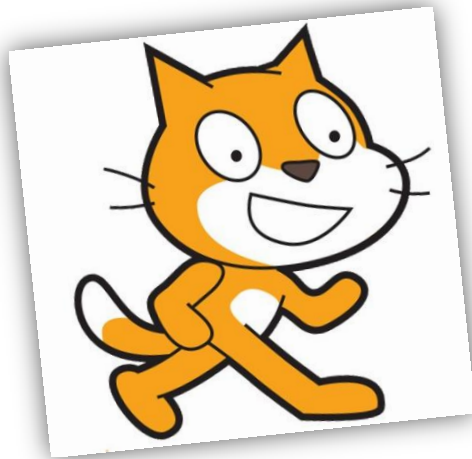


Enseigner
la programmation et l'algorithmique
au COLLÈGE avec

SCRATCH



LIVRET N°2

(Exemples d'activités pédagogiques)

2015-2016

Rappel des textes (extraits des **nouveaux programmes** de mathématiques)

Comment faire vivre la programmation en cours de mathématiques ?

Concepts de programmation (boucle, ...)

Exemples d'activités **mathématiques**

Exemples d'activités "**ludiques**" (programmation de jeux)

Il est conseillé de faire réaliser au moins deux projets de jeux par les élèves.

Exemples d'activités **transversales** (dans le cadre des EPI et/ou de l'AP)

Rappel des textes

BO N°11 du 26 novembre 2015

AU CYCLE 4

BO n°11 du 26 novembre 2015, ESPACE et GEOMETRIE

Dans la continuité du cycle 3, les élèves se familiarisent avec les fonctionnalités d'un logiciel de géométrie dynamique ou de programmation pour construire des figures.

BO n°11 du 26 novembre 2015, ALGORITHMIQUE ET PROGRAMMATION

Thème E - Algorithmique et programmation

Au cycle 4, les élèves s'initient à la programmation, en développant dans une démarche de projet quelques programmes simples, sans viser une connaissance experte et exhaustive d'un langage ou d'un logiciel particulier. En créant un programme, ils développent des méthodes de programmation, revisitent les notions de variables et de fonctions sous une forme différente, et s'entraînent au raisonnement.

Attendus de fin de cycle

Écrire, mettre au point et exécuter un programme simple

Connaissances et compétences associées	Exemples de situations, d'activités et de ressources pour l'élève
Décomposer un problème en sous-problèmes afin de structurer un programme ; reconnaître des schémas. Écrire, mettre au point (tester, corriger) et exécuter un programme en réponse à un problème donné. Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs. Programmer des scripts se déroulant en parallèle. » Notions d'algorithme et de programme. » Notion de variable informatique. » Déclenchement d'une action par un événement, séquences d'instructions, boucles, instructions conditionnelles.	Jeux dans un labyrinthe, jeu de Pong, bataille navale, jeu de nim, tic tac toe. Réalisation de figure à l'aide d'un logiciel de programmation pour consolider les notions de longueur et d'angle. Initiation au chiffrement (Morse, chiffre de César, code ASCII...). Construction de tables de conjugaison, de pluriels, jeu du cadavre exquis... Calculs simples de calendrier. Calculs de répertoire (recherche, recherche inversée...). Calculs de fréquences d'apparition de chaque lettre dans un texte pour distinguer sa langue d'origine : français, anglais, italien, etc.

Repères de progressivité

En 5^e, les élèves s'initient à la programmation événementielle. Progressivement, ils développent de nouvelles compétences, en programmant des actions en parallèle, en utilisant la notion de variable informatique, en découvrant les boucles et les instructions conditionnelles qui complètent les structures de contrôle liées aux événements.

AU CYCLE 3

BO n°11 du 26 novembre 2015, Volet 2 : Contributions essentielles des différents enseignements au socle commun

La maîtrise des techniques et la connaissance des règles des outils numériques se construisent notamment à travers l'enseignement des sciences et de la technologie où les élèves apprennent à connaître l'organisation d'un environnement numérique et à utiliser différents périphériques ainsi que des logiciels de traitement de données numériques (images, textes, sons...). En mathématiques, ils apprennent à utiliser des logiciels de calculs et d'initiation à la programmation.

BO n°11 du 26 novembre 2015, préambule du programme de mathématiques

En mathématiques, ils apprennent à utiliser des logiciels de calculs et d'initiation à la programmation.

BO n°11 du 26 novembre 2015, ESPACE ET GEOMETRIE

Les activités spatiales et géométriques sont à mettre en lien avec les deux autres thèmes : résoudre dans un autre cadre des problèmes relevant de la proportionnalité ; utiliser en situation les grandeurs (géométriques) et leur mesure. Par ailleurs, elles constituent des moments privilégiés pour une première initiation à la programmation notamment à travers la programmation de déplacements ou de construction de figures.

Connaissances et compétences associées	Exemples de situations, d'activités et de ressources pour l'élève
(Se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations	
Se repérer, décrire ou exécuter des déplacements, sur un plan ou sur une carte. Accomplir, décrire, coder des déplacements dans des espaces familiers. Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran. » Vocabulaire permettant de définir des positions et des déplacements. » Divers modes de représentation de l'espace.	Situations donnant lieu à des repérages dans l'espace ou à la description, au codage ou au décodage de déplacements. Travailler : » dans des espaces de travail de tailles différentes (la feuille de papier, la cour de récréation, le quartier, la ville, etc.) ; » à partir de plans schématiques (par exemple, chercher l'itinéraire le plus court ou demandant le moins de correspondances sur un plan de métro ou d'autobus) ; » avec de nouvelles ressources comme les systèmes d'information géographique, des logiciels d'initiation à la programmation...

Reconnaitre et utiliser quelques relations géométriques

Effectuer des tracés correspondant à des relations de perpendicularité ou de parallélisme de droites et de segments.

Déterminer le plus court chemin entre deux points (en lien avec la notion d'alignement).

Déterminer le plus court chemin entre un point et une droite ou entre deux droites parallèles (en lien avec la perpendicularité).

- » Alignement, appartenance.
- » Perpendicularité, parallélisme (construction de droites parallèles, lien avec la propriété reliant droites parallèles et perpendiculaires).
- » Egalité de longueurs.
- » Egalité d'angles.
- » Distance entre deux points, entre un point et une droite.

Situations conduisant les élèves à utiliser des techniques qui évoluent en fonction des supports et des instruments choisis ; par exemple pour la symétrie axiale, passer du pliage ou de l'utilisation de papier calque à la construction du symétrique d'un point par rapport à une droite à l'équerre ou au compas.

Exemples d'instruments : règle graduée, équerre, compas, gabarits d'angles, bandes de papier, papier calque.

Exemples de supports variés : géoplans, papier quadrillé, papier pointé, papier uni.

Exemples de matériels : papier/crayon, logiciels de géométrie dynamique, d'initiation à la programmation, logiciels de visualisation de cartes, de plans.

Initiation à la programmation : Une initiation à la programmation est faite à l'occasion notamment d'activités de repérage ou de déplacement (programmer les déplacements d'un robot ou ceux d'un personnage sur un écran), ou d'activités géométriques (construction de figures simples ou de figures composées de figures simples). Au CM1, on réserve l'usage de logiciels de géométrie dynamique à des fins d'apprentissage manipulatoires (à travers la visualisation de constructions instrumentées) et de validation des constructions de figures planes. À partir du CM2, leur usage progressif pour effectuer des constructions, familiarise les élèves avec les représentations en perspective cavalière et avec la notion de conservation des propriétés lors de certaines transformations.

AU CYCLE 2

BO n°11 du 26 novembre 2015, programme du CYCLE 2 (ESPACE et GEOMETRIE)





Dès le CE1, les élèves peuvent coder des déplacements à l'aide d'un logiciel de programmation adapté, ce qui les amènera au CE2 à la compréhension, et la production d'algorithmes simples.

Comment faire vivre la programmation en cours de mathématiques ?

Pour l'attendu de fin de cycle « Ecrire, mettre au point et exécuter un programme simple » :

- ⇒ avec de **petits exercices** sur *des programmes de calcul*, de *construction géométrique*, ou liés à d'autres thèmes, courts, souvent faciles, et **réguliers**
- ⇒ avec **au moins un ou deux projets de programmation**, par exemple d'un jeu (jeu de Pong; jeu de labyrinthe ...)
- ⇒ avec l'interface Scratch, très intuitive, très puissante mais facile

Les compétences qui peuvent être travaillées et évaluées à travers l'algorithmique et la programmation

Algorithmique et programmation				
Je suis capable de reproduire un programme à l'aide d'un modèle	RR	R	V	VV
Je comprends et j'analyse le fonctionnement d'un programme	RR	R	V	VV
Je suis capable de tester et d'exécuter un programme simple	RR	R	V	VV
Je suis capable d'écrire un programme simple	RR	R	V	VV
Je suis capable de modifier un programme pour l'améliorer	RR	R	V	VV
Je suis capable de créer un programme en réponse à un problème donné	RR	R	V	VV
Je suis capable de mener jusqu'au bout un projet de programmation (jeu, histoire interactive ...)	RR	R	V	VV

RR: *compétence non maîtrisée*

R: *compétence à consolider*

V: *compétence acquise*

VV: *compétence dominée*

Concepts de programmation et compétences développées avec SCRATCH

CONCEPTS DE PROGRAMMATION ET COMPÉTENCES DÉVELOPPÉS AVEC SCRATCH

Dans le processus de création d'histoires interactives, de jeux, et d'animations avec Scratch, les jeunes peuvent acquérir des compétences et apprendre d'importants concepts informatiques.

CAPACITÉS À LA RÉOLUTION DE PROBLÈMES ET À LA CONCEPTION DE PROJETS

- raisonner logiquement
- rechercher des bugs
- développer des idées depuis la conception initiale jusqu'à la réalisation de projet complet
- favoriser la capacité de concentration et la persévérance

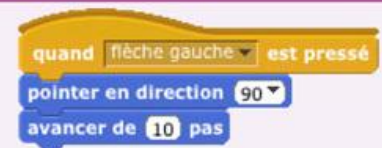


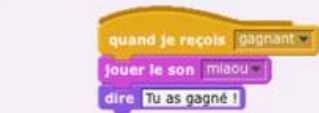
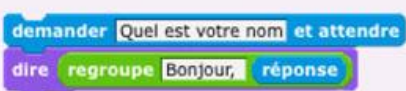




IDÉES FONDAMENTALES SUR LES ORDINATEURS ET LA PROGRAMMATION

- Écrire un programme informatique impose de dire à l'ordinateur ce qu'il doit faire de façon précise et ceci pas à pas
- Écrire un programme informatique ne nécessite pas d'être un expert mais d'avoir une pensée claire et attentive

CONCEPTS SPÉCIFIQUES DE PROGRAMMATION

Concept	Explication	Exemple
séquence	Pour créer un programme avec Scratch, vous devez systématiquement penser à l'ordre des étapes.	
itération (boucle)	répéter indéfiniment et répéter peuvent être utilisés pour une itération (répétition d'une série d'instructions)	
instructions conditionnelles	si et si - sinon permettent d'engager une action suivant qu'une condition est réalisée ou non.	
variables	Les blocs variable permettent de créer des variables et de les utiliser dans un programme. Les variables peuvent contenir des nombres ou des chaînes de caractères. Scratch permet d'utiliser aussi bien des variables globales que des variables spécifiques à un objet.	
listes (tableaux)	Les blocs listes permettent de stocker et d'avoir accès à une liste de nombres et de chaînes de caractères. Cette sorte de structure de données peut être considérée comme un tableau dynamique.	

CONCEPTS DE PROGRAMMATION ET COMPÉTENCES DÉVELOPPÉS AVEC SCRATCH

Concept	Explication	Exemple
gestion d'évènements	L'utilisation du bloc quand ... est pressé ou un clic sur un lutin sont des exemples de gestion d'évènements déclenchés par l'utilisateur ou par une autre partie du programme.	
processus (exécution parallèle)	Lancer 2 piles en même temps crée 2 processus indépendants qui s'exécutent en parallèle.	
coordination et synchronisation	envoyer à tous ... et quand je reçois ... peuvent coordonner les actions de plusieurs lutins. En utilisant envoie et attends il est possible de synchroniser plusieurs actions.	Par exemple, Lutin 1 envoie le message "gagnant" quand cette condition est remplie :  Le script du Lutin 2 est activé quand le message "gagnant" est reçu : 
entrée au clavier	demande ... et attends invite l'utilisateur à saisir sa réponse . réponse enregistre l'entrée au clavier après sa validation.	
nombres au hasard	nombre aléatoire entre ... et ... choisit un nombre entier au hasard dans un intervalle donné.	
opérateur logique booléen	et , ou , non sont des exemples d'opérateurs logiques booléens.	
interaction dynamique	Pour une interaction dynamique en temps réel souris x , souris y , et volume peuvent être utilisés.	
conception d'une interface utilisateur	Dans Scratch, vous pouvez concevoir une interface utilisateur : par exemple, en déclenchant une action par un clic sur un lutin.	

CONCEPTS DE PROGRAMMATION NON INTRODUCIS ACTUELLEMENT DANS SCRATCH

- procédures et fonctions
- passage de paramètres et retour de valeurs
- récursion
- définition de classes d'objets
- héritage
- gestion d'exceptions
- fichier d'entrée sortie

Exemples

d'activités

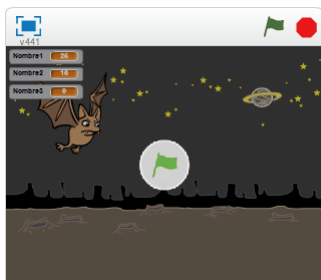
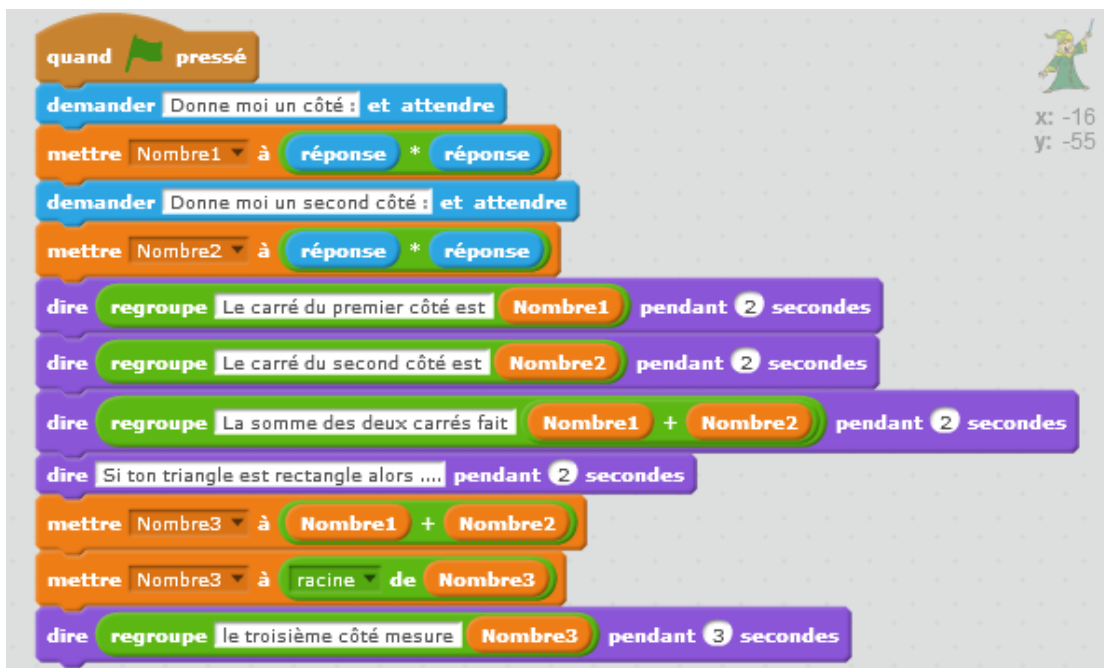
MATHÉMATIQUES

Voici les deux programmes :



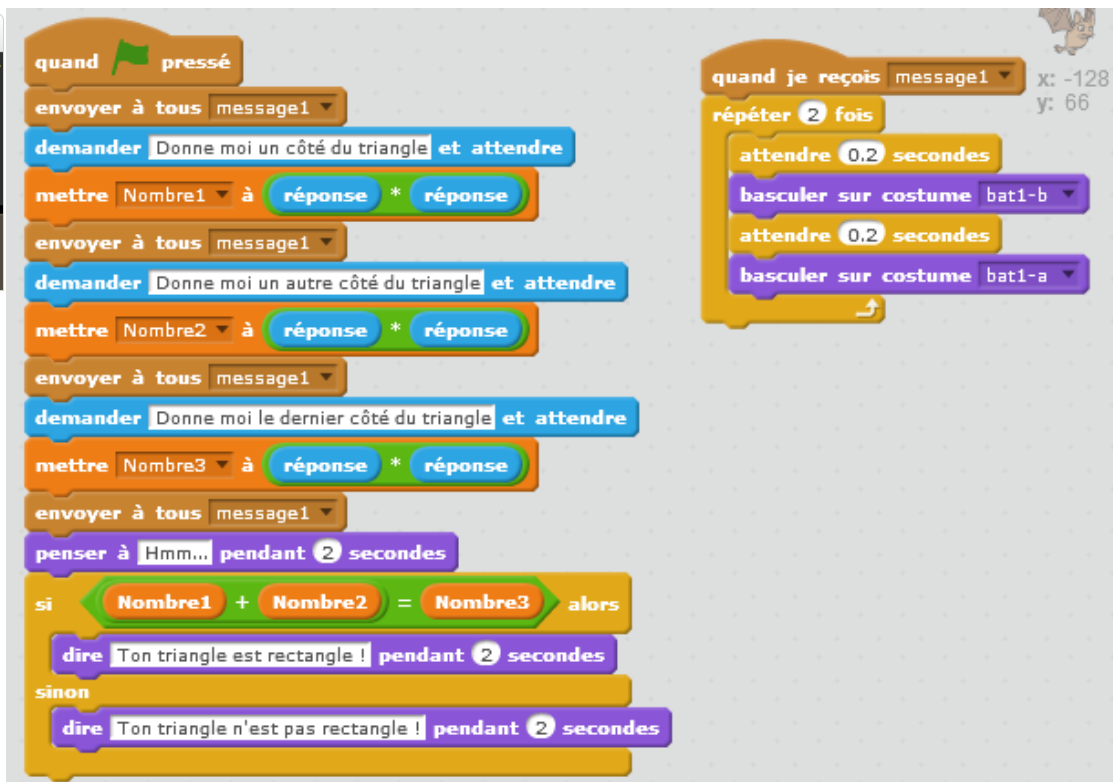
Le lien vers la source du code :

<https://scratch.mit.edu/projects/83887276/#editor>



Le lien vers la source du code :

<https://scratch.mit.edu/projects/83901096/#editor>



Un exemple de fiche élève:

Voici ci-contre un programme *Scratch*.

- ☞ Teste ce code sur un ordinateur.
- ☞ Il y a un problème non ?
- ☞ Améliore ce code.

```
quand pressé
demander "Donne moi un côté : " et attendre
mettre Nombre1 à réponse + réponse
demander "Donne moi un second côté : " et attendre
mettre Nombre2 à réponse + réponse
dire "regroupe Le carré du premier côté est " Nombre1 " pendant 2 secondes"
dire "regroupe Le carré du second côté est " Nombre2 " pendant 2 secondes"
dire "regroupe La somme des deux carrés fait " Nombre1 + Nombre2 " pendant 2 secondes"
dire "Si ton triangle est rectangle alors ..." pendant 2 secondes
mettre Nombre3 à Nombre1 + Nombre2
mettre Nombre3 à racine de Nombre3
dire "regroupe le troisième côté mesure " Nombre3 " pendant 3 secondes"
```

=====

Votre triangle est-il rectangle ?

Voici un programme *Scratch* :

<http://tableauxmaths.fr/spip/spip.php?article144#wrap2>

- ☞ Teste ce code sur un ordinateur.
- ☞ Il y a un problème non ?
- ☞ Améliore ce code.



Partie 1 : Un premier programme

- Décrivez ce que fait le programme.
- Testez le programme pour 5; -4 et 10.
- Que remarquez-vous entre le nombre de départ et le résultat final ?

Partie 2 : A vous de coder !

Voici un programme de calcul :

- ⇒ Choisir un nombre
- ⇒ Multiplier ce nombre par 2
- ⇒ Ajouter 9 au résultat précédent
- ⇒ Soustraire 15
- ⇒ Annoncer le résultat

- A l'aide d'un programme, donner le résultat final pour 3 ; 9 et -6.

Partie 3 : Tour de Magie

Mathématrix est un « vrai » magicien. En effet, si une personne choisit un nombre, puis effectue mentalement certaines étapes de calculs, il est capable de deviner le nombre choisi par cette personne en ne sachant que le résultat final.



- A l'aide d'un programme et du programme de calcul précédent (partie 2), **retrouve le nombre de départ** si le résultat final est 30; 48 et -13.

Partie 4 : Prolongement

Si tu as tout terminé, améliore ton programme en faisant preuve de créativité. Voici quelques idées, change le lutin, bascule l'arrière plan de la scène vers un autre arrière plan quand le magicien donne le résultat ...

Partie 1 : Sur le papier !Principe de l'algorithme d'Euclide :

L'algorithme d'Euclide, consiste à effectuer une suite de divisions euclidiennes :

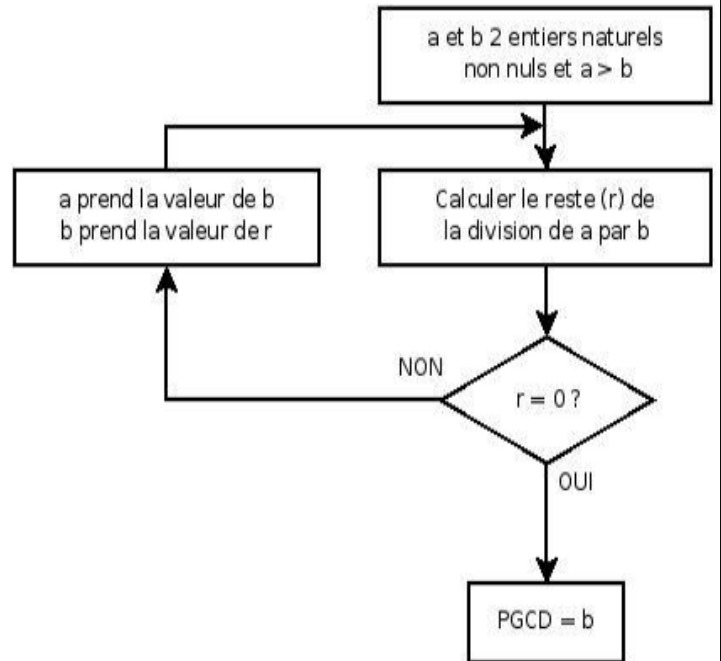
- On effectue la division euclidienne de a par b et on note r le reste.

- Ensuite, b devient a et r devient b ;

_ Et on recommence: on effectue la division euclidienne de a par b et on note r le reste.

- On continue ainsi de suite jusqu'à ce qu'une division donne un reste égal à 0.

Dans cette méthode le PGCD est le dernier reste non nul.




- Par ce même procédé, calculer le PGCD de 406 et 119.

Partie 2 : A vous de coder !

- Créer un programme avec Scratch, permettant de calculer le PGCD de deux nombres en utilisant l'organigramme ci-dessus.



Petite aide : pour calculer le reste dans la division euclidienne de deux nombres sur Scratch, il faut utiliser l'opérateur .

Partie 3 : Prolongement

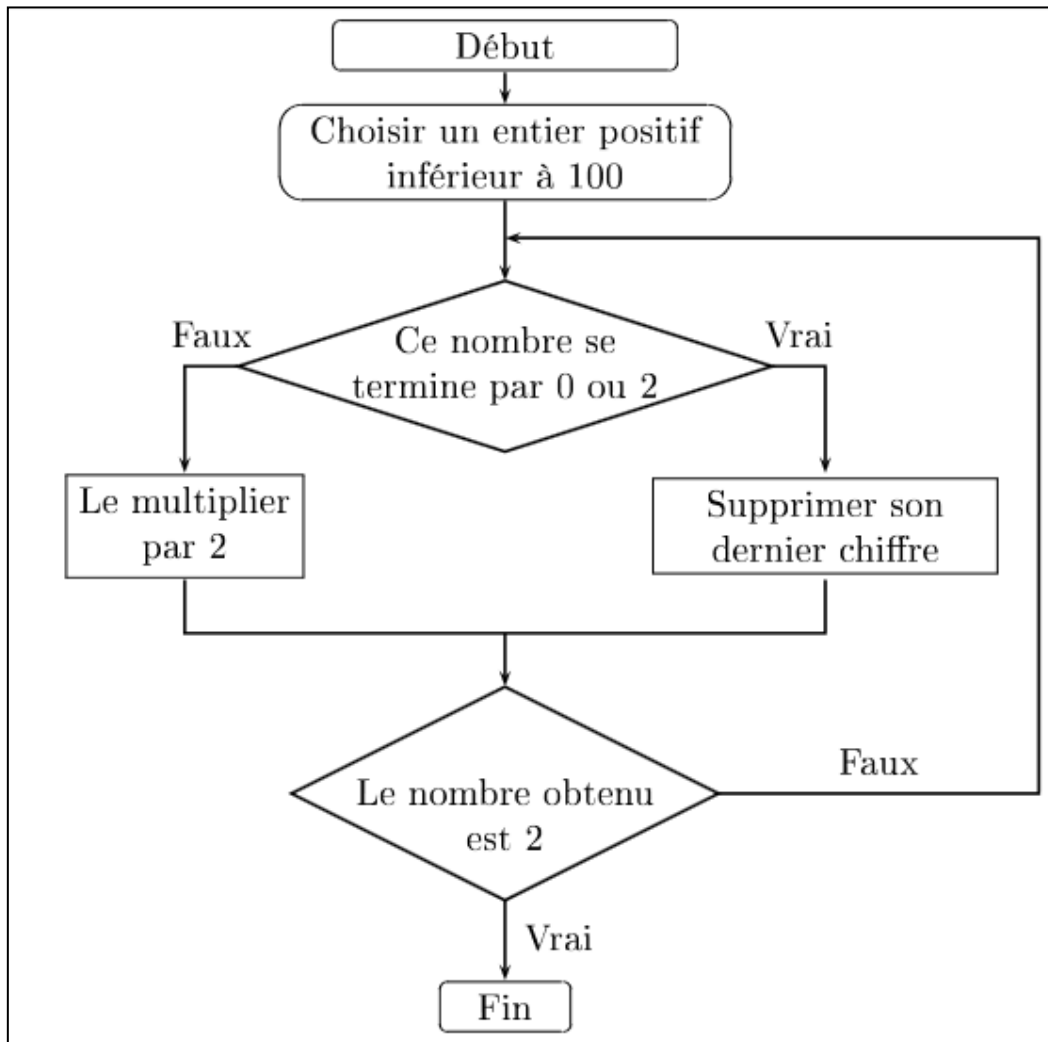
Si tu as fini ton travail et qu'il te reste un peu de temps :

- Tu peux changer l'apparence du lutin.
- Tu peux lui faire faire une pirouette à la fin du programme car il est heureux d'avoir réussi !

Un exemple de corrigé



Voici un organigramme :



A l'aide d'un programme sur Scratch :

- Tester cet algorithme avec 19 et 29.
- L'appliquer à l'entier 43. Que remarque-t-on ?
- Déterminer tous les nombres entiers compris entre 1 et 100 pour lesquels l'algorithme "tourne" indéfiniment.

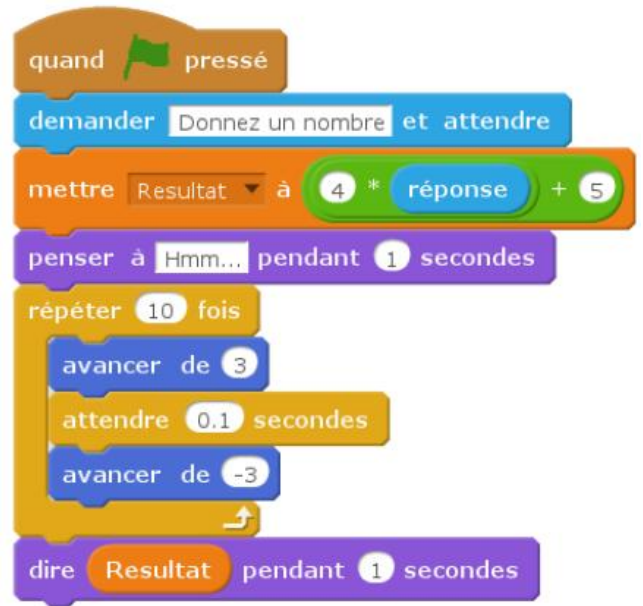
Exemple de corrigé

```
graph TD; Start[quand flag pressé] --> Ask[demander Choisir un entier compris entre 1 et 100 et attendre]; Ask --> SetA[mettre a à réponse]; SetA --> Repeat[ répéter jusqu'à a = 2 ]; Repeat --> If{si a modulo 10 = 0 ou a modulo 10 = 2 alors}; If --> SetFloor[mettre a à plancher de a / 10]; If --> Else[sinon]; Else --> SetDouble[mettre a à a * 2]; SetDouble --> Repeat; If --> Repeat; Repeat --> End[ ];
```

The image shows a Scratch script designed to find the last digit of a number. It starts with a 'when green flag is clicked' event, followed by a 'ask' block: 'Choisir un entier compris entre 1 et 100 et attendre'. The user's input is stored in a variable 'a'. A 'repeat until' loop is set to run until 'a' equals 2. Inside this loop, an 'if' block checks if 'a modulo 10' is equal to 0 or 2. If true, 'a' is set to the integer part of 'a / 10'. If false, 'a' is set to 'a * 2'. The loop repeats until 'a' reaches 2.

Partie 1 : Un premier programme

- Quelle est l'image de 3 par la fonction f ?
- Quelle est l'image de 97 par la fonction f ?
- Quelle est l'expression algébrique de f ?
- Donnez un antécédent de 65 par la fonction f :
- Donnez un antécédent de 100 par la fonction f .

Partie 2 : A vous de coder

Voici un programme de calculs :

- ⇒ Choisir un nombre
- ⇒ L'élever au carré
- ⇒ Multiplier le résultat par 3
- ⇒ Retrancher 16 fois le nombre choisi au résultat
- ⇒ Ajouter 21 à l'ensemble

- Quel est le résultat de ce programme si le nombre choisi est 3 ?
- Utilisez Scratch pour modéliser ce programme.
- Testez votre code en choisissant 3 au départ.
- Quel est le résultat de ce programme si le nombre choisi est 9 ?

Ce programme représente une fonction g .

- Donnez l'expression algébrique de cette fonction.
- Donnez un antécédent de 1365 par la fonction g :

Partie 3 : Une autre fonction

On donne la fonction $h(x) = (x - 3)(3x - 7)$

- A l'aide d'un programme, calculez les images de 3 ; de 9 puis de -6 par la fonction h .
- Comparez les fonctions h et g .

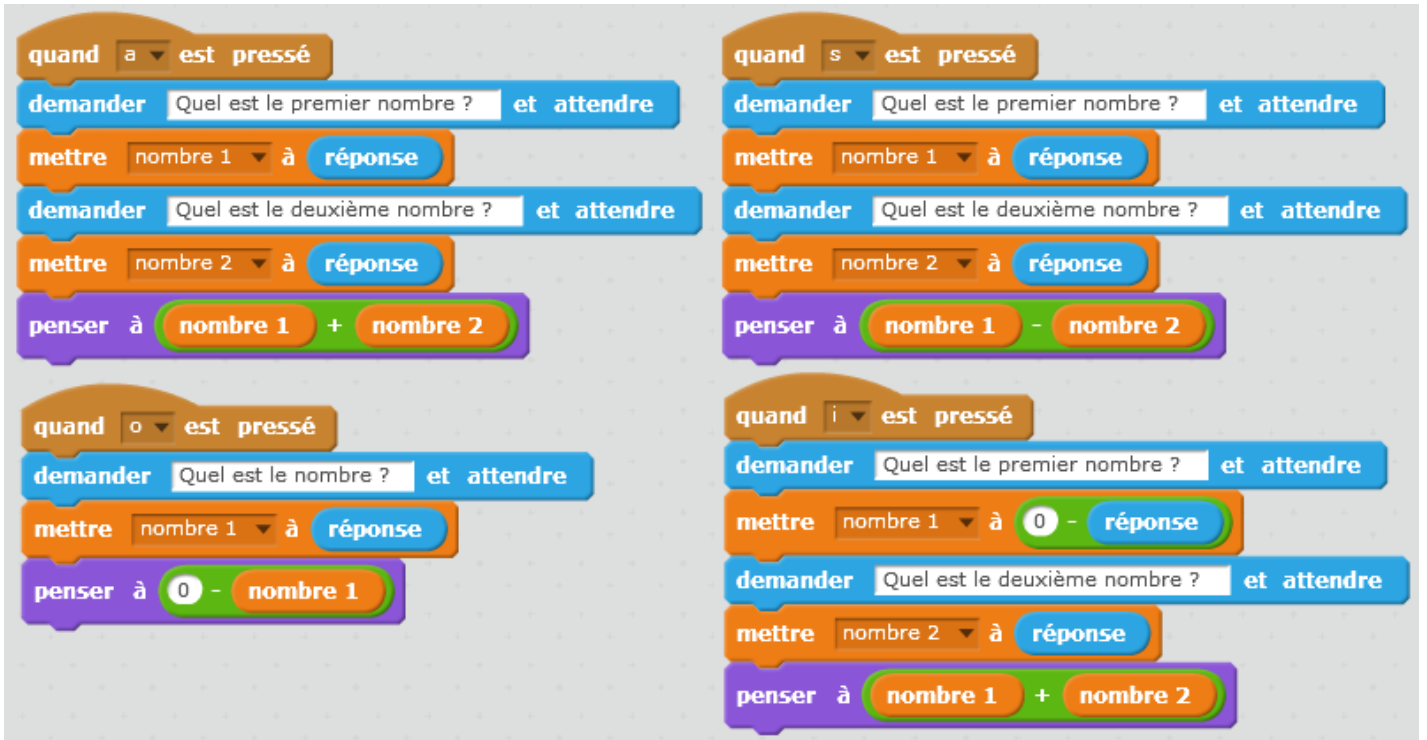
Partie 4 : Un travail pour plus tard

Voici un programme de calculs

- ⇒ Choisir un nombre
- ⇒ Multiplier par 2
- ⇒ Élever le résultat au carré
- ⇒ Retrancher 12 fois le nombre choisi au résultat
- ⇒ Ajouter 20 à l'ensemble

- Utilisez un programme Scratch pour tracer la fonction représentée par ce programme.
Commencez par un tableau de valeurs...
- Donnez une expression algébrique de cette fonction

Voici les éléments d'un programme avec le logiciel Scratch.



1^{ère} partie :

- 1) Décrire ce qu'il se passe lorsque l'utilisateur appuie sur « a ».
- 2) Décrire ce qu'il se passe lorsque l'utilisateur appuie sur « s ».
- 3) Décrire ce qu'il se passe lorsque l'utilisateur appuie sur « o ».
- 4) Décrire ce qu'il se passe lorsque l'utilisateur appuie sur « i ».

2^{ème} partie :

Image 1



Pour chacune des deux images ci-contre, proposer des nombres qui conviennent pour que le chat pense à cette réponse lorsque l'utilisateur appuie

- 1/ sur « a »
- 2/ sur « s »
- 3/ sur « o »
- 4/ sur « i »

Image 2



3ème partie :



On sait dans cette partie que le premier nombre est 12.

1/ Sur quelles touches l'utilisateur a-t-il pu appuyer ?

2/ Pour chacune des touches possibles, quel doit être le deuxième nombre si le chat pense à la réponse

- a) 5
- b) -8
- c) 0

L'objectif de cette activité est de tracer des figures géométriques avec l'outil Stylo de Scratch.

A) Suivre un programme :



Que se passe-t-il ?

.....

En s'inspirant du programme précédent, écrire un programme permettant de tracer un rectangle de dimension 100 unités sur 30 unités. On utilisera la définition et les propriétés du rectangle.

.....

.....

.....

.....

.....

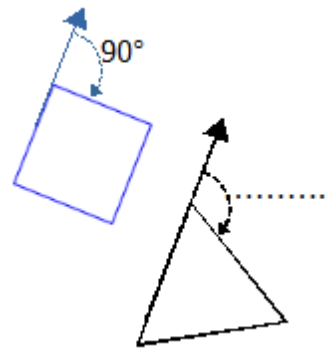
.....

.....

.....

B) Avec des polygones réguliers.

Pour tracer un carré, ou tout autre polygone régulier, on peut utiliser les propriétés des angles des polygones pour trouver l'angle de rotation



En déduire ainsi la mesure de chaque rotation pour construire un triangle équilatéral :

a) Ecrire un programme pour tracer un triangle équilatéral puis le tester :

.....

.....

.....

.....

.....

.....

.....

.....

b) Ecrire un programme pour tracer un octogone régulier puis le tester :

.....

.....

.....

.....

.....

.....

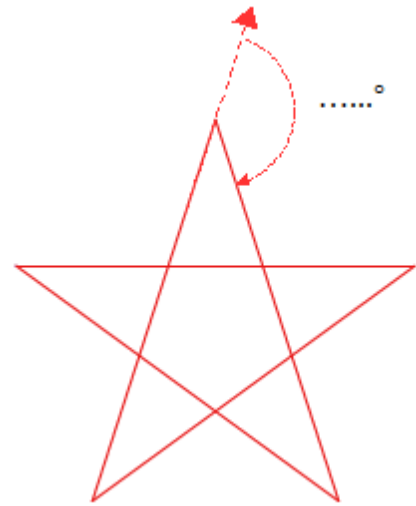
.....

.....

C) L'étoile à 5 branches :

a) Cette étoile est composée d'un pentagone régulier et de 5 triangles isocèles. Quelle est la mesure de l'angle de rotation ? Justifier par des calculs.

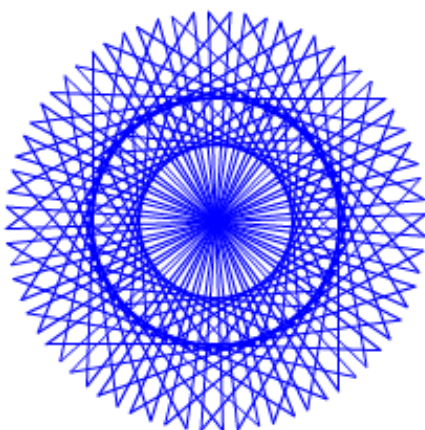
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....



b) Ecrire un programme afin de tracer cette étoile et tester le.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

c) Pour aller plus loin : écrire un programme afin de tracer cette figure composée de 36 étoiles :



.....
.....
.....
.....
.....
.....
.....
.....
.....
.....



Coup de pouce : il faut répéter 36 fois l'action tracer une étoile mais avant chaque action de tracer il faut tourner de °

CORRIGÉ ROSACE

```

quand flag verte pressée
effacer tout
aller à x: -134 y: 5
répéter 36 fois
  relever le stylo
  tourner de 10 degrés
  avancer de 3
  choisir l'intensité 60 pour le stylo
  ajouter 20 à couleur du stylo
  répéter 36 fois
    stylo en position d'écriture
    avancer de 10
    tourner de 10 degrés
  
```

CORRIGÉ ANNEAUX OLYMPIQUES (attention, il faut créer un lutin pour chaque cercle)

```

quand flag verte pressée
effacer tout
choisir la taille 10 pour le stylo
cercle bleu

définir cercle bleu
  relever le stylo
  aller à x: -213 y: 0
  s'orienter à 180
  choisir la couleur pour le stylo
  stylo en position d'écriture
  répéter 36 fois
    avancer de 10
    tourner de 10 degrés

quand flag verte pressée
choisir la taille 10 pour le stylo
cercle jaune

définir cercle jaune
  relever le stylo
  aller à x: -32 y: -75
  s'orienter à 0
  choisir la couleur pour le stylo
  stylo en position d'écriture
  répéter 36 fois
    avancer de 10
    tourner de 10 degrés

quand flag verte pressée
choisir la taille 10 pour le stylo
cercle noir

définir cercle noir
  relever le stylo
  aller à x: -80 y: 0
  s'orienter à 180
  choisir la couleur pour le stylo
  stylo en position d'écriture
  répéter 36 fois
    avancer de 10
    tourner de 10 degrés

définir cercle vert
  relever le stylo
  aller à x: 100 y: -75
  s'orienter à 0
  choisir la couleur pour le stylo
  stylo en position d'écriture
  répéter 36 fois
    avancer de 10
    tourner de 10 degrés

quand flag verte pressée
choisir la taille 10 pour le stylo
cercle vert

définir cercle rouge
  relever le stylo
  aller à x: 53 y: 0
  s'orienter à 180
  choisir la couleur pour le stylo
  stylo en position d'écriture
  répéter 36 fois
    avancer de 10
    tourner de 10 degrés

quand flag verte pressée
choisir la taille 10 pour le stylo
cercle rouge
  
```

1. Dans chaque cas, observe bien le programme et précise ce que va tracer le lutin.

<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture répéter 4 fois avancer de 80 tourner ↻ de 90 degrés relever le stylo </pre>	<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture répéter 2 fois avancer de 80 tourner ↻ de 90 degrés avancer de 150 tourner ↻ de 90 degrés relever le stylo </pre>	<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture répéter 4 fois avancer de -80 tourner ↻ de 90 degrés relever le stylo </pre>
<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture répéter 3 fois avancer de 80 tourner ↻ de 60 degrés relever le stylo </pre>	<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture répéter 3 fois avancer de 80 tourner ↻ de 120 degrés relever le stylo </pre>	<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture répéter 6 fois avancer de 80 tourner ↻ de 60 degrés relever le stylo </pre>
<pre> quand [drapeau vert] pressé effacer tout aller à x: 0 y: 0 stylo en position d'écriture aller à x: 0 y: 60 aller à x: 60 y: 60 aller à x: 60 y: 0 aller à x: 0 y: 0 relever le stylo </pre>	<pre> quand [drapeau vert] pressé effacer tout aller à x: -15 y: 30 stylo en position d'écriture aller à x: -15 y: 60 aller à x: 60 y: 60 aller à x: -15 y: 30 relever le stylo </pre>	<pre> quand [drapeau vert] pressé effacer tout aller à x: 10 y: 10 stylo en position d'écriture aller à x: 10 y: 150 aller à x: 60 y: 150 aller à x: 60 y: 10 aller à x: 10 y: 10 relever le stylo </pre>

2. Programme-les avec SCRATCH.

1. Dans chaque cas, indique la longueur du segment tracé par le lutin

<p>aller à x: 56 y: 0</p> <p>stylo en position d'écriture</p> <p>aller à x: 105 y: 0</p>	<p>aller à x: -60 y: 15</p> <p>stylo en position d'écriture</p> <p>aller à x: 36 y: 15</p>	<p>aller à x: -155 y: 23</p> <p>stylo en position d'écriture</p> <p>aller à x: -60 y: 23</p>
<p>aller à x: 0 y: 56</p> <p>stylo en position d'écriture</p> <p>aller à x: 0 y: 105</p>	<p>aller à x: 0 y: 0</p> <p>stylo en position d'écriture</p> <p>aller à x: 10 y: 10</p>	<p>aller à x: 0 y: -17</p> <p>stylo en position d'écriture</p> <p>aller à x: 0 y: 17</p>
<p>stylo en position d'écriture</p> <p>avancer de 30</p> <p>avancer de 10</p> <p>avancer de 5</p> <p>avancer de 60</p>	<p>stylo en position d'écriture</p> <p>avancer de -25</p> <p>avancer de -3</p> <p>avancer de 5</p> <p>avancer de 60</p>	<p>stylo en position d'écriture</p> <p>avancer de -3</p> <p>avancer de -7</p> <p>avancer de -19</p> <p>avancer de -31</p>
<p>stylo en position d'écriture</p> <p>répéter 10 fois</p> <p>avancer de 3</p>	<p>stylo en position d'écriture</p> <p>répéter 7 fois</p> <p>avancer de -5</p>	<p>stylo en position d'écriture</p> <p>avancer de 8</p> <p>répéter 6 fois</p> <p>avancer de 3</p>
<p>stylo en position d'écriture</p> <p>avancer de 15</p> <p>répéter 7 fois</p> <p>avancer de 5</p> <p>avancer de 10</p>	<p>stylo en position d'écriture</p> <p>avancer de 50</p> <p>répéter 6 fois</p> <p>avancer de -3</p> <p>avancer de -25</p>	<p>stylo en position d'écriture</p> <p>répéter 10 fois</p> <p>répéter 6 fois</p> <p>avancer de -2</p>
<p>stylo en position d'écriture</p> <p>répéter 10 fois</p> <p>avancer de 20</p> <p>répéter 6 fois</p> <p>avancer de -25</p>	<p>stylo en position d'écriture</p> <p>répéter 20 fois</p> <p>avancer de -6</p> <p>répéter 10 fois</p> <p>avancer de 5</p>	<p>stylo en position d'écriture</p> <p>répéter 15 fois</p> <p>avancer de -2</p> <p>répéter 10 fois</p> <p>avancer de -3</p>
<p>aller à x: 0 y: 0</p> <p>stylo en position d'écriture</p> <p>répéter 15 fois</p> <p>avancer de 2</p> <p>aller à x: 60 y: 0</p>	<p>aller à x: 0 y: 0</p> <p>stylo en position d'écriture</p> <p>répéter 60 fois</p> <p>avancer de 2</p> <p>aller à x: 10 y: 0</p>	<p>aller à x: 0 y: 0</p> <p>stylo en position d'écriture</p> <p>répéter 10 fois</p> <p>avancer de -5</p> <p>aller à x: 10 y: 0</p>

1. Réécrire ces programmes en utilisant les boucles.

		<p>Qu'obtient-on ici ?</p>
--	--	----------------------------

2. Réécrire ces programmes sans utiliser de boucle

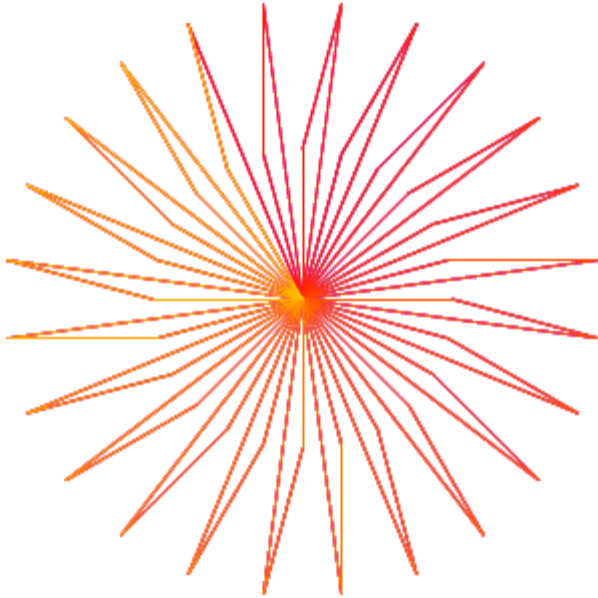
--	--	--

3. Réécrire ces programmes en utilisant une seule boucle.

--	--	--

Ecrire un programme avec SCRATCH permettant de tracer les figures géométriques ci-dessous.

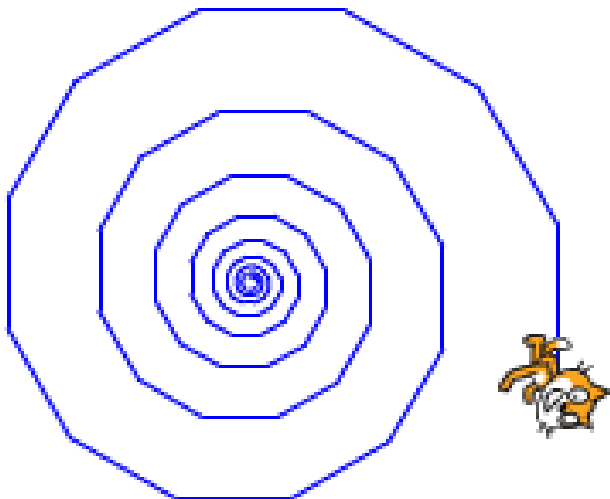
Une rosace



```

quand cliqué
  aller à x: 0 y: 0
  effacer tout
  stylo en position d'écriture
  choisir la couleur pour le stylo
  répéter indéfiniment
    aller à x: 0 y: 0
    avancer de 75
    tourner de 15 degrés
    avancer de 75
    ajouter 1 à couleur du stylo
  
```

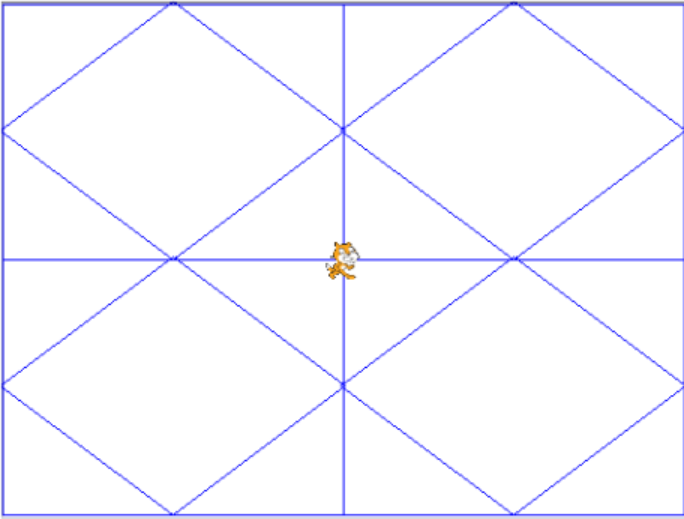
Une spirale



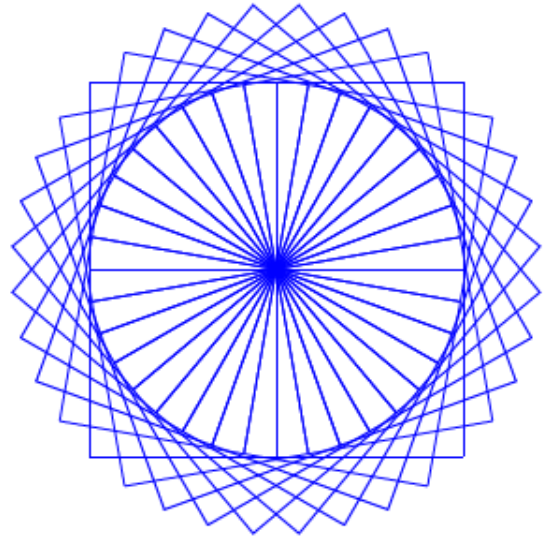
```

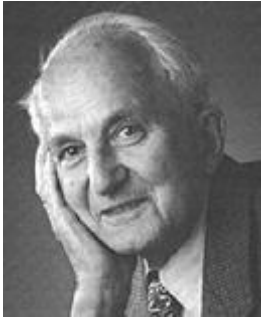
quand pressé
  effacer tout
  mettre à 30 % de la taille initiale
  aller à x: 0 y: 0
  s'orienter à 90
  mettre pas à 1
  stylo en position d'écriture
  répéter 100 fois
    avancer de pas
    tourner de 30 degrés
    mettre pas à pas * 1.04
  relever le stylo
  
```

Un pavage



Une rosace de carrés





En mathématiques, on appelle conjecture, une règle qui n'a jamais été prouvée. On a vérifié cette règle sur beaucoup d'exemples mais on n'est pas sûr qu'elle soit toujours vraie.

C'est le cas de la conjecture de Syracuse découverte par le mathématicien allemand Lothar Collatz (photo ci-contre) en 1930.

Depuis, bon nombre de mathématiciens cherchent à expliquer pourquoi cette conjecture est vraie, mais aujourd'hui personne n'y est encore arrivée.

Toi, jeune mathématicien en herbe, si tu réussis à prouver cette conjecture, tu deviendras célèbre et ton nom figurera dans les livres de maths !

Énoncé de la conjecture de Syracuse

Prendre un nombre entier:

- Si ce nombre est pair, le diviser par 2.
- Si ce nombre est impair, prendre le triple et ajouter 1.

On obtient un nouveau nombre entier et on recommence :

- Si ce nombre est pair, le diviser par 2.
- Si ce nombre est impair, prendre le triple et ajouter 1.

Et on recommence ainsi de suite avec les entiers successifs obtenus...

La conjecture de Syracuse dit qu'à la fin, on obtient toujours 1.

- Tester cet algorithme avec les nombres 10; 13 et 21.
- Programmer cet algorithme sur SCRATCH

Prolongement possible

Améliore ton programme afin qu'il compte le nombre d'itérations (étapes) nécessaires pour obtenir 1.

CORRIGÉ

```
quand flag pressé
supprimer l'élément tout de la liste Syracuse
demander Quel nombre choisissez-vous ? et attendre
mettre nombre à réponse
ajouter nombre à Syracuse
répéter jusqu'à nombre = 1
  si nombre modulo 2 = 0 alors
    mettre nombre à nombre / 2
  sinon
    mettre nombre à 3 * nombre + 1
  ajouter nombre à Syracuse
```

CORRIGÉ prolongement

```
quand flag pressé
supprimer l'élément tout de la liste Syracuse
mettre nombre d'itérations à 0
mettre plus haute valeur obtenue à 0
demander Quel nombre choisissez-vous ? et attendre
mettre nombre à réponse
mettre plus haute valeur obtenue à nombre
ajouter nombre à Syracuse
répéter jusqu'à nombre = 1
  si plus haute valeur obtenue < nombre alors
    mettre plus haute valeur obtenue à nombre
  mettre nombre d'itérations à nombre d'itérations + 1
  si nombre modulo 2 = 0 alors
    mettre nombre à nombre / 2
  sinon
    mettre nombre à 3 * nombre + 1
  ajouter nombre à Syracuse
```

Convertir des degrés Celsius en degrés Fahrenheit

extrait du sujet DNB entres étrangers 17 juin 2014

Il existe différentes unités de mesure de la température : en France on utilise le degré Celsius ($^{\circ}\text{C}$), aux États-Unis on utilise le degré Fahrenheit ($^{\circ}\text{F}$).

Pour passer des degrés Celsius aux degrés Fahrenheit, on multiplie le nombre de départ par 1,8 et on ajoute 32 au résultat. 1.

1. Qu'indiquerait un thermomètre en degrés Fahrenheit si on le plongeait dans une casserole d'eau qui gèle? On rappelle que l'eau gèle à 0°C .
2. Qu'indiquerait un thermomètre en degrés Celsius si on le plongeait dans une casserole d'eau portée à 212°F ? Que se passe-t-il?
3. a) Si l'on note x la température en degré Celsius et $f(x)$ la température en degré Fahrenheit, exprimer $f(x)$ en fonction de x .
b) Comment nomme-t-on ce type de fonction ?
c) Quelle est l'image de 5 par la fonction f ?
d) Quel est l'antécédent de 5 par la fonction f ?
e) Traduire en terme de conversion de température la relation $f(10) = 50$.

Prolongement

Réalise un petit programme qui demande une température en degrés Celsius et qui permet de la convertir en degré Fahrenheit et inversement.

Des éléments de correction

```

démarrer quand le drapeau est cliqué
demander Donner une température en °C, et attendre
mettre Celsius à réponse
mettre Fahrenheit à 1.8 * Celsius + 32
mettre Fahrenheit à arrondi de Fahrenheit
dire regroupe Celsius regroupe °C = regroupe Fahrenheit °F
demander Donner une température en °F, et attendre
mettre Fahrenheit à réponse
mettre Celsius à Fahrenheit - 32 / 1.8
mettre Celsius à arrondi de Celsius
dire regroupe Fahrenheit regroupe °F = regroupe Celsius °C
  
```

Exemples

d'activités

"LUDIQUES"

(programmation de jeux)

Partie 1 : Dans la cour

1. Par groupe de 4, en utilisant des instructions simples, en français, il faudra indiquer un parcours allant de carré en carré permettant de rejoindre l'objectif fixé, en évitant les obstacles indiqués sur le plan.
2. Puis Chaque groupe donne ses instructions à un autre groupe qui devra les suivre pour atteindre l'objectif.

Partie 2 : En salle informatique

1. Ouvrir le logiciel Scratch (un raccourci se trouve sur le bureau)
2. Cliquer sur fichier puis ouvrir
3. Choisir le fichier nommé **deplacements_dans_le_college.sb2** (il se trouve dans GROUPE / 6a ou 6b /TRAVAIL/MATHS 6e)
4. Choisir l'arrière plan correspondant à celui de votre groupe
5. Dessiner un lutin qui aura la forme d'un point.
6. Reproduire les déplacements écrits sur papier à l'aide du logiciel Scratch. Pour cela programmer, les déplacements gauche, droite, haut ou bas en utilisant les flèches du clavier.

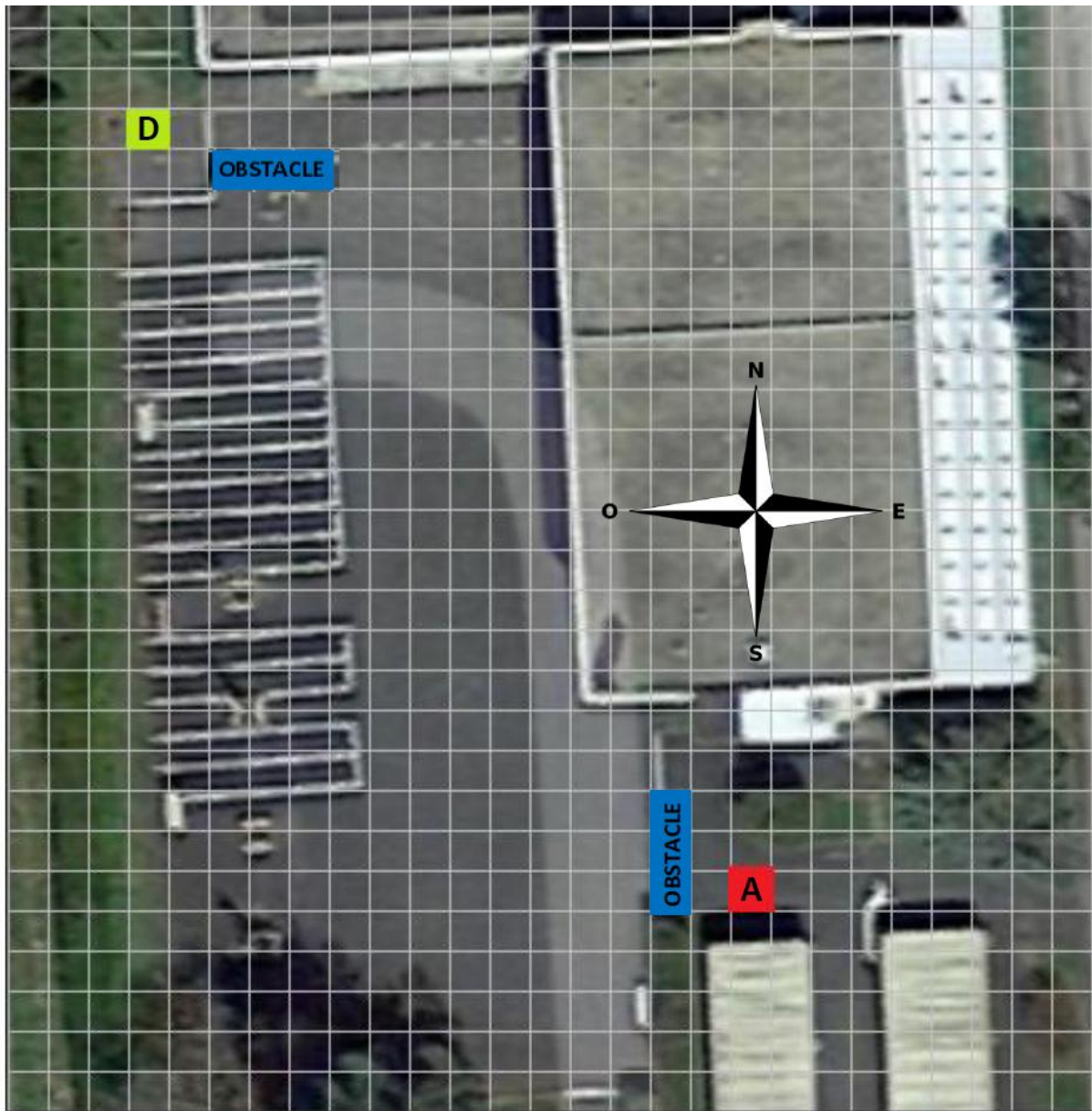
aide : utiliser cette brique

A Scratch block with a light blue background and a dark blue border. The text inside is "quand flèche haut est cliqué". The word "quand" is in a rounded rectangle, "flèche haut" is in a dropdown menu, and "est cliqué" is in a rounded rectangle.

7. Puis enregistrer votre travail et le nommer **deplacements_dans_le_college_PRENOMS.sb2**

Un exemple de document distribué aux élèves (groupe n°1)

La vue aérienne du collège a été prise sur Google Maps puis copiée dans Photofiltre avec lequel on a rajouté le quadrillage. Puis on a rajouté le départ D et l'arrivée A et des obstacles. Chaque groupe part d'un point de départ différents. Cette activité est aussi l'occasion de travailler sur les consignes. Le document est imprimé en couleurs et au format A4 (un ou deux exemplaires par groupe).



GROUPE 1

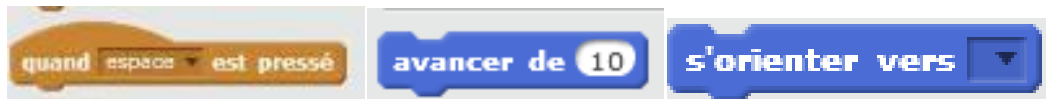
En utilisant des instructions simples, en français, il faudra indiquer un parcours allant de carré en carré permettant de rejoindre l'objectif fixé, en évitant les obstacles indiqués sur le plan.

Travail de préparation en amont

Mission 1 : faire avancer le lutin

Programme les déplacements de ton lutin vers le haut, vers le bas, vers la droite, vers la gauche en appuyant sur les flèches du clavier.

Pour cela, tu peux utiliser les blocs ci-dessous :



Mission 2 : créer un labyrinthe

1. Tu peux créer un labyrinthe en dessinant des murs sur la scène ou tu peux choisir un labyrinthe sur ce site : <http://www.mazegenerator.net/> puis repasser sur les murs en dessinant sur la scène.
2. Complète le script de ton lutin, en ajoutant le fait que le lutin rebondit et recule de quelques pas s'il touche les murs du labyrinthe et de même s'il touche les bord (car sinon, ton lutin sort de l'écran)

Projet : créer un jeu de labyrinthe

Consigne :

Créer un jeu de labyrinthe avec un chevalier, une princesse et un dragon.

Le but est de réaliser un jeu où l'on dirige le chevalier dans le labyrinthe pour éviter un dragon et sauver la princesse.

Améliore ton programme

Créer un nouvel objet nommé potion en forme de fiole.

Rajouter les instructions pour faire en sorte que si le chevalier touche la fiole alors il devient invincible, et toucher le dragon n'a plus aucun effet.

Changer la fiole de place toutes les 2 secondes au hasard sur l'écran pour pimenter le jeu.

Deux ressources très bien faites :

http://www.loria.fr/~quinson/Mediation/Coding4Kids/Scratch_Calestroupat.pdf

<http://www.derepas.com/scratch/cm1/>

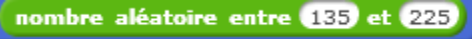
Travail de préparation

Mission n°1 :

Créer un lutin en forme de raquette et programmer ses déplacements vers la droite et vers la gauche.

Mission n°2:

Rajouter un lutin en forme de balle et écrire un programme permettant :

- de la faire se déplacer de manière aléatoire **en utilisant le bloc** 
- et de la faire rebondir quand elle touche les bords et quand elle touche la raquette **en utilisant les blocs :**









Réalisation d'un jeu de Pong

Travail différencié (proposer l'une de ces activités en fonction des besoins de chaque élève)

- ⇒ Réalise un programme permettant de jouer au jeu de Pong à un joueur
- ⇒ Réalise un programme permettant de jouer au jeu de Pong à deux joueurs
- ⇒ Améliore le jeu de Pong (compteur de points, vitesse de la balle, créer un arrière plan qui donne le score final)

Proposer des aides différenciées sous forme:

- ⇒ de fiches (voir ci-après)
- ⇒ ou de vidéos tutoriels (par exemple : la vidéo de Justinien un élève du collège de Loudun, <https://youtu.be/Em5SLBWUm30>)

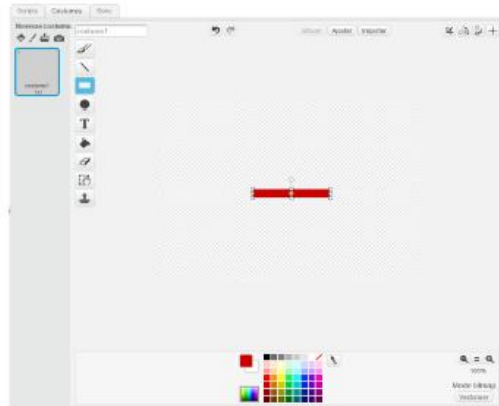
Création d'un jeu de pong. Partie graphique.



Tout d'abord, il faut créer deux lutins (une raquette et une balle) et un arrière-plan.

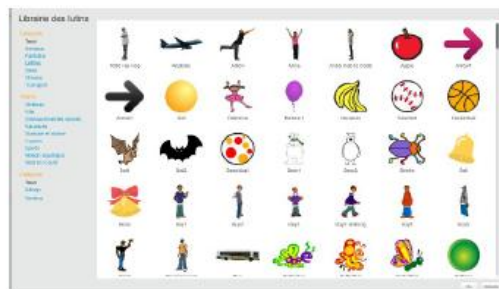
Créer une raquette :

Dessiner un nouveau lutin de forme rectangulaire.



Créer une balle :

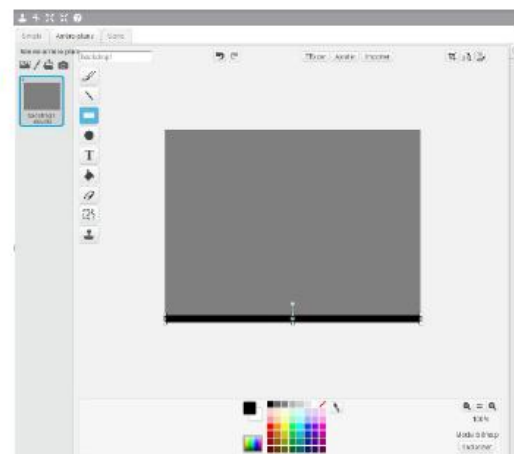
On peut choisir un lutin dans la bibliothèque



ou en créer un.

Créer l'arrière plan :

Il faut créer une bande horizontale de couleur qui symbolisera la fin de la partie.



Création d'un jeu de pong. Partie scripts.

Maintenant, il faut gérer les déplacements de la raquette et de la balle.

Déplacement de la raquette :

Une partie débutera en cliquant sur le drapeau vert.

En début de partie, la raquette doit être positionnée au centre.

Ensuite, elle se déplace à l'aide des flèches du clavier (à droite ou à gauche).

La commande "Rebondir si le bord est atteint" évitera à la raquette de sortir de l'écran.

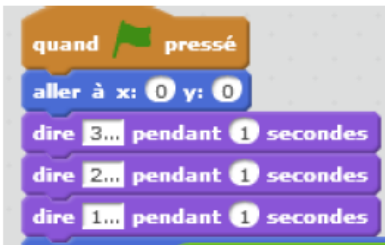


Déplacement de la balle :

La balle doit avancer et rebondir, jusqu'à ce qu'elle touche le bas de l'écran, c'est à dire qu'elle touche la bande colorée.

Lorsque la balle touche la raquette, l'angle de rebond est défini de façon aléatoire.

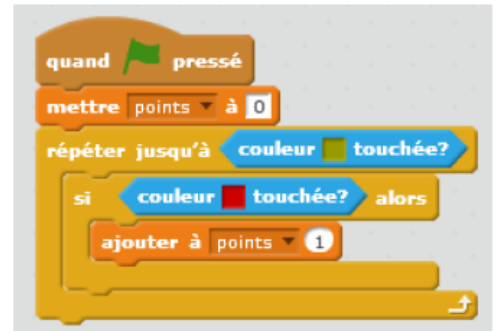
Pour éviter que la balle parte directement, on peut insérer un compte à rebours :



Création d'un jeu de pong. Améliorations possibles.

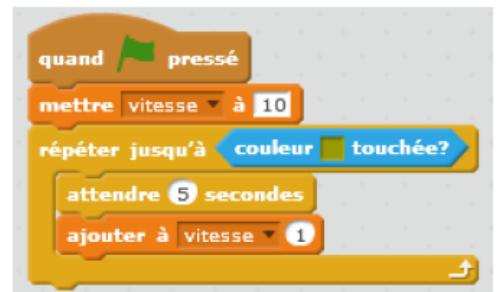
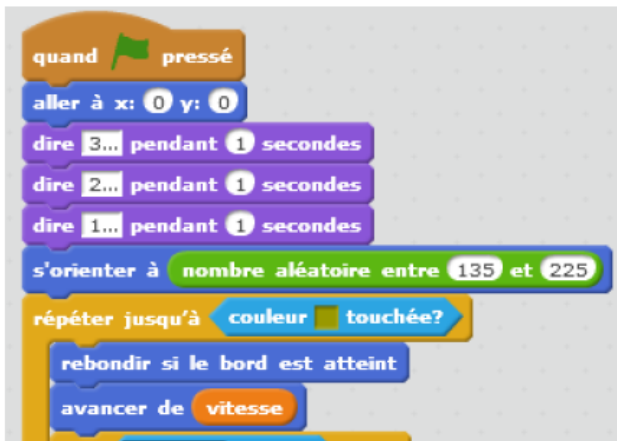
Insérer un compteur de points :

On marque un point à chaque fois que la balle touche la raquette. Il faut créer une variable "points", initialisée à 0 et qui augmente de 1 à chaque fois que la balle touche la raquette.



Augmenter la vitesse de la balle :

Pour que la balle accélère progressivement, il faut créer une variable "vitesse", initialisée à 10 et qui augmente de 1 toutes les 5 secondes. Ensuite, il faut changer le "Avancer de 10" en "Avancer de "Vitesse".



Créer un arrière plan qui donne le score final :

Il faut créer un arrière-plan et un nouveau lutin qui indiquera le score.

Pour changer d'arrière-plan, on utilise :



Il faut penser à "cacher" ou "montrer" les différents lutins suivant les fonds d'écrans, on pourra alors utiliser les blocs :



et



Exemples d'activités

TRANSVERSALES

(dans le cadre des EPI ou de l'Accompagnement Personnalisé)

Activité 1: PLURIEL

Il existe une règle principale de construction du pluriel d'un mot : on met un 's' à la fin du mot.

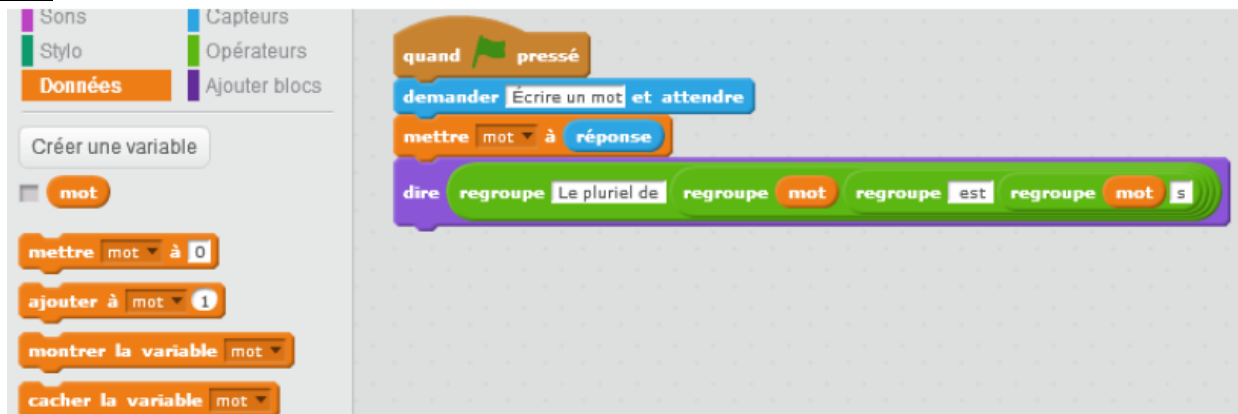
Exemples : un arbre/des arbres ; une voiture/des voitures

L'objectif est d'écrire un programme construisant le pluriel de mots respectant cette règle du pluriel :



a) Établir une liste de 20 mots respectant cette règle du pluriel.

b) Écrire un programme avec Scratch construisant le pluriel de ces mots.

CORRIGÉ**Activité 2: FÉMININ**

Il existe une règle principale de construction du féminin d'un adjectif : on met un 'e' à la fin.

Exemple : grand/grande ; lent/lente

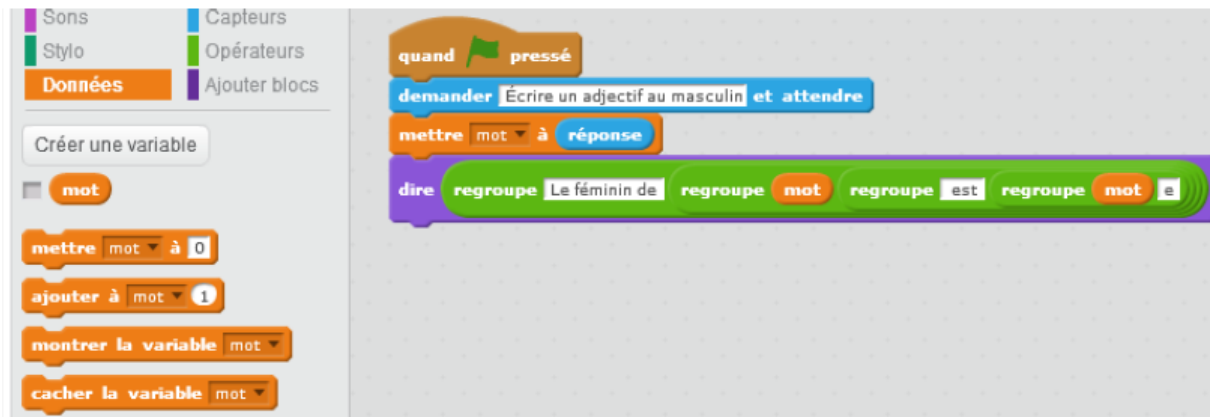
L'objectif est d'écrire un programme construisant le féminin d'adjectifs respectant cette règle :



a) Établir une liste de 20 adjectifs respectant cette règle du féminin.

b) Écrire un programme avec Scratch construisant le féminin de ces adjectifs.

CORRIGÉ



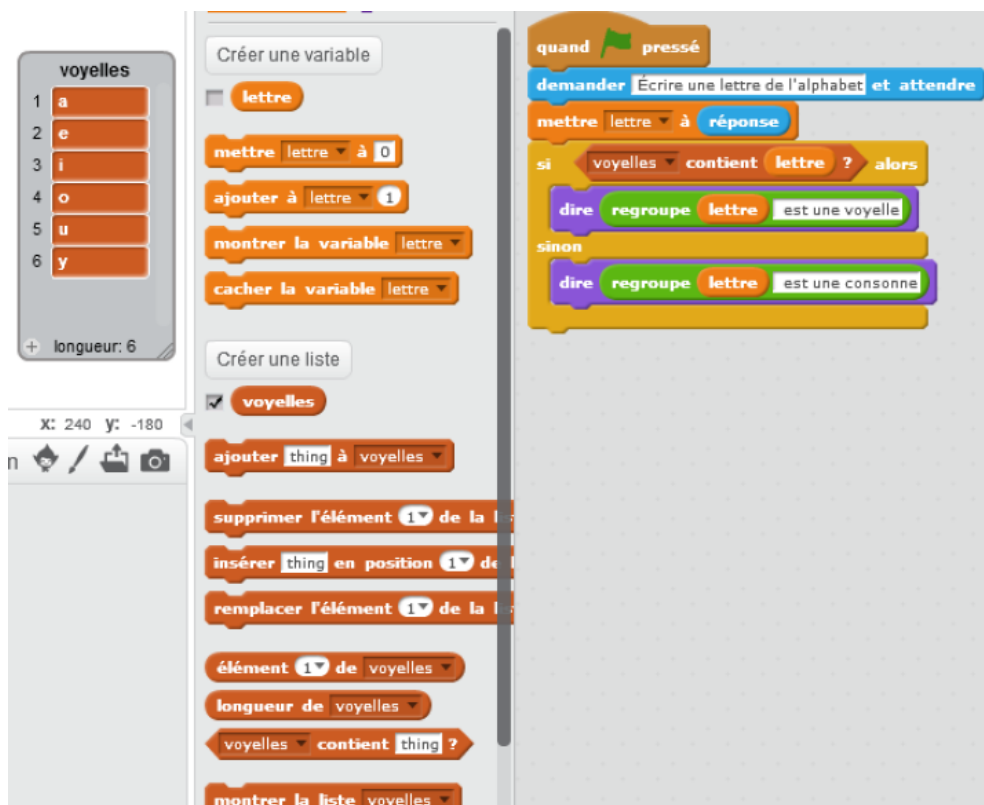
Activité 3: VOYELLE OU CONSONNE ?

1) L'objectif est de demander au lutin de Scratch si une lettre est une voyelle ou si c'est une consonne :



Pour cela, il faudra d'abord créer une liste de voyelles :
Et construire le programme en s'appuyant sur cette liste.

CORRIGÉ

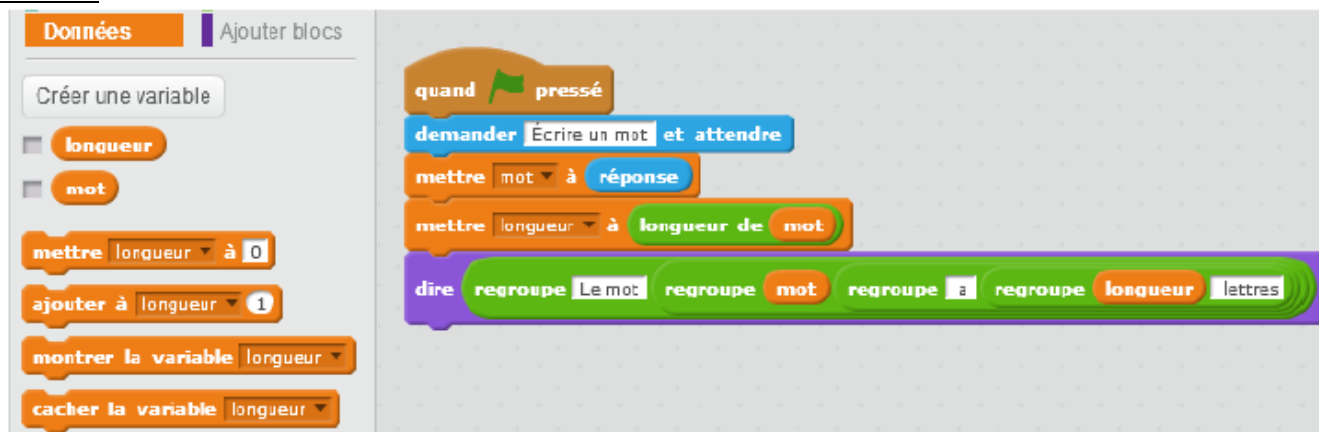


Activité 4 : NOMBRE DE LETTRES DANS UN MOT

Dans cette partie, on écrit un mot et il s'agit de donner le nombre de lettres de ce mot.



CORRIGÉ



Activité 5 : PALINDROME

⇒ Première partie :

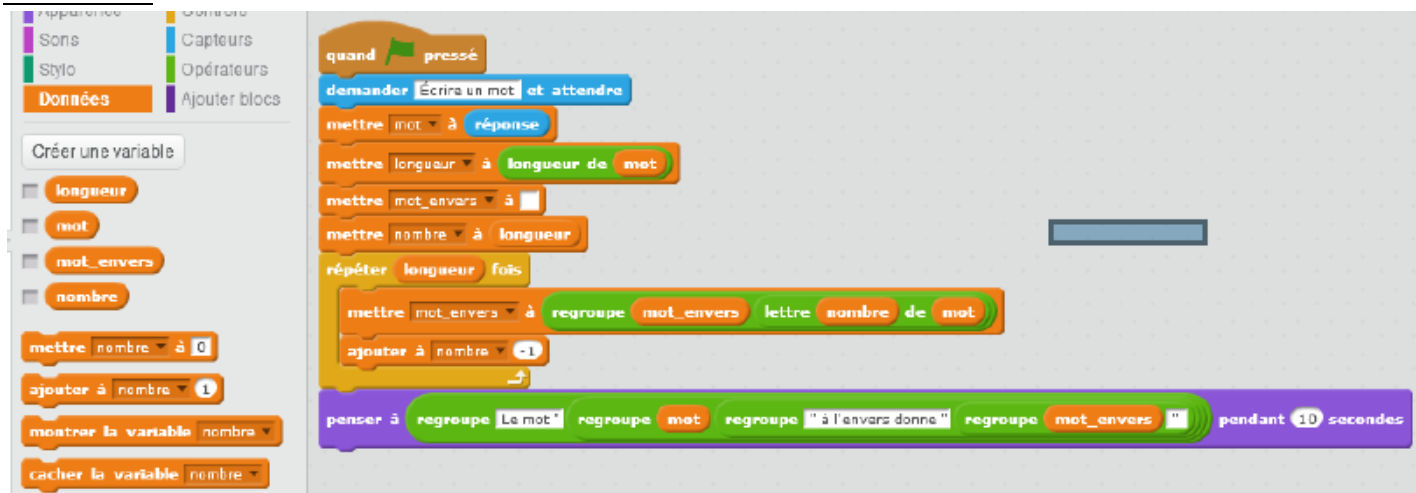
- 1) Chercher ce qu'est un palindrome.
- 2) Établir alors une liste d'une dizaine de palindromes.

⇒ Deuxième partie :

Construire un programme qui écrit un mot donné à l'envers



CORRIGÉ



⇒ **Troisième partie :**

Écrire un programme qui indique si un mot est un palindrome ou n'en est pas un.



CORRIGÉ

```

    quand le drapeau est cliqué
    demander "Écrire un mot" et attendre
    mettre mot à réponse
    mettre longueur à longueur de mot
    mettre mot_ens_à ""
    mettre nombre à longueur
    répéter longueur fois
    mettre mot_ens_à regroupe mot_ens_ lettre nombre de mot
    ajouter à nombre -1
    si mot = mot_ens_ alors
    penser à regroupe mot est un palindrome pendant 5 secondes
    sinon
    penser à regroupe mot n'est pas un palindrome pendant 5 secondes
  
```

Activité 6 : PLURIELS PARTICULIERS

⇒ **Première partie : le pluriel des mots en 'ou'**

- 1) Les mots en 'ou' prennent un 's' au pluriel, à l'exception de 7 mots. Quels sont ces 7 mots ?
- 2) Établir une liste de 20 mots en 'ou'.
- 3) Réaliser un programme donnant le pluriel des mots en 'ou' et tenant compte des 7 cas particuliers.



CORRIGÉ

```

    quand le drapeau est cliqué
    demander "Écrire un mot en 'ou'" et attendre
    mettre mot à réponse
    si mots en 'ou' contient mot ? alors
    penser à regroupe Le pluriel de regroupe mot regroupe est regroupe mot
    sinon
    penser à regroupe Le pluriel de regroupe mot regroupe est regroupe mot s
  
```

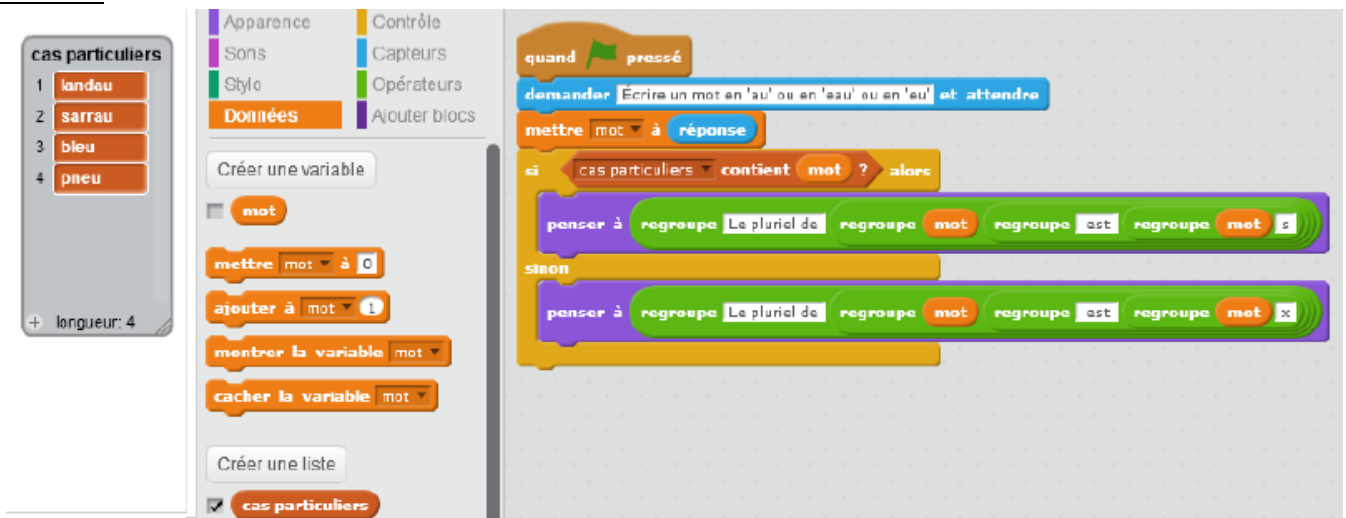
⇒ Deuxième partie : le pluriel des mots en 'au', 'eau' et 'eu'

Ces mots prennent un 'x' au pluriel, à l'exception de landau, sarrau, pneu et bleu.

- 1) Donner la définition de 'landau' et de 'sarrau'.
- 2) Établir une liste de 10 mots en 'au', de 10 mots en 'eau' et de 10 mots en 'eu'.
- 3) Réaliser un programme donnant le pluriel des mots en 'au', 'eau' et 'eu' et tenant compte des 4 cas particuliers.



CORRIGÉ

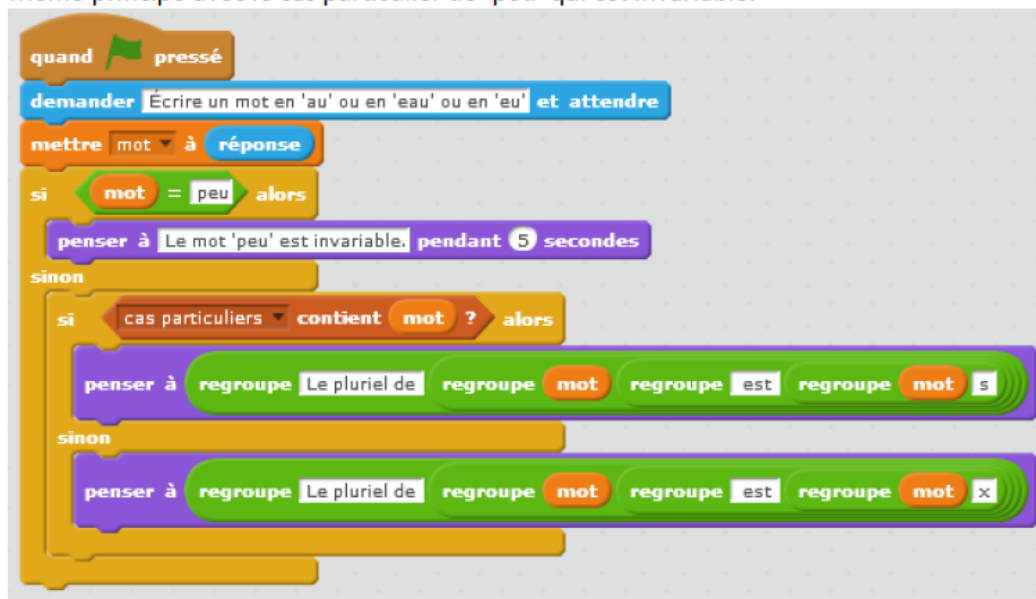


⇒ Troisième partie :

Réunir les deux programmes précédents en un seul !

CORRIGÉ

Même principe avec le cas particulier de 'peu' qui est invariable.



À noter que 'prou' est invariable aussi, mais nettement moins utilisé.

```

quand flag cliquée
demander "Écrire un mot en 'ou', 'eu', 'eau', 'au' et attendre"
mettre mot à réponse
mettre taille à longueur de mot - 1
si lettre taille de mot = 0 alors
  si mots en 'ou' contient mot ? alors
    penser à regroupe Le pluriel de regroupe mot regroupe est regroupe mot x
  sinon
    penser à regroupe Le pluriel de regroupe mot regroupe est regroupe mot s
  sinon
    si mots en eu au contient mot ? alors
      penser à regroupe Le pluriel de regroupe mot regroupe est regroupe mot s
    sinon
      penser à regroupe Le pluriel de regroupe mot regroupe est regroupe mot x
  
```

Avec bien sûr les deux listes présentes :

mots en 'ou'		mots en eu au	
1	bijou	1	landau
2	caillou	2	sarrau
3	chou	3	pneu
4	genou	4	bleu
5	hibou		
6	joujou		
7	pou		
+ longueur: 7		+ longueur: 4	

La conception du quizz se fera de cette façon :

1. on pose une question
2. si c'est la bonne réponse, un évènement apparaît
3. si la réponse n'est pas celle attendue, un autre évènement a lieu

Pour **faire un questionnaire dans Scratch**, il faut :

ATTENTION : Dans les explications suivantes, il a été oublié de poser la question (« demander... »)

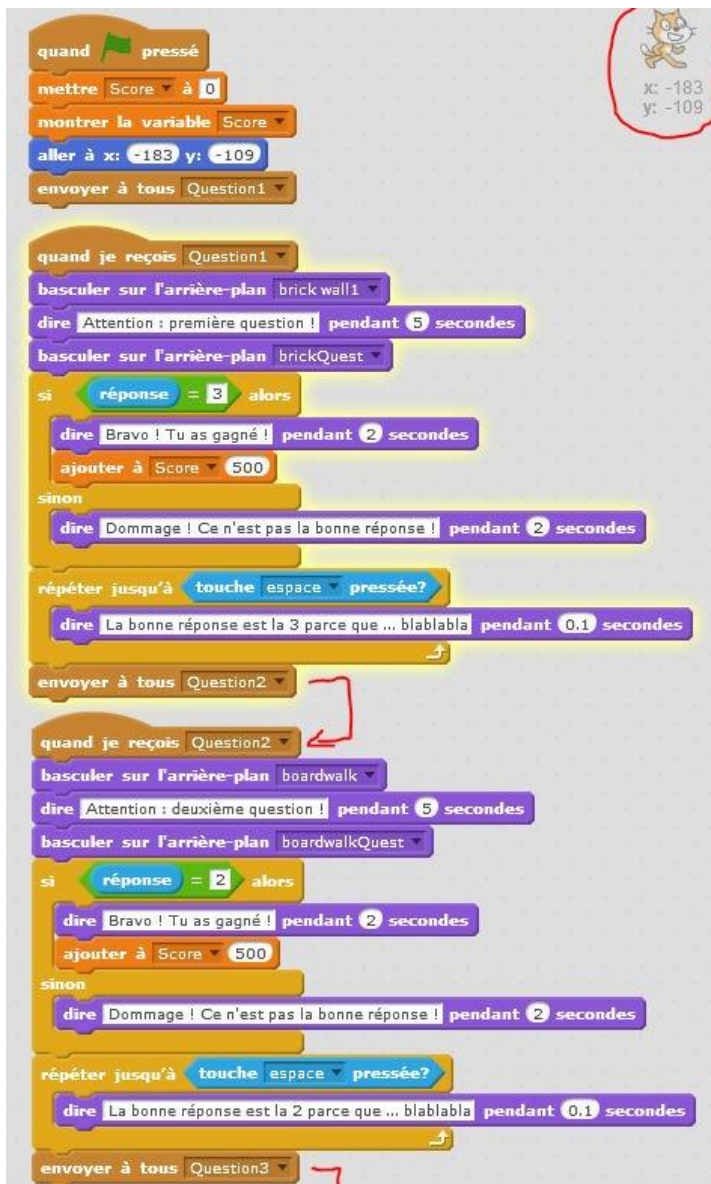
→ **ÉTAPE 1** : Charger deux arrières plans identiques et écrire les réponses sur l'un des deux

TUTORIEL en vidéo : https://youtu.be/PZ_-dPOG5Lc

→ **ÉTAPE 2** : Créer un évènement « Question1 » et le script nécessaire

TUTORIEL en vidéo : https://youtu.be/QuR_SvUDj-g

A la fin de la question1, pour passer à la question suivante, créer et envoyer un évènement « Question2 »



Des exemples de quizz :

- un quizz sur les repères en histoire-géographie :

<https://scratch.mit.edu/projects/3140182/>

- un quizz en SVT :

<https://scratch.mit.edu/projects/61555496/>

Un exemple de programme