins trois éléments dans la suite et si la différence entre l'élément en cours et le précédente est de signe différent de la différence du précédent et de l'antépénultième.]

Exercice 4.- (Conversions)

- 1°) Écrire une fonction `in2cm()` en langage C dont l'argument et le type de retour de retour sont des réels, qui renvoie la longueur en centimètres alors qu'elle est fournie en pouces.

[1 pouce = 2,54 cm.]

- 2°) Écrire une fonction `cm2in()` en langage C dont l'argument et le type de retour de retour sont des réels, qui renvoie la longueur en pouces alors qu'elle est fournie en centimètres.

- 3°) Écrire un programme C qui demande une longueur, qui demande l'unité ('i' ou 'c') et qui affiche la longueur dans l'autre unité.