

## Énoncé

On considère la suite  $(u_n)$  définie par 
$$\begin{cases} u_0 = 0 \\ \text{pour tout } n \in \mathbb{N}, u_{n+1} = 3u_n - 2n + 3 \end{cases}.$$

1. Calculer  $u_1$  et  $u_2$ .

a. Démontrer par récurrence que, pour tout entier naturel  $n$ ,  
 $u_n \geq n$

.

b. Justifier que la suite  $(u_n)$  est croissante.

c. Déterminer la limite de la suite  $(u_n)$ .

2. Pourquoi peut-on affirmer qu'il existe au moins un entier  $n_0$  tel que, pour tout  
 $n \geq n_0$

,  
 $u_n \geq 10^p$

? L'ensemble des entiers  $n_0$  tels que  
 $n \geq n_0 \Rightarrow u_n \geq 10^p$

est donc un sous-ensemble non vide de

$\mathbb{N}$

, ce sous-ensemble admet donc un plus petit élément  $m_0$ . On s'intéresse maintenant au plus petit entier  $m_0$ .

3. Proposer un script en Python qui, pour une valeur de  $p$  donnée en entrée, affiche en sortie la valeur du plus petit entier  $m_0$  tel que  
 $n \geq m_0 \Rightarrow u_n \geq 10^p$

. Déterminer à l'aide du programme cet entier  $m_0$  pour la valeur  $p = 3$ .

## La bonne méthode

1. On remplace  $n$  par 0 puis par 1 dans la relation définissant la suite.

2. a. On procède, comme l'énoncé le demande, par récurrence. On prend bien soin d'argumenter correctement l'hérédité.

2. b. On évalue le signe de  $u_{n+1} - u_n$  à la lumière de ce qui précède.

2. c. On utilise le théorème de comparaison.

3. Il faut revenir à la définition du résultat obtenu à la question précédente.

4. Il s'agit d'un algorithme dit de seuil, on construit une boucle tant que.