Exercices sur la récurrence

1) La suite (u_n) est définie par $u_0 = 0$ et $u_{n+1} = \frac{1}{2}u_n + 3$. Faire afficher les termes u_1 à u_n, n étant saisi au clavier.

2) Conjecture Syracuse (ou de Kollek, ou de Collatz) :

La suite (u_n) est définie par l'entier naturel u_0 et par la relation suivante :

si u_n est impair $u_{n+1} = 3u_n + 1$ et si u_n est pair $u_{n+1} = \frac{u_n}{2}$, u_0 étant saisi au clavier.

Faire afficher tous les termes de la suite jusqu'à ce que l'un d'eux soit égal à 1.

3) Algorithme Babylonien ou méthode de Newton pour calculer $\sqrt{a}$

En partant de a et u_0 saisis au clavier, avec la formule de récurrence $u_{n+1} = \frac{1}{2}(u_n + \frac{a}{u_n})$, donner le premier terme de la suite vérifiant : $\left|\frac{u_{n+1} - u_n}{u_n}\right| < 10^{-5}$

4) La suite de Fibonacci est définie par $u_0 = u_1 = 1$ et par la relation : $u_{n+2} = u_{n+1} + u_n$.
Faire afficher les termes u_2 à u_n, n étant saisi au clavier.

Exercices sur les sommes ou produits

5) Faire afficher la somme $S_1(n) = 1 + 2 + ... + n$, n étant saisi au clavier.
Reprendre l'exercice avec la somme $S_2(n)$ des carrés puis la somme $S_3(n)$ des cubes des entiers de 1 à n.

6) On démontre que la limite de la somme $S(n)$ suivante, lorsque n tend vers $+\infty$, est égale à $\frac{\pi}{4}$:

$$S(n) = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + ... + \frac{(-1)^n}{2n+1}$$

Faire afficher une valeur approchée de π en calculant $S(n)$, n étant saisi au clavier.

7) Un nombre est parfait s'il est égal à la somme de ses autres diviseurs. C'est le cas, par exemple, de 6 = 1 + 2 + 3 ou de 28 = 1 + 2 + 4 + 7 + 14.
Ecrire un programme testant si un nombre est parfait.
Modifier ce programme pour faire afficher la liste des nombres parfaits compris entre deux entiers saisis au clavier.

8) Un nombre est premier s'il admet exactement deux diviseurs : lui-même et l'unité.
Ecrire un programme testant si un nombre est premier.
Modifier le programme précédent pour faire afficher la liste des nombres premiers compris entre deux entiers saisis au clavier.

Chaînes de caractères

9) Sur les enveloppes utilisées par la Poste, les codes postaux écrits en chiffres par l'envoyeur sont codés par des bandes colorées imprimées dans la partie inférieure.
Les codages des cinq chiffres du code postal sont écrits de droite à gauche, jointivement. La table de conversion est la suivante :

<table>
<tr><td>0:</td><td>..||||</td><td>1:</td><td>.|.|||</td><td>2:</td><td>.||.||</td><td>3:</td><td>.|||.|</td><td>4:</td><td>|..|||</td></tr>
<tr><td>5:</td><td>|.|.||</td><td>6:</td><td>|.||.|</td><td>7:</td><td>||..||</td><td>8:</td><td>||.|.|</td><td>9:</td><td>|||..|</td></tr>
</table>

où le . représente un espace de code ASCII 32

et | est le caractère de code ASCII 124

On suppose que ce codage est contenu dans le tableau Tableau de taille 10 contenant les chaînes de la table de conversion.
Ainsi Tableau(4) est la chaîne '.|||.|' (en considérant un décalage de 1 entre la valeur et son repérage)

Ecrire la fonction permettant d'écrire le codage c correspondant au code postal donné par *n*.

En vue d'un décodage, on doit s'assurer tout d'abord de la validité de la lecture du code (par un scanner ou lecture optique). Ecrire la fonction retournant la valeur vraie si le code est possible et la valeur faux sinon sachant que le codage de chaque chiffre doit comprendre 4 barres et nécessairement une barre en première position (dans son écriture qui est de droite à gauche ! *cf* table de conversion).

Ecrire la fonction permettant, après s'être assuré de la fiabilité du codage c, de déterminer le code postal en l'entier n.

Tableau

10) a) Ecrire une fonction permettant de déterminer si l'entier *n* donné en paramètre est un nombre de Janus. Un nombre ou un mot de Janus peut se lire indifféremment dans les 2 sens, comme 13431 ou ressasser.
b) Ecrire un programme qui conserve dans un tableau tous les nombres de 1 à *N* , entier donné au clavier, dont le carré est un nombre de Janus.
Vous afficherez le tableau obtenu.

Le roi Janus
l'homme aux deux visages

Caricature représentant Louis XVI sous la forme de Janus aux deux visages. Qui d'un coté prête serment aux représentants de la Nation

"Je soutiendrais la Constitution"

et de l'autre affirme aux représentants de l'église

"Je détruirai la Constitution ".

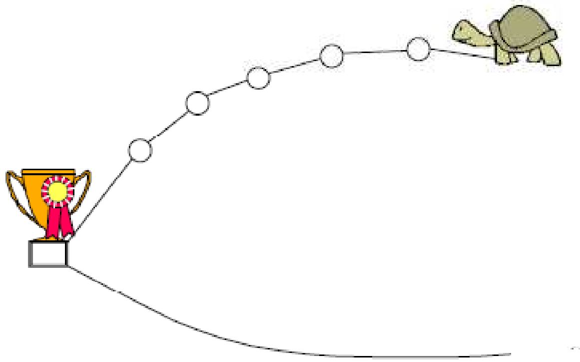
Divers

9) Le jeu se joue à deux : toi (l'élève) contre l'ordinateur (le prof). C'est chacun à son tour de jouer.
Le tapis comporte une seule rangée de 21 allumettes (on prendra la variable *n* pour le nombre d'allumettes).
Lorsque c'est à ton tour de jouer, tu dois enlever une, deux ou trois allumettes.
Celui qui retire la dernière allumette perd la partie.
Il s'agit de te laisser jouer le premier et de te faire perdre à tous les coups en construisant le programme dont l'algorithme est le suivant :
tant que la partie n'est pas finie :

- afficher le message "Il reste xxxx allumettes, vous devez en retirer entre 1 et 3" où xxxx représente le nombre d'allumettes restantes.
- demander à l'utilisateur un nombre entre 1 et 3. Redemander ce nombre s'il ne se trouve pas entre 1 et 3, et mettre à jour le nombre d'allumettes disponibles.
- l'ordinateur doit enlever à son tour des allumettes.
  La stratégie qu'il doit adopter est d'enlever le complément à 4 d'allumettes
  (C'est à dire : si vous enlevez 1 allumette, l'ordinateur en enlève 3, si
  vous en enlevez 2 l'ordinateur 2 et si vous en enlevez 3 l'ordinateur en enlève 1).
- S'il ne reste plus d'allumette après votre tour de jeu, afficher : « L'ordinateur a gagné ».

Si c'est après le tour de l'ordinateur (???), afficher : « L'ordinateur a perdu ».

10) On lance un dé.
Si le 6 sort, le lièvre gagne.
Sinon, la tortue avance d'une case.
On continue jusqu'à ce qu'il y ait un gagnant.
Quelle est la situation la plus enviable, celle du lièvre ou celle de la tortue ?



a) Expliquer pourquoi la formule tapée (ici sous Scilab) : $int(n\times \text{rand}()) + 1$ permet d'obtenir un entier quelconque (et équiprobable) entre 1 et n.
b) Il s'agit dans cette question de simuler une seule partie en construisant une somme contenant les déplacements de la tortue : tant que le résultat n'est pas 6 ou que la tortue n'est pas arrivée, retirer un nombre quelconque entre 1 et 6.

Afficher alors le gagnant de la partie.

c) Modifier le programme précédent de manière à réaliser 100 simulations de parties.

Quelle est la situation la plus avantageuse, celle de la tortue ou celle du lièvre ?

Vous proposerez un affichage pour répondre à cette question.

11) Conjecture d'Erdös
Pour tout entier $n > 1$, il existe trois entiers x, y et z tels que :

$$\frac{4}{n} = \frac{1}{x} + \frac{1}{y} + \frac{1}{z}$$

Construire un programme permettant de vérifier cette conjecture pour $n \in \{1; 2; \ldots; 100\}$, dans ce cas, vous supposerez que les entiers x, y et z sont à rechercher parmi les entiers entre 1 et n^2.
S'il existe un entier pour lequel l'égalité n'est pas vérifiée, vous ferez afficher "La conjecture est fausse" sinon vous ferez afficher "La conjecture semble vraie".

Remarque : *C'est un cas particulier de fractions égyptiennes où l'on cherche à écrire un rationnel comme somme d'un nombre donné d'inverses d'entiers, conjecture vérifiée pour* $n \leqslant 10^8$.
De même existe la conjecture de Sierpinski qui suppose que pour $n > 1$, *il existe trois entiers naturels a, b et c tels que* $\frac{5}{n} = \frac{1}{a} + \frac{1}{b} + \frac{1}{c}$.