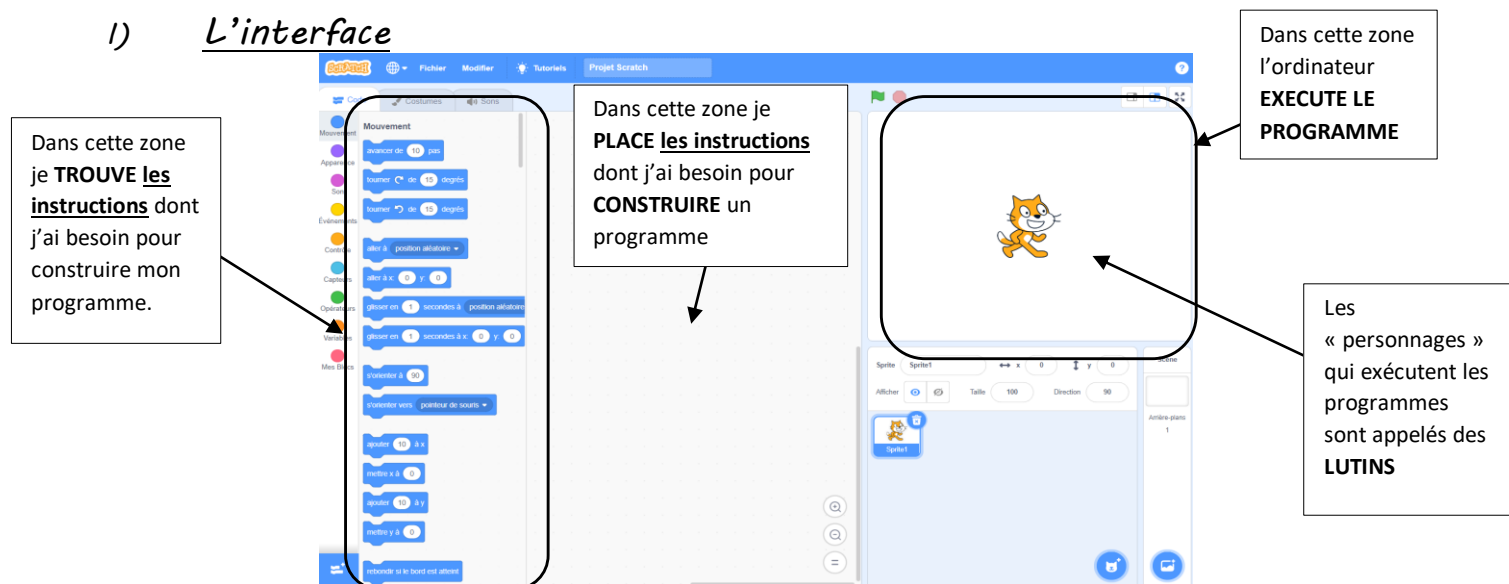
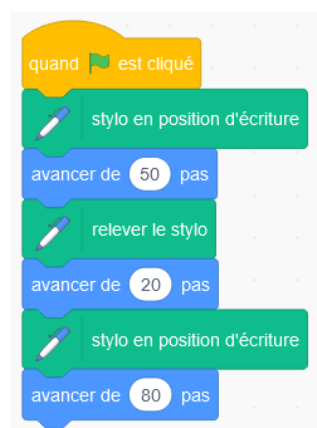


I) L'interface



Avant de commencer : Cliquez en bas à gauche sur puis sur Stylo

II) Avancer/Tracer



1) **Cherche** les instructions pour construire le programme ci-contre puis exécute-le.

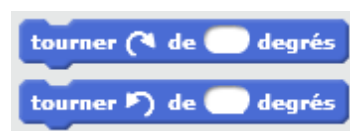
2) **Réponds** aux questions ci-dessous :

- Où dois-tu appuyer pour exécuter/lancer le programme ?
- De combien de pas avance le lutin en tout ?
- De combien de pas avance le lutin en traçant ?
- De combien de pas avance le lutin sans tracer ?
- Dans quelle direction part le lutin au début ?

○ Appelle le professeur pour qu'il vérifie. Une fois fait tu peux supprimer.

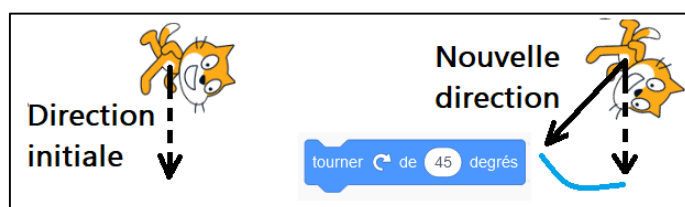
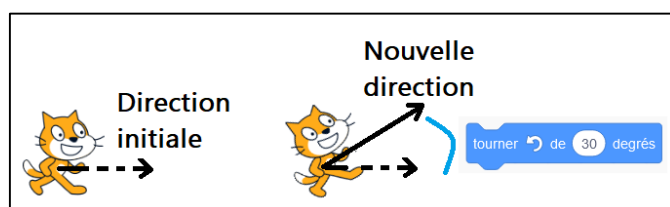
III) Tourner

Pour que le lutin tourne on utilise les instructions :



Attention : Le lutin tourne par rapport à sa direction initiale (« là où il aurait dû aller »).

Exemples :



1) Dans chaque cas **complète** l'instruction correspondant au mouvement que le lutin a réalisé.

Direction initiale
Nouvelle direction
47°

tourner de degrés

Nouvelle direction
Direction initiale
52°

tourner de degrés

Direction initiale
Nouvelle direction
161°

tourner de degrés

Direction initiale
Nouvelle direction
120°

tourner de degrés

Rappel

$\widehat{AOB} = 180^\circ$

Angle plat

Direction initiale
Nouvelle direction
95°

tourner de degrés

Nouvelle direction
Direction initiale
60°

tourner de degrés

Direction initiale
Nouvelle direction
82°

tourner de degrés

○ Appelle le professeur pour qu'il vérifie.

2) **Complète** le programme pour obtenir la figure ci-dessous.

quand est cliqué

s'orienter à 90

aller à x: -60 y: 0

stylo en position d'écriture

effacer tout

avancer de pas

tourner de degrés

avancer de pas

tourner de degrés

avancer de pas

Cette partie la permet de remettre le lutin dans le bon sens (il regarde à droite) et d'effacer les anciens tracés. On le travaillera une prochaine fois 😊

3) **Construis** le programme sur l'ordinateur et **exécute-le** pour vérifier.

○ Appelle le professeur si tu n'arrives pas à obtenir le triangle équilatéral ou pour qu'il vérifie avant de supprimer

4) **Crée** un programme permettant de tracer chaque figure :

○ Vérification

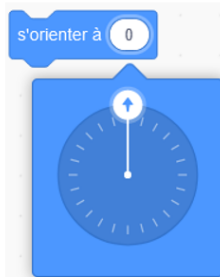
○ Vérification

○ Vérification

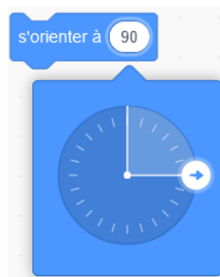
S'orienter - Boucle - Bloc

I) S'orienter

Si on n'indique rien dans un programme au lutin il partira toujours vers la droite au début (par défaut).
Pour être sûr que le lutin parte dans la bonne direction au début il faut l'**orienter** correctement.



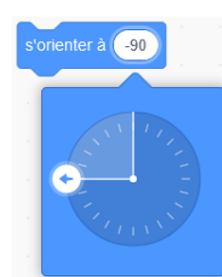
S'orienter vers le haut
0°



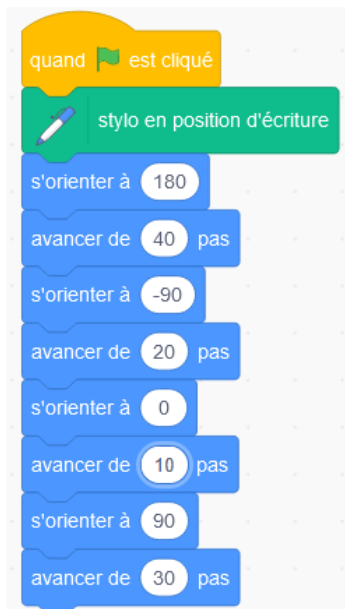
S'orienter vers la droite
90°



S'orienter vers le bas
180°



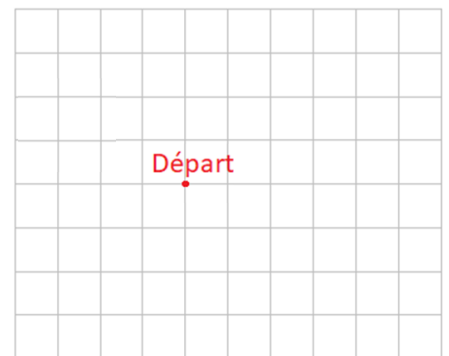
S'orienter vers la gauche
- 90°



SANS T'AIDER DE L'ORDINATEUR, **trace** le chemin obtenu lorsqu'on exécute le programme ci-contre.

Un côté de carré correspond à 10 pas.

○ Appelle le professeur pour qu'il vérifie.



II) Boucle



1) **Complète** le programme de droite pour qu'il **trace le même chemin** que celui de gauche.

Pour t'aider **entoure** sur le programme de gauche la partie répétée plusieurs fois.

Cette partie est répétée fois.

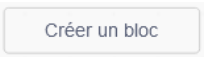
2) **Construis** et **exécute** le programme de droite.

La figure obtenue est un régulier.

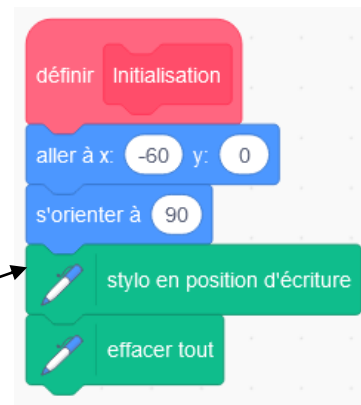
○ Appelle le professeur pour qu'il vérifie puis supprime.

III) Bloc

Dans Scratch les blocs sont des sous-programmes (script) qui peuvent être réutilisés.

- 1) Dans Mes Blocs **clique** sur  et nomme le bloc « Initialisation »
- Construis** le bloc Initialisation comme ci-contre.

Cette instruction permet de s'orienter vers

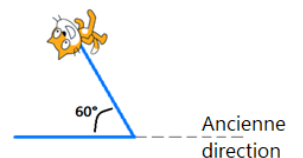


- 2) **Crée** un bloc nommé « Carré » et un bloc nommé « Triangle ».

Complète les deux blocs ci-dessous afin de tracer les figures correspondantes:

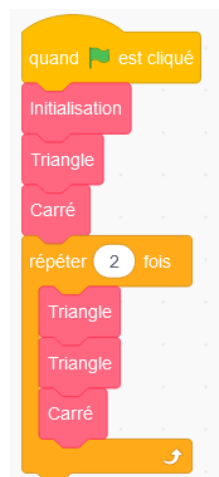
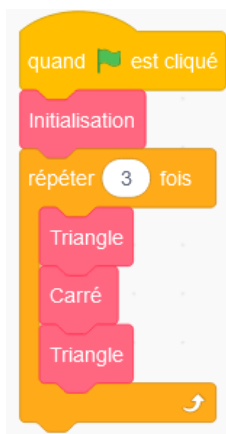
Carré de côté 30 pas	Triangle équilatéral de côté 30 pas

Aide : Fiche SCRATCH n°1



- Appelle le professeur pour qu'il vérifie.

- 3) SANS T'AIDER DE L'ORDINATEUR **relie** chaque tracé au programme correspondant



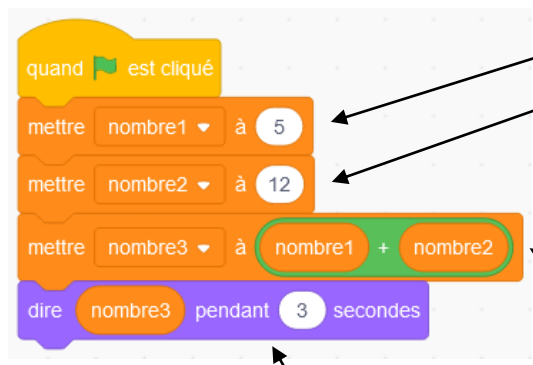
Variable - Opérateur - Instruction conditionnelle (Si Alors Sinon)

1) Variables et opérateurs

Variable : En programmation les variables sont des « boîtes » dans laquelle on peut mettre **une seule information** (un mot ou un nombre). On peut remplacer l'information mais dans ce cas l'ordinateur oublie l'information précédente.

1) Dans « Variable » clique sur  et crée un variable et nomme-la nombre1.
PUIS Crée deux autres variables que tu appelleras nombre2 et nombre3.

2) **OBSERVE** le programme ci-dessous et **LIS** les explications

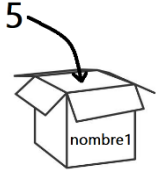


L'ordinateur met dans la variable nombre1 la valeur 5.

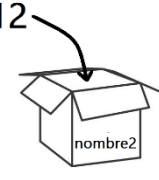
L'ordinateur met dans la variable nombre1 la valeur 12.

L'ordinateur met dans la variable nombre3 le résultat de l'opération

5



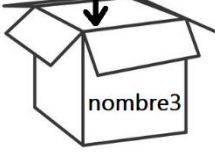
12



nombre1 + nombre2



↓



17



Le lutin dit ce qu'il y a dans la boîte nombre3 pendant 3 secondes ici.

3) **Observe** le programme ci-dessous et **complète** :



a. Le calcul en ligne 4 effectué par l'ordinateur est

b. Dans la variable nombre3 il y a au final la valeur

c. Le lutin va dire la valeur pendant secondes.

L'opérateur  correspond à une multiplication

Remarque importante : Scratch n'utilise pas de parenthèse. Lorsqu'il y a plusieurs opérateurs, l'opération prioritaire est celle qui est le plus à l'intérieur

nombre1 + nombre1 * nombre2

Ici l'opération prioritaire est

nombre1 + nombre1 * nombre2

Ici l'opération prioritaire est

○ Appelle le professeur pour qu'il vérifie puis supprime le programme.

III) Instruction conditionnelle (Si Alors Sinon)

1) **Construis** le programme ci-dessous sur l'ordinateur, **exécute-le** puis **réponds** aux questions :



a. Dans quelle variable se situe ma réponse à « choisis un nombre entre 1 et 10 » ?

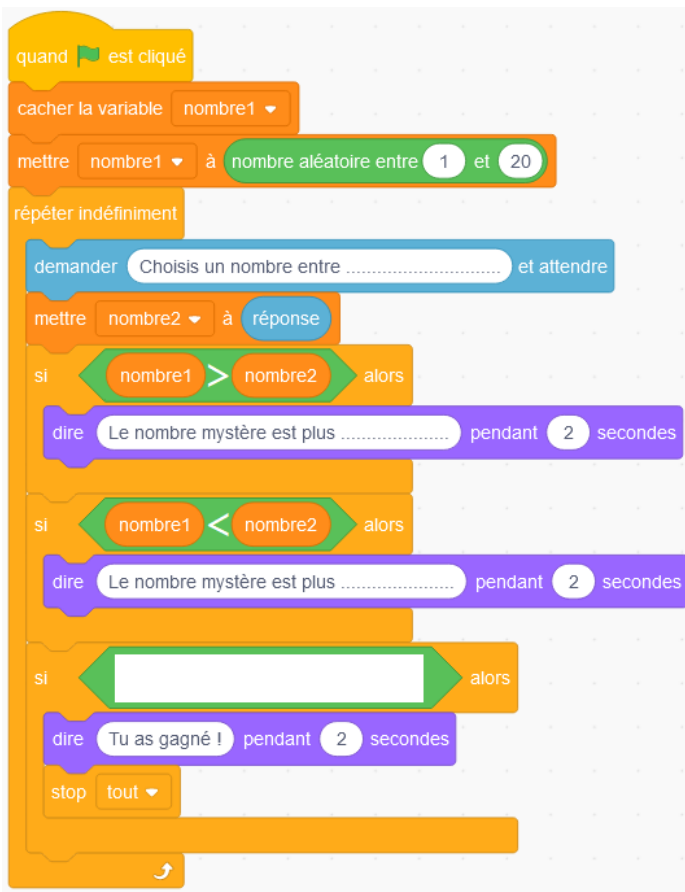
b. Qui choisit le nombre qui se situe dans la variable nombre1 ?

c. Que se passe-t-il si le nombre que j'ai choisi est égal au nombre mystère ?

d. En jouant à ce jeu tu as 1 chance sur de gagner.

○ Appelle le professeur pour qu'il vérifie.

2) **Complète** le programme ci-dessous afin qu'il aide le joueur (toi) à trouver le nombre mystère.



3) **Complète** :

a. Ce programme utilise instructions conditionnelles différentes.

b. Le  sert à

4) **Construis** le programme puis exécute-le.

5) Que se passe-t-il si on enlève le  ?

○ Appelle le professeur pour qu'il vérifie.

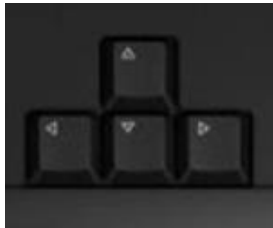
6) **Modifie** le programme pour que le nombre mystère soit entre 1 et 100.

7) **Modifie** le programme pour que le lutin dise à la fin en combien de tour de jeu le joueur à trouver le nombre mystère. ○ Appelle le professeur pour qu'il vérifie.

Interraction entre lutins - Révisions

Dans cette partie nous allons construire un programme constitué de plusieurs scripts et de plusieurs lutins. Il ne faudra RIEN supprimer sans l'autorisation du professeur.

1) Attraper la pomme



1) **Construis** le script ci-contre puis TROIS AUTRES script (un pour chaque flèche restante).



2) **Teste** les 4 flèches du clavier pour vérifier.

(Exemple : si tu appuies sur flèche du haut il faut que le lutin aille vers le haut)

○ Appelle le professeur pour qu'il vérifie MAIS ne supprime pas.



3) **Ajoute** un nouveau lutin à ton programme. **Choisis** le lutin nommé Apple.



Modifie sa taille pour qu'elle soit de 30 pixels.

4) **Construis** le programme ci-dessous dans la zone dédiée aux scripts de la pomme.



A quoi sert ce programme ? Tu peux l'exécuter pour t'aider.

.....

.....

.....

.....

.....

.....

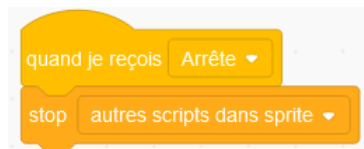
○ Appelle le professeur pour qu'il vérifie MAIS ne supprime pas.

6) a. **Clique** sur le sprite1 dans le choix des lutins.

b. A l'aide des instructions ci-dessous, **construis** un autre script qui permet au sprite1 (le chat) de dire « J'ai gagné ! » lorsque le lutin Apple est touché et qui permet également d'envoyer à la pomme le message « arrête ».



7) **Retourne** dans la partie de la pomme et **construis** ce dernier script.



A quoi sert-il ?

.....

.....

○ Lance le jeu pour le tester puis appelle le professeur pour qu'il vérifie.

8) **Joue** quelques parties. Ce jeu te paraît-il facile à jouer ?

9) a. Que pourrait-on faire pour compliquer le jeu ?

b. **Essaye** de **modifier** un ou plusieurs scripts pour rendre le jeu plus difficile.

○ Apelle le professeur pour qu'il vérifie ou si tu n'y arrives pas (après avoir réellement essayé)

II) Lancer de dés

Voici le déroulement du jeu qu'il va falloir simuler :

Le joueur choisit un nombre entre 2 et 12. On lance deux dés et on calcule la somme des nombres obtenus. Si la somme est égale au nombre du joueur alors il a gagné, sinon il a perdu.

1) **Explique** pourquoi le joueur doit choisir un nombre entre 2 et 12.

2) **Crée et construis** un programme permettant de simuler ce jeu.

Aide 1 : Commence par faire demander à l'utilisateur un nombre.

Aide 2 : Crée deux variables (boîtes). Une pour chaque dé.

Aide 3 : nombre aléatoire entre et

Aide 4 : Utilise une instruction conditionnelle (Si, Alors, Sinon)

○ Apelle le professeur pour qu'il vérifie ou si tu n'y arrives pas (après avoir réellement essayé)

3) **Modifie** le programme pour que le joueur joue 10 parties et que l'ordinateur compte le nombre de fois qu'il a gagné.

Aide 1 : Crée une variable « compteur »

Aide 2 : ajouter 1 à Compteur

○ Apelle le professeur pour qu'il vérifie ou si tu n'y arrives pas (après avoir réellement essayé)

4) Quel nombre le joueur doit-il choisir pour avoir le plus de chance de gagner ? **Justifie.**

Aide : **Remplis** ce tableau de somme

+	1	2	3	4	5	6
1						
2						
3						
4						
5						
6						

.....

.....

.....

.....

.....

.....

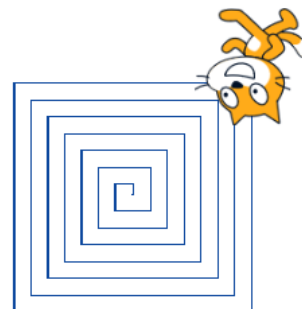
I) La spirale colorée

1) En t'aidant de la fiche Scratch n°3 **construis** un bloc Initialisation qui :

- Place le lutin au centre de la zone (coordonnées : (0 ; 0))
- Oriente le lutin vers le haut au départ

2) **Construis** un programme qui :

- S'exécute/Se lance lorsque l'utilisateur appuie sur le touche espace du clavier.
- Utilise le bloc initialisation.
- Répète 200 fois la même chose : Avancer puis tourner à gauche en angle droit.
- Trace un premier trait d'une longueur de 1 pas.
- A chaque répétition augmente la longueur du trait de 1 pas.

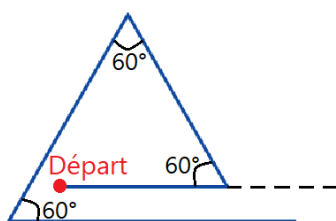


Aide : Créer une variable nommée longueur.

- Appelle le professeur pour qu'il vérifie mais ne supprime pas le script ensuite.

3) **Détermine** approximativement combien de répétition au maximum le lutin peut effectuer avant de « toucher les bords » de sa zone ?

4) **Modifie** le programme pour qu'à chaque répétition il change la couleur du stylo à l'aide de l'instruction ci-contre puis **exécute** le programme.



5) **Modifie** le programme pour qu'il trace une spirale en forme triangulaire.

- Appelle le professeur pour qu'il vérifie.

II) Existence d'un triangle

1) A l'aide du chapitre sur les triangles répond **aux questions** en justifiant :

a. Le triangle de côtés 4cm, 7cm et 8cm existe-t-il ?

.....

.....

.....

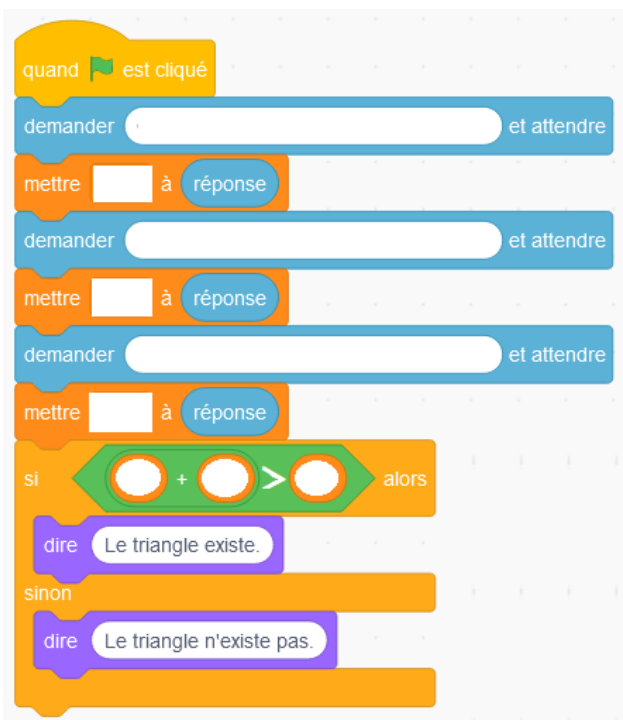
.....

.....

b. Le triangle 3cm, 5cm et 9cm existe-t-il ?

○ Appelle le professeur pour qu'il vérifie.

2) a. **Complète** le programme ci-dessous afin que l'utilisateur rentre les trois longueurs et que le lutin lui dise si le triangle existe ou non.



Aide 1 : Il y a trois longueurs donc il te faut créer trois variables (boîtes) pour mettre les valeurs à l'intérieur. (Tu peux les nommer : a, b et c)

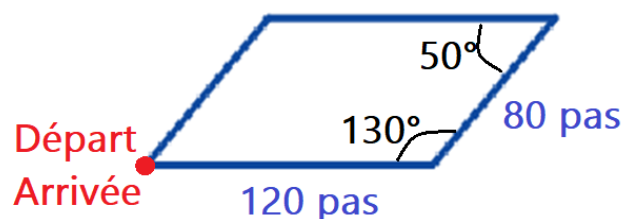
Aide 2 : Pense bien que les longueurs ne doivent pas être données dans n'importe quel ordre.

b. **Teste** ton programme en utilisant les valeurs de la question 1 pour déterminer s'il fonctionne.

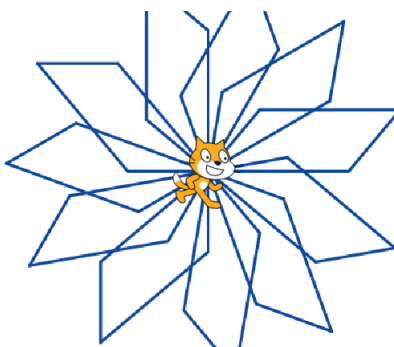
○ Appelle le professeur pour qu'il vérifie.

III) Tracer une rosace

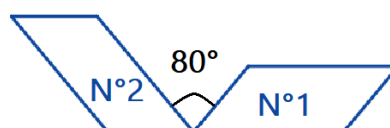
1) **Crée** un bloc nommé « Parallélogramme » qui permet de tracer le parallélogramme ci-contre. (Attention aux angles !)



2) **Crée** et construis un programme utilisant les blocs Initialisation et Parallélogramme permettant de tracer la rosace ci-dessous :



Aide 1 :



Aide 2 : Compte le nombre de parallélogramme pour savoir le nombre de répétition.

○ Appelle le professeur pour qu'il vérifie.