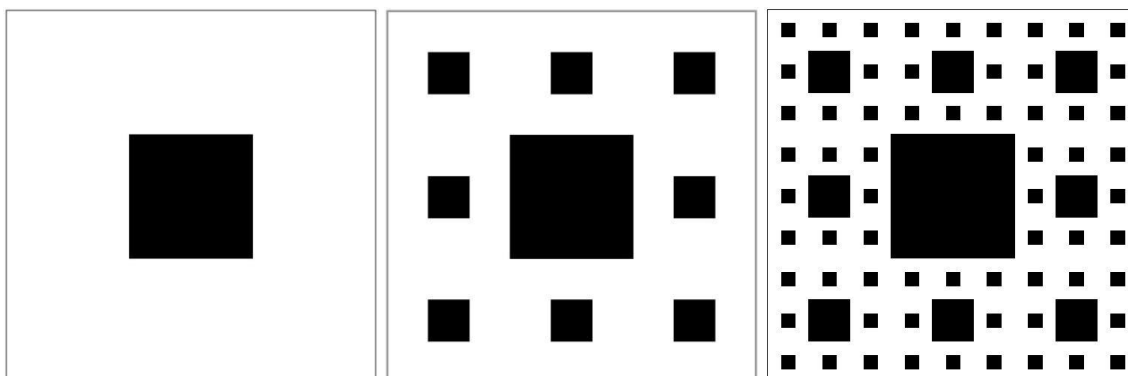


Tapis de Sierpinski



Niveau ***

Présentation :

A partir d'un carré de côté donné, on le découpe en neuf carré égaux et on retire la pièce centrale.

On réitère la procédure aux carrés restants.

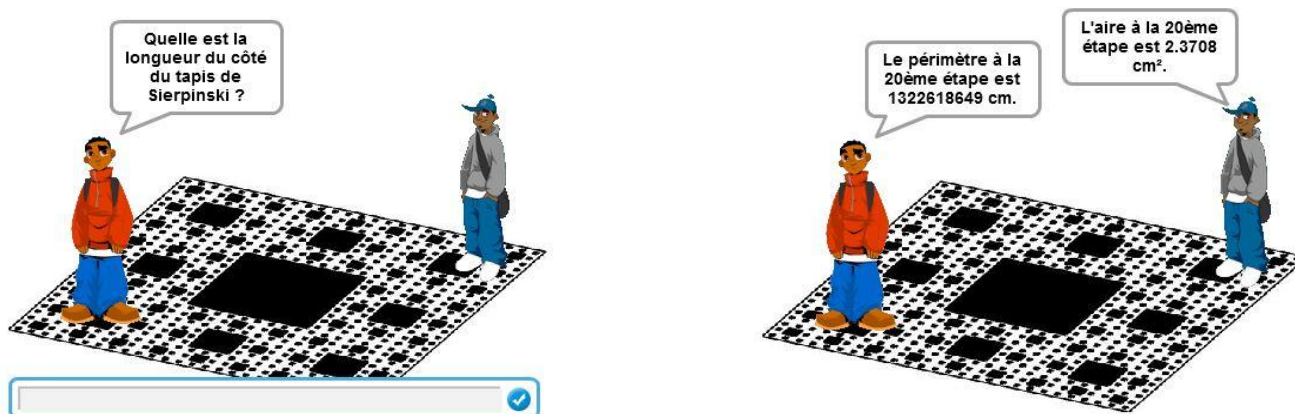
On se propose de calculer l'aire et le périmètre du tapis à chaque étape à partir de suites récurrentes.

Quelques informations en vue du codage :

Penser à demander à l'utilisateur les deux informations initiales nécessaires aux calculs.

Pour l'aire une variable suffit.

Pour le périmètre, le calcul direct d'une puissance dans Scratch n'étant pas possible, il faut utiliser des variables intermédiaires obtenues en interprétant la formule.



Deuxième version : Quelques idées pour mettre en forme les scripts. Niveau **

Calcul du périmètre :

$$P_0 = 4l$$

$$P_n = P_{n-1} + 8^{n-1} \times 4 \times \frac{l}{3^n}$$

Calcul de l'aire :

$$A_0 = l^2$$

$$A_{n+1} = \frac{8}{9} A_n$$

Variables : (décocher l'affichage de toutes ces variables à l'exception de Longueur initiale)

- Aire
- Périmètre
- Longueur initiale (qu'on ne modifiera pas en vue de l'affichage final)
- l (longueur du côté de chaque nouveau carré créé)
- n (nombre d'itérations)
- NbeNxCarrés

Premier lutin :

- Demander à l'utilisateur les deux informations initiales nécessaires aux calculs (*longueur initiale* et *l* d'un côté et *n* de l'autre)
- Initialiser le *périmètre* et l'*aire*.
- Initialiser la variable *NbeNxCarrés*.
- Créer une boucle permettant d'obtenir le *périmètre* et l'*aire*, sans oublier d'ajuster les valeurs de *l* et de *NbeNxCarrés*.
- Afficher le *périmètre* final et envoyer un message à tous les lutins.

Deuxième lutin :

A réception du message, afficher l'*aire*.

Amélioration de l'affichage des résultats :

Optimiser l'affichage en arrondissant l'*aire* et le *périmètre*.

Troisième version : Avec les captures d'écran des scripts.

Niveau *

Calcul du périmètre :

$$P_0 = 4l$$

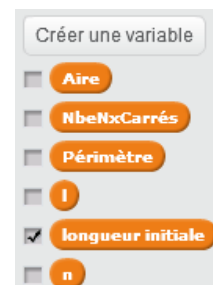
$$P_n = P_{n-1} + 8^{n-1} \times 4 \times \frac{l}{3^n}$$

Calcul de l'aire :

$$A_0 = l^2$$

$$A_{n+1} = \frac{8}{9} A_n$$

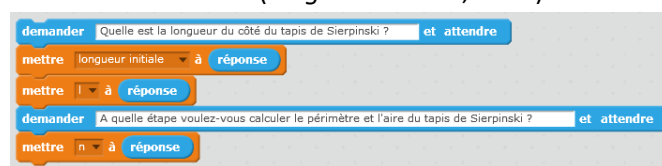
Variables : (décocher l'affichage de toutes ces variables à l'exception de Longueur initiale)



- Aire
- Périmètre
- Longueur initiale (qu'on ne modifiera pas en vue de l'affichage final)
- l (longueur du côté de chaque nouveau carré créé)
- n (nombre d'itérations)
- NbeNxCarrés

Premier lutin :

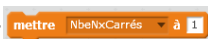
- Demander à l'utilisateur les deux informations initiales nécessaires aux calculs (*longueur initiale, l et n*).



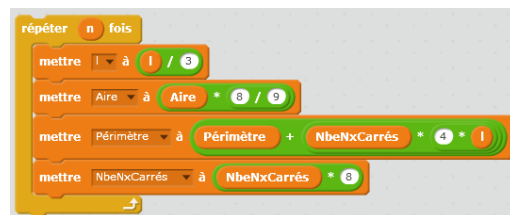
- Initialiser le *périmètre* et *l'aire*.



- Initialiser la variable *NbeNxCarrés*.



- Créer une boucle permettant d'obtenir le périmètre et l'aire, sans oublier d'ajuster les valeurs de *l* et de *NbeNxCarrés*.



- Afficher le *périmètre* final et envoyer un message à tous les lutins.



Deuxième lutin :

- A réception du message, afficher *l'aire*.



Amélioration de l'affichage des résultats :

- Optimiser l'affichage en arrondissant *l'aire* et le *périmètre*.

