Fiche 5

Le stylo

Exercice 1 Dessiner un 4

Le stylo

Dessinez, comme vous le feriez à la main, un chiffre 4 qui ressemble à celui-ci : 

On pourra choisir où placer le crayon avant de cliquer sur le drapeau vert. Il faudra donc vérifier que le stylo est placé suffisamment au centre pour qu'il ne sorte pas de la scène lors de ses déplacements. Dans le cas contraire, le crayon dira à l'utilisateur de mieux le centrer.

Exercice 2 Un carré et un pentagone

Copier un script

Utilisez deux lutins (le chat et le cafard par exemple).
Le chat dessinera un pentagone rouge et le cafard un carré orange.

Ajoutez une commande qui permet de tout effacer quand on presse sur la touche « espace ».

Exercice 3 Polygones réguliers à *n* cotés

Il serait intéressant de pouvoir dessiner n'importe quel polygone convexe à *n* côtés, plutôt que seulement un pentagone.
Trouvez une formule qui donnera l'angle de rotation du lutin à chaque sommet en fonction de *n*.

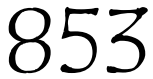
Dans la liste des palettes, choisissez **Capteurs** et prenez :

Vous pouvez changer le texte à votre guise.

Un message apparaîtra sur la scène où vous pourrez écrire un nombre, qui sera stocké dans :

Enfin, pour indiquer l'angle de rotation, allez chercher dans la palette **Opérateurs** ce dont vous aurez besoin.
Attention ! Pour que le polygone soit bien dessiné, le lutin ne doit pas sortir de la scène.

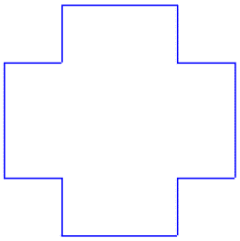
Exercice 4 Dessiner 853

De la même manière que l'exercice 1, dessinez le nombre : 

Indication : en informatique, les cercles sont des polygones avec beaucoup de côtés...

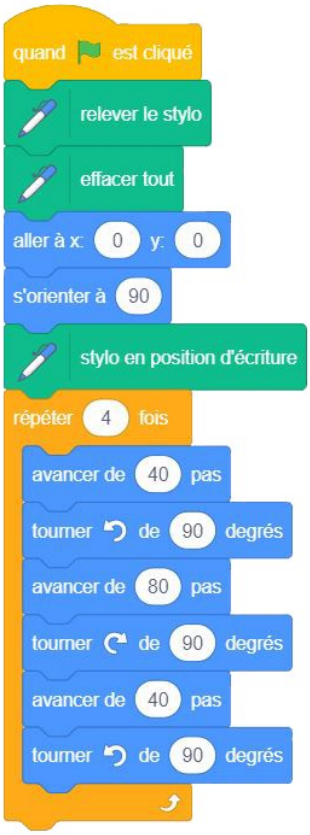
Exercice 5 Les croix

Voici une croix :



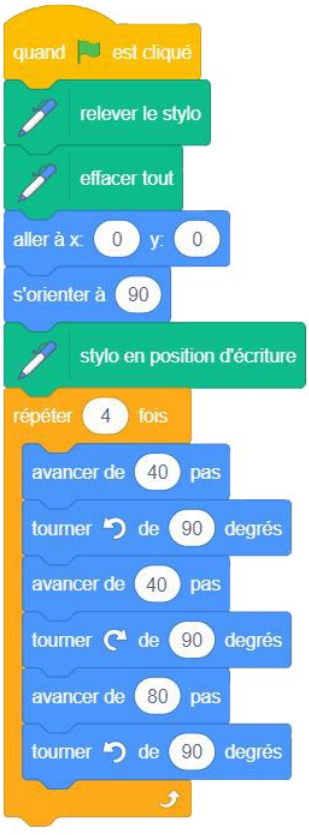
a. Par lequel des trois scripts ci-après a-t-elle été dessinée ?
Répondez à cette question en exécutant « à la main » ces scripts.

Script A

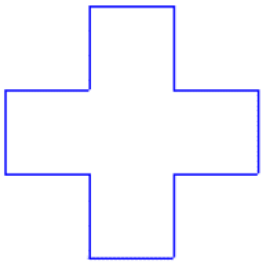


Script B

Script C



b. Comment modifier le script choisi à la question **a** pour obtenir la croix ci-dessous ?



Les blocs

Bloc

Un *bloc* est une suite d'instructions. Vous pouvez ensuite utiliser le nom du bloc comme une nouvelle instruction.

Ci-contre, le bloc « carré » dessine un… carré. On pourra utiliser cette nouvelle instruction ailleurs dans le script principal ou dans un autre bloc.

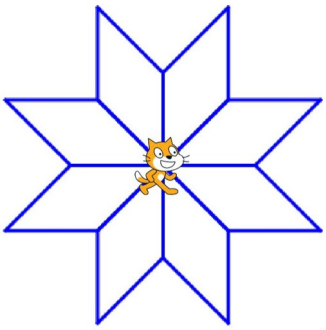
Décomposer un problème en sous-problèmes est une des bases de la programmation. L'utilisation de blocs aide l'écriture des longs scripts en découpant le programme en plusieurs parties plus simples.

Cela facilite aussi la lecture et la compréhension des scripts.

Exercice 6 Une rosace

Bloc avec un paramètre

Dessinez cette rosace :



a. Utilisez un bloc pour dessiner un losange, que vous ferez tourner huit fois pour obtenir la rosace.

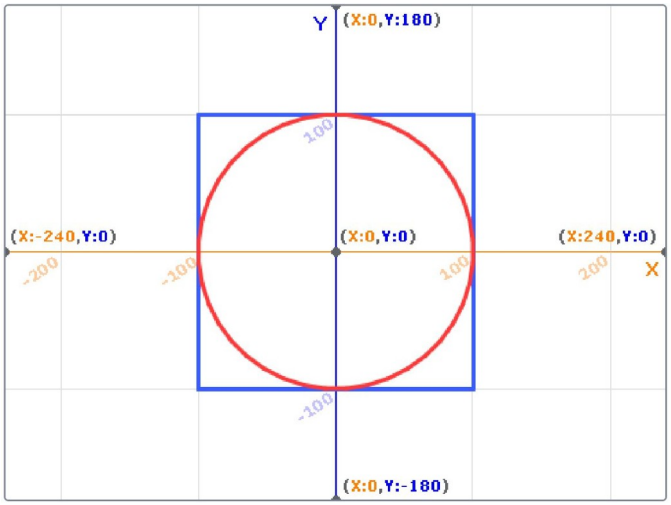
b. Généralisez la partie **a** de l'exercice pour dessiner une rosace comportant *n* losanges.

Exercice 7* Le cercle inscrit

Partie a

Bloc avec un paramètre

Dessinez un carré centré à l'origine de côté 200, puis dessinez un cercle de rayon 100 inscrit dans ce carré.



Partie b

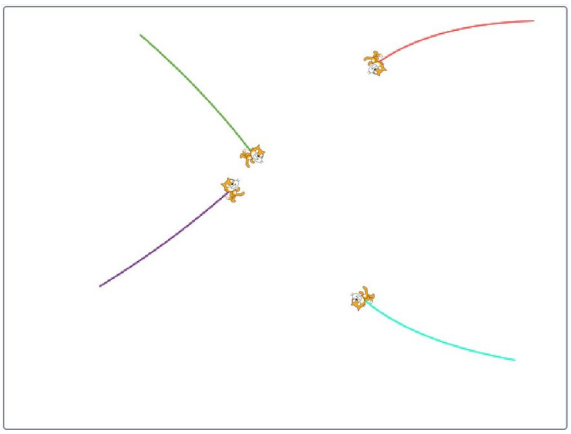
Même exercice que la partie **a**, mais en plus général. Implémentez :

- un bloc dessinant un carré dont on donnera en paramètres les coordonnées du centre et la longueur de son côté ;
- un bloc dessinant un cercle dont on donnera en paramètres les coordonnées du centre et le rayon.

Exercice 8 Courbes de poursuite

Plusieurs chats se poursuivent. Le chat 1 poursuit le chat 2, le chat 2 poursuit le chat 3, …, et le dernier chat poursuit le chat 1.

Mettez en scène ce scénario avec le nombre de chats que vous souhaitez, en donnant aux chats des positions de départ aléatoires. Les chats laissent une trace derrière eux :

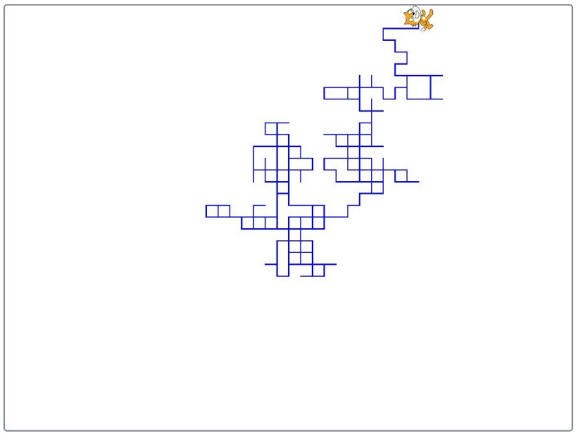


Les chats dessinent ce qu'on appelle en mathématiques des « courbes de poursuite » ou « courbes du chien ».

Commencez par programmer un seul chat en réfléchissant bien à tous les détails. Une fois que cela fonctionne, dupliquez votre chat autant de fois que vous voulez et faites pour chaque chat les petites modifications nécessaires.

Exercice 9* La marche aléatoire

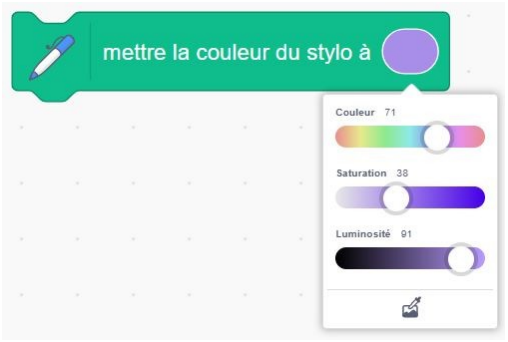
En partant du centre de la scène, Scratch se déplace de 10 pas aléatoirement dans une des quatre directions, jusqu'à ce qu'il ait fait 1000 déplacements ou qu'il touche un bord.



Nombres aléatoires

Les couleurs dans Scratch

Pour définir une couleur, Scratch utilise le modèle TSL pour Teinte Saturation Luminosité (en anglais HSL pour Hue Saturation Lightness).

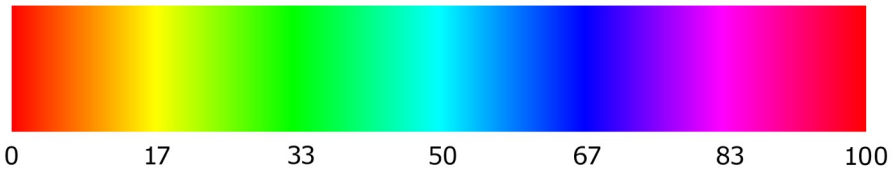


On peut aussi copier une couleur déjà présente dans la scène avec la pipette. Indispensable si on a besoin d'une couleur précise.

La teinte est donnée par un nombre compris entre 0 et 99 (voir la bande colorée ci-dessous).
La saturation mesure la distance depuis une couleur « achromatique » (blanc, gris ou noir).
La luminosité mesure la position de la couleur entre le noir et le blanc.
En jouant avec les trois curseurs, on peut produire 100 x 101 x 101 = 1'020'100 couleurs.

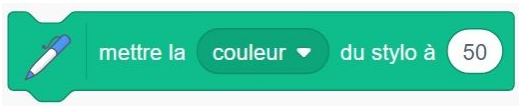
Pour en savoir plus :

fr.wikipedia.org/wiki/Teinte_saturation_luminosité



On peut aussi donner le « numéro » de la couleur, qui correspond à la teinte du modèle TSL.

Cette brique permet aussi de fixer la saturation, la teinte et la transparence (100 = invisible).



Ainsi, 0 = rouge, 10 = orange, 17 = jaune, …, 50 = cyan, … Pratique pour faire des dégradés de couleurs, comme dans l'exercice 9.

Exercice 10 Le cercle chromatique

Dessinez le cercle chromatique de Scratch :