

Utiliser le code pour transformer des triangles

Vous allez créer de nouveaux sous-programmes en écrivant un code qui transformera les sommets d'un triangle.

Voici le lien à l'application Scratch de la Fiche 5 qui comprend des sous-programmes.

<https://scratch.mit.edu/projects/1193312388/editor/>

Partie 1 : Translation horizontale du triangle

1. Écrivons un code d'un sous-programme qui déplacera horizontalement les trois sommets en décalant la coordonnée x de chaque point.

Tout d'abord, créez un nouveau bloc pour le sous-programme en sélectionnant **Mes Blocs**. Nommez-le **décalerX**.

Il apparaîtra dans la zone d'édition du code. Faites-le glisser dans une zone où il y a de la place pour mettre des blocs en dessous.

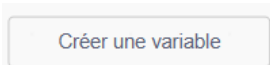
2. Vous utiliserez des blocs **mettre** pour décaler la coordonnée x de chaque point.

Pour ce faire, vous aurez besoin d'une variable qui contiendra la valeur par laquelle les coordonnées x seront décalées.

- Sélectionnez **Variables**.



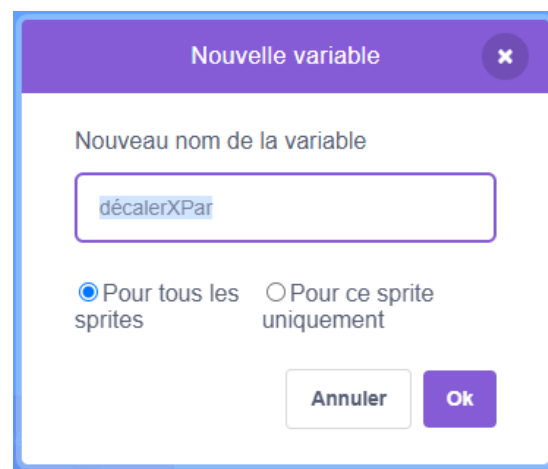
Sélectionnez **Créer une variable**.



- Nommez la variable **décalerXPar**.

Si vous avez un compte Scratch, sauvegardez le projet dans votre compte en sélectionnant **Remix** en haut de l'écran.

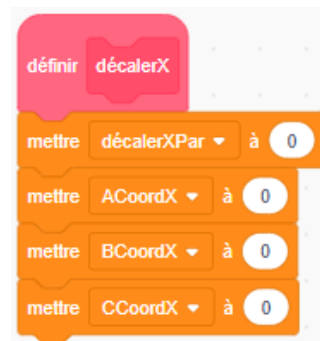
Une connexion n'est pas nécessaire pour travailler avec le code, mais vous ne pourrez pas sauvegarder vos modifications sans elle.



Utiliser le code pour transformer des triangles (suite)

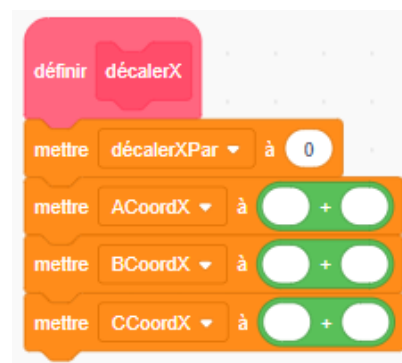
3. Glissez quatre blocs **mettre** sous le sous-programme **décalerX** et ajustez les menus déroulants comme indiqué à droite.

Vérifiez bien que vous avez sélectionné les coordonnées x de A, B et C des blocs **mettre** et non les coordonnées y.



4. Vous utiliserez des blocs **opérateurs** pour ajouter la valeur par laquelle les coordonnées x se décaleront aux valeurs originales des coordonnées x.

- Dans **Opérateurs**, glissez un **opérateur d'addition** à l'intérieur de chacun des blocs **mettre**, comme indiqué à droite.
- Dans **Variables**, glissez les variables de la coordonnée x dans la première partie de chaque **opérateur d'addition**.
- Glissez la variable **décalerXPar** dans la deuxième partie de l'**opérateur d'addition** pour chaque bloc **mettre**.



Remarque : Cette opération peut être délicate — prenez votre temps. Si les pièces ne fonctionnent pas comme prévu, il suffit de les démonter et de réessayer !

Utiliser le code pour transformer des triangles (suite)

5. Commençons par décaler les coordonnées x de 100.

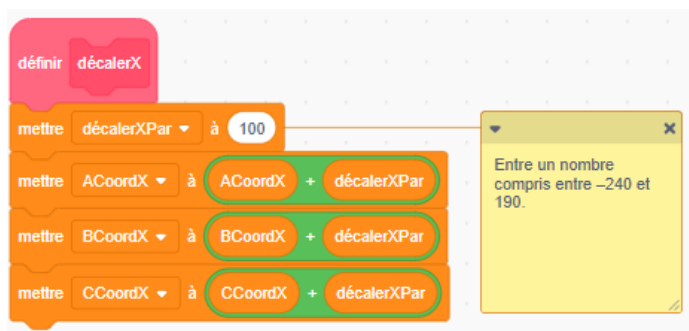
Pour ce faire, entrez 100 dans le bloc **mettre** pour la variable **décalerXPar**, comme indiqué à droite.



6. Ajoutez un commentaire pour indiquer à toute personne qui utilise votre code l'intervalle de valeurs qu'elle peut entrer pour la variable **décalerXPar**.

Pour ce faire, cliquez avec le bouton droit de la souris sur le bloc **mettre décalerXPar** à, sélectionnez **Ajouter un commentaire**, et écrivez cette instruction : **Entre un nombre compris entre -240 et 190**.

En fonction du triangle d'origine, cela permettra de s'assurer que la valeur entrée par la personne produira une translation qui pourra être affichée sur la scène. L'intervalle devra être ajusté en fonction des différents triangles de départ.



7. Finalement, vous devez ajouter un code au programme principal pour activer votre nouveau sous-programme et dessiner un nouveau triangle dont les sommets ont été déplacés horizontalement de 100 unités par rapport aux sommets d'origine.

Modifiez le code en ajoutant les 4 blocs suivants à la fin du programme principal.

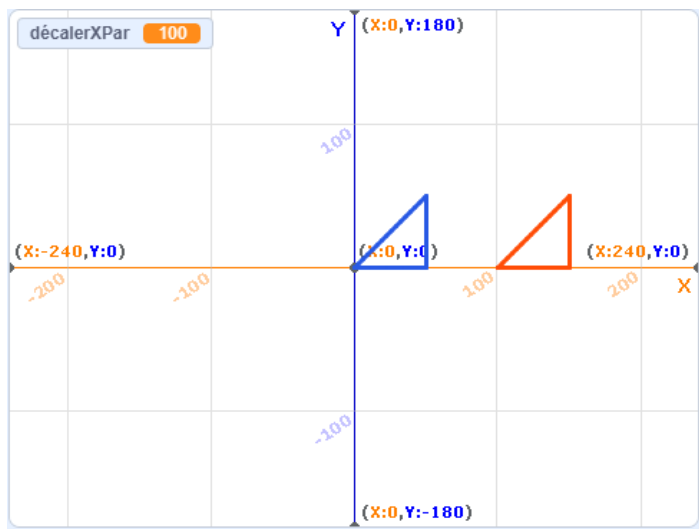


Utiliser le code pour transformer des triangles (suite)

Voici ce que font ces quatre blocs de code :

- Après que le programme principal a dessiné le triangle bleu, le sous-programme **décalerX** est activé pour décaler les coordonnées x des sommets de 100 unités.
- Ensuite, le stylo est relevé pour éviter de dessiner une ligne pendant que l'image-objet se déplace vers sa nouvelle position de départ.
- Ensuite, la couleur du stylo est changée en orange afin que le triangle qui a subi une translation puisse être distingué du triangle d'origine.
- Finalement, le sous-programme **dessinerTriangle** est activé une fois de plus !

8. Cliquez sur le drapeau vert pour faire un test de la nouvelle application. Lorsque le programme a été exécuté, la scène devrait ressembler à ceci. Décrivez le déplacement du triangle.



9. Modifiez le code pour que le triangle original soit déplacé de 100 unités vers la gauche au lieu de la droite. Quelle modification avez-vous apportée ?

Utiliser le code pour transformer des triangles (suite)

Partie 2 : Translation verticale du triangle

Vous écrirez ensuite le code d'un sous-programme qui déplace les 3 sommets du triangle vers le haut ou vers le bas le long de l'axe des y. Voici quelques conseils :

- Commencez par votre version de l'application de l'étape 8 qui inclut le sous-programme **décalerX**.
- Écrivez un code semblable à celui de votre sous-programme **décalerX**, mais appelez-le **décalerY**.
- Vous aurez besoin d'une nouvelle variable, **décalerYPar**. Commencez par une valeur de translation de 100 unités.
- Une fois que vous avez écrit votre sous-programme, ajoutez le commentaire suivant à votre bloc **mettre** pour la variable **décalerYPar** :

Entre un nombre compris entre -180 et 130.

Le programme principal pourrait ressembler à ceci. Remarquez qu'il réinitialise les points initiaux avant de procéder à la translation verticale en activant le sous-programme **pointsInitiaux**. Cela signifie que le triangle sera décalé verticalement par rapport au triangle d'origine et non par rapport au triangle que vous avez fait subir une translation horizontale. Il est possible de faire une translation du deuxième triangle à la place.

C'est à vous de décider en tant que programmeur !

Si vous choisissez de le faire, il manquera à votre programme le bloc qui active le sous-programme **pointsInitiaux**. Sinon, il devrait ressembler à ce qui est montré ici.

Si vous pensez que votre sous-programme **décalerY** ne fonctionne pas correctement ou si vous êtes bloqué, consultez votre enseignant pour trouver une solution.



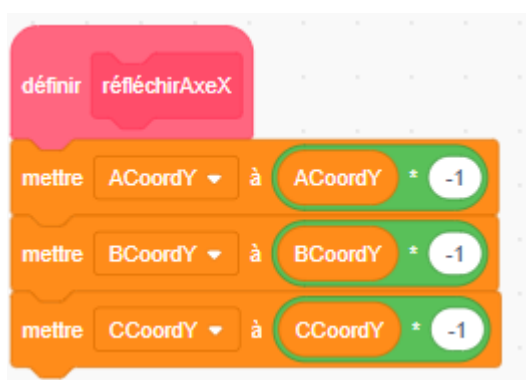
Utiliser le code pour transformer des triangles (suite)

Défis additionnels

- Écrivez un code pour un sous-programme qui produit une réflexion des points du triangle dans l'axe des x . Appelez le sous-programme **réfléchirAxeX**.

Conseil : Vous aurez besoin d'un opérateur de multiplication.

Pourquoi le sous-programme présenté ici fonctionne-t-il ?



- Écrivez un code pour un sous-programme qui produit une réflexion des points du triangle dans l'axe des y . Appelez le sous-programme **réfléchirAxeY**.

Réfléchissez aux similarités et aux différences avec **réfléchirAxeX**.

- Invitez l'utilisateur à entrer des valeurs pour le décalage de la translation horizontale.

Conseil : Utilisez le bloc **Demander et attendre** dans **Capteurs**.