

Informatique : T.D. n°5

Exercice 1

Rappelons l'exercice 1 du T.D. n°2 :

On demandait d'écrire un programme qui fait trouver un nombre mystérieux à l'utilisateur.

On définit donc une constante entière *myst* dont on fixera la valeur.

On demande à l'utilisateur un nombre entier *n*.

Si *n* est égal à *myst*, on affiche " gagné ", sinon, on affiche " perdu ", en précisant "c'est plus" ou "c'est moins".

- 1) Charger le fichier *mystere.pas* et enregistrer immédiatement sous un nouveau nom.
- 2) Améliorons ce jeu : l'utilisateur ne joue plus une seule fois, mais joue jusqu'à ce qu'il gagne.
- 3) Améliorer encore ce programme en affichant à la fin le nombre de coups nécessaires pour gagner.

Exercice 2

Soit (u_n) la suite définie par
$$\begin{cases} u_0 = \frac{1}{2} \\ \forall n \in \mathbb{N}, u_{n+1} = u_n^2 - u_n + 1 \end{cases}$$

On admet que $\lim_{n \rightarrow +\infty} u_n = 1$

1. Ecrire un programme qui calcule les termes successifs de (u_n) jusqu'à ce que $|u_n - 1| \leq 10^{-3}$, et affiche la valeur de *n* correspondante.
2. Ecrire un programme qui demande un réel ε à l'utilisateur, et qui renvoie la première valeur de *n* pour laquelle $|u_n - 1| \leq \varepsilon$.

On testera ce programme avec des valeurs de ε strictement positives et très proches de 0 (10^{-2} , 10^{-3} , ...).

Exercice 3 (Extrait E.M.L. 2004)

Soit u la suite définie par :
$$\begin{cases} u_0 = 1 \\ \forall n \in \mathbb{N}, u_{n+1} = f(u_n) \end{cases}$$
, où f est la fonction définie par :

$$\forall x \in \mathbb{R}, f(x) = \frac{2e^x}{\sqrt{1+x^2}}.$$

Ecrire un programme en Pascal qui calcule et affiche le plus petit entier *n* tel que $u_n > 10^6$

Exercice 4

Soit (u_n) la suite définie par :
$$\begin{cases} u_1 = 2 \\ \forall n \geq 1, u_{n+1} = \frac{1}{2}u_n + \frac{2}{3} \end{cases}$$
. Pour $n \geq 1$, on pose $S_n = \sum_{k=1}^n u_k$.

Déterminer la première valeur de *n* pour laquelle $S_n > 100$.