

www.freemaths.fr

TLE

Technologique Mathématiques

Suites
arithmético-géométriques



CORRIGÉ DE L'EXERCICE

LE FORAGE

CORRECTION

1. a. Calculons U_3 :

$$U_3 = 2\,000 \times (1,008)^{3-1} \Leftrightarrow U_3 = 2\,000 \times (1,008)^2$$

$$\Rightarrow U_3 = 2\,032,13\text{€}.$$

Ainsi, le forage entre 20 et 30 mètres coûte: 2 032,13€.

1. b. Déterminons le coût total de forage des 30 premiers mètres:

Soit C_3 , le coût total des 30 premiers mètres.

$$C_3 = U_1 + U_2 + U_3 \Rightarrow C_3 = 6\,048,13\text{€}.$$

Au total, le coût total de forage des 30 premiers mètres est de: 6 048,13€.

2. a. a1. Exprimons U_{n+1} en fonction de U_n :

$$U_{n+1} = 2\,000 \times (1,008)^n \text{ et } U_n = 2\,000 \times (1,008)^{n-1},$$

$$\text{par conséquent: } U_{n+1} = 1,008 \times U_n.$$

Au total, pour tout entier naturel n : $U_{n+1} = 1,008 \times U_n$.

2. a. a2. Précisons le nature (U_n):

Ici: (U_n) est une suite géométrique de raison $q = 1,008$ et de premier terme $U_1 = 2\,000$.

2. b. Déduisons-en le pourcentage demandé:

$$\text{Il s'agit de calculer: } \left(\frac{U_{n+1} - U_n}{U_n} \right) \times 100.$$

$$\begin{aligned} \text{Or: } \frac{u_{n+1} - u_n}{u_n} &= \frac{2\,000 \times (1,008)^n - 2\,000 \times (1,008)^{n-1}}{2\,000 \times (1,008)^{n-1}} \\ &= \frac{1,008 - 1}{1} \\ &= 0,008. \end{aligned}$$

$$\text{D'où: } \left(\frac{u_{n+1} - u_n}{u_n} \right) \times 100 = 0,8\%.$$

Ainsi, le pourcentage d'augmentation du coût du forage de la $(n + 1)$ -ième dizaine de mètres par rapport à celui de n -ième dizaine de mètres est égal à: 0,8%.

3. a. Résumons les résultats obtenus dans un tableau:

Nous avons le tableau suivant:

Valeur de i		2	3	4	5
Valeur de u	2000	2016	2032,13	2048,39	2064,77
Valeur de S	2000	4016	6048,13	8096,51	10161,29

3 b. Déterminons la valeur de S affichée en sortie et interprétons:

La valeur de S affichée en sortie est: $\approx 10\,161,29$.

10 161,29€ correspond au coût de forage à 50 mètres de profondeur.

4. a. Calculons la profondeur maximale:

Nous savons que pour tout entier naturel non nul n :

$$S_n = u_1 + u_2 + u_3 + \dots + u_n$$

$$\text{cad: } S_n = -250\,000 + 250\,000 \times (1,008)^n.$$

Or, le budget donné est: $B = 125\,000$ €.

D'où, la profondeur maximale est telle que: $S_n \leq B$.

$$S_n \leq B \Leftrightarrow -250\,000 + 250\,000 \times (1,008)^n \leq 125\,000$$

$$\Leftrightarrow (1,008)^n \leq 1,5$$

$$\Leftrightarrow n \times \ln(1,008) \leq \ln(1,5)$$

$$\Rightarrow n \leq 50,88.$$

Comme n est un entier naturel non nul, nous prendrons $n = 50$, et la profondeur maximale du puits que l'on peut espérer forer avec le budget B est de:
 $50 \times 10 = 500$ mètres (U_{50}).

4. b. Modifions l'algorithme précédent:

L'algorithme modifié est:

Initialisation:

u prend la valeur 2 000

S prend la valeur 2 000

n prend la valeur 1

Traitement:

Tant que $S \leq 125\,000$ **faire**

u prend la valeur $u \times 1,008$

S prend la valeur $S + u$

n prend la valeur $n + 1$

Fin du Tant que

Sortie:

Afficher $(n - 1) \times 10$