

La plupart des exercices ont été imaginés par Henri Garreta.

1 Echauffement

1. Calcul de la somme des carrés des n premiers entiers : écrire un programme qui calcule la somme des carrés des n premiers entiers :

$$\sum_{k=0}^n k^2.$$

2. Calcul de la somme de l’inverse du carré des n premiers entiers : écrire un programme qui calcule la somme

$$\sum_{k=1}^n \frac{1}{k^2}.$$

Vers quelle valeur converge cette somme ?

3. Calcul de la factorielle : écrire un programme qui calcule $n! = 1 \times 2 \times \dots \times (n - 1) \times n$.

2 Jour de la semaine

Ecrivez un programme qui calcule le jour de la semaine correspondant à une date donnée, supposée correcte, exprimée sous forme de trois nombres entiers j (jour), m (mois) et a (année). Exemple

```
$ python joursemaine.py
jour? 24
mois? 9
annee? 2009
le 24/9/2009 est un jeudi
$
```

Utilisez les formules suivantes :

$$m_1 = \begin{cases} m - 2 & \text{si } m \geq 3 \\ m + 10 & \text{sinon} \end{cases} \quad a_1 = \begin{cases} a & \text{si } m \geq 3 \\ a - 1 & \text{sinon} \end{cases},$$

n_s = le siècle compté à partir de 0 (i.e. tel qu’exprimé par les deux premiers chiffres de a_1)

a_s = l’année dans le siècle (exprimée par les deux derniers chiffres de a_1)

$$f = j + a_s + \frac{a_s}{4} - 2 \times n_s + \frac{n_s}{4} + \frac{26 \times m_1 - 2}{10}.$$

Dans ces conditions, le jour de la semaine est donné par le reste r de la division de f par 7 ($r = 0 \Rightarrow$ dimanche, $r = 1 \Rightarrow$ lundi, $r = 6 \Rightarrow$ samedi). Ci-dessus, les barres de fraction indiquent des divisions entières.

3 Nombre mystère

Votre programme va vous permettre de trouver, en plusieurs étapes, la valeur d’un nombre qui a été choisi aléatoirement par la machine et que vous ne connaissez pas. L’exécution du programme devra se dérouler de la manière suivante :

1. La machine choisit de façon aléatoire un nombre entier N entre 0 et 100.
2. Vous proposez un nombre entre 0 et 100, et la machine vous dit s'il est inférieur, supérieur ou égal à N .
3. Trois cas se présentent
 - (a) S'il est inférieur à N , l'exécution reprend au point 2.
 - (b) S'il est supérieur à N , l'exécution reprend au point 2.
 - (c) S'il est égal à N , vous avez gagné et le jeu est fini.

A la fin du jeu, la machine vous indiquera en combien d'étapes vous avez trouvé N . Ensuite, votre programme vous demandera si vous voulez recommencer (retour au point 1.), ce qui vous permettra de jouer plusieurs fois de suite pendant la même session.

Faire ce même programme en inversant maintenant les rôles : l'utilisateur doit choisir un nombre aléatoire et c'est à la machine de le deviner (en un nombre minimal d'étapes).

4 Rendu de monnaie

Écrivez un programme qui lit une somme exprimée en euros, sans centimes, et affiche le nombre minimal de billets et de pièces de 2 et 1 euros nécessaires pour la composer.

L'exécution du programme doit rendre le résultat suivant :

```
$ python monnaie.py
Somme ? 1949
1949 = + 3 x 500 + 2 x 200 + 0 x 100 + 0 x 50 + 2 x 20 + 0 x 10 + 1 x 5 + 2 x 2 + 0 x 1
```

5 La date du lendemain

Ecrivez un programme qui acquiert au clavier une date, exprimée sous forme de trois nombres entiers j (jour), m (mois) et a (année) et qui affiche la date du lendemain. Exemple :

```
$ python lendemain.py
jour? 30
mois? 9
annee? 2009
aujourd'hui: 30 9 2009
demain: 1 10 2009
$
```

Vous pouvez supposer que la date saisie est correcte.

Suggestion. Vous aurez fait le plus gros du travail quand vous aurez écrit une expression logique, portant sur les nombres j , m et a , qui traduit exactement la proposition « (j, m, a) est le dernier jour du mois ».

Rappel. Les années bissextiles sont les années multiples de 4 qui ne sont pas multiples de 100, et les années multiples de 400.