

# POINTS DE FIDÉLITÉ

## Compétences visées

Les compétences visées sont proposées à titre indicatif et peuvent être modifiées par le professeur.

- **Modéliser** : Traduire en langage mathématique une situation réelle [...] Utiliser, comprendre, élaborer une simulation numérique [...] utilisant un logiciel.
- **Calculer** : Mettre en œuvre des algorithmes simples.
- **Communiquer** : Développer une argumentation mathématique correcte à l'écrit ou à l'oral.

## Situation déclenchante

Deux sites de musique en ligne proposent des abonnements aux mêmes tarifs. Pour fidéliser leurs abonnés, ils ont mis en place des systèmes de « points de fidélité » différents. A l'ouverture du compte sur chacun des sites, 1000 points fidélité sont offerts.

Chaque mois, les deux sites ajoutent des points de fidélité aux comptes de leurs abonnés :

- L'offre du site Music OnLine : « **Chaque mois, 20% de points en plus.** »
- L'offre de SoundMelody : « Chaque mois, une prime de fidélité de 200 points supplémentaires par mois d'abonnement. »

Tous les 20 000 points de fidélité, un cadeau peut être gagné.

Pierre veut souscrire un abonnement mais il hésite entre les deux offres.

Les cadeaux à 20 000 points et à 60 000 points le tentent particulièrement.

## Problématique : Comment aider Pierre dans son choix ?

Proposer une méthode qui permettrait de répondre à la problématique.



Appeler le professeur

# PROGRAMMER UN ALGORITHME SIMPLE POUR COMPARER DEUX SUITES

## Proposition de résolution

Comparer les deux offres revient à étudier deux suites  $u_n$  et  $v_n$  correspondant au nombre de points de fidélité sur le compte d'un abonné de l'un des deux sites au bout de  $n$  mois d'abonnement.

Pour l'offre du site Music OnLine :  $u_0 = 1000$  et  $u_{n+1} = 1,2u_n$ .

Pour l'offre de SoundMelody :  $v_0 = 1000$  et  $v_{n+1} = v_n + 200n$ .

Pour aider Pierre dans son choix, il faut déterminer quelle offre permet d'obtenir le cadeau à 20 000 points le plus rapidement, de même pour celui à 60 000 points. Pour cela on utilise un algorithme (ci-contre) que l'on va programmer sur la calculatrice.

Créer un nouveau programme que l'on nommera « SEUIL ». L'instruction **Prompt S** permet de demander à l'utilisateur de définir le seuil à atteindre.

Les instructions **While**, **End**, **Prompt** et **Disp** s'obtiennent en

cliquant sur la touche  et respectivement les touches  et  dans l'onglet CTL et les touches ,  dans l'onglet E/S.

Les variables U et V contiennent les valeurs de  $u_n$  et  $v_n$ . Elles sont donc initialisées à 1000.

La variable N contenant le rang des termes est initialisée à 0.

On utilise une boucle **while** car on ne sait pas combien de termes de la suite on doit calculer.

A la fin de la boucle, on affiche le rang du dernier terme calculé.

Pour la deuxième suite, on réinitialise N à 0 et on écrit le même type d'instructions.

Une fois le programme saisi, on quitte l'édition de programme et on l'exécute dans l'onglet EXEC.

En saisissant 20000 puis 60000 pour le seuil, on obtient les résultats suivants.

On peut donc **conclure** :

Pierre a intérêt à choisir l'abonnement de SoundMelody pour obtenir le cadeau à 20 000 points plus rapidement mais il devra choisir l'abonnement de Music OnLine pour le cadeau à 60 000 points.

En langage naturel pour la suite ( $u_n$ ) :

- Entrer** le seuil S à atteindre
- Initialiser** la variable U à 1000
- Initialiser** la variable N à 0
- Tant que** U < S
- Affecter** à N la valeur N+1
- Affecter** à U la valeur 1,2U
- FinTantque**
- Afficher** N

```
NORMAL FLOTT AUTO REEL RAD MP
PROGRAM: SEUIL
: Prompt S
: 1000→U
: 1000→V
: 0→N
: While U<S
: N+1→N
: 1.2U→U
: End
: Disp N
```

```
: 0→N
: While V<S
: N+1→N
: V+200N→V
: End
: Disp N
: █
```

```
NORMAL FLOTT AUTO REEL RAD MP
S=?20000
17
14
..... Fait
prgmSEUIL
S=?60000
23
24
..... Fait
```

QRCode

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus !

