

On appelle pourcentage de compression d'une image, le pourcentage de réduction de sa taille en ko (kilo-octets) après compression.

Une image a une taille initiale de 800 ko. Après une première compression, sa taille est de 664 ko.

1. Calculer le pourcentage de réduction associé à cette première compression.

Dans la suite de l'exercice, on fixe le pourcentage de réduction à 17 %. On effectue n compressions successives.

Pour tout entier naturel n , on note t_n la taille de l'image en ko après n compressions.

On a donc $t_0 = 800$.

2. Pour tout entier naturel n , exprimer t_{n+1} en fonction de t_n et en déduire la nature de la suite (t_n) .
3. Pour tout entier naturel n , exprimer t_n en fonction de n .

Afin de déterminer le nombre minimal n de compressions successives à effectuer pour que cette image ait une taille finale inférieure à 50 ko, on considère la fonction Python suivante:

```
def nombreCompressions(A):
    t = 800
    n = 0
    While t > A:
        t = t*0,83
        n = n+1
    return n
```

4. Préciser, en justifiant, le nombre A de sorte que l'appel `nombreCompressions(A)` renvoie le nombre de compressions successives à effectuer que l'on cherche à déterminer.
5. Quel est le nombre minimal de compressions successives à effectuer pour que ce fichier ait une taille finale inférieure à 50 ko ?