



LIVRET SCRATCH 3EME

NOM :

Prénom :

Classe :

**Ce livret est à conserver précieusement dans ton cahier de maths.
Il contient toutes les séances sur Scratch que tu feras en salle informatique cette année.**

**Lors de la 1^{ère} séance tu créeras dans ton répertoire personnel sur le disque réseau, un répertoire « scratch »
Tu enregistreras dans ce répertoire tout le travail fait à chaque séance.
Tu nommeras les fichiers « scratch-seance1 », « scratch-seance2 »**

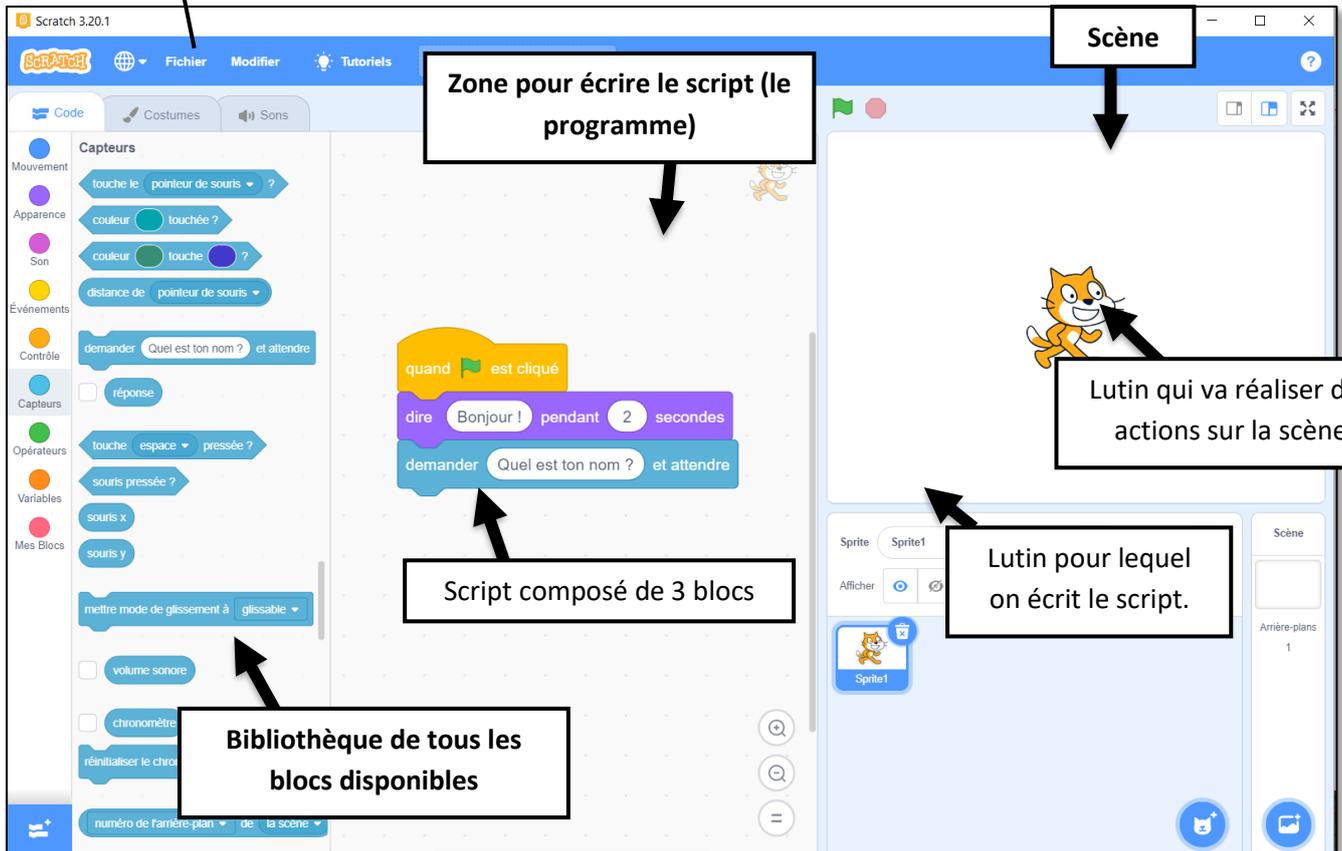
A quoi sert scratch ?

C'est un logiciel de programmation.

Il permet de créer des scripts qui contiennent des instructions pour :

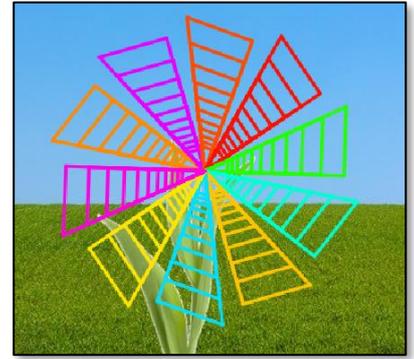
- réaliser une construction géométrique
- des calculs
- des jeux,

Menu permettant d'ouvrir
ou d'enregistrer des fichiers





Dessiner une fleur



But du script :

L'objectif de ce TP est de réaliser la fleur géométrique ci-contre. Cette fleur est construite à partir de triangles rectangles de plus en plus grands.

	Compétence évaluée	Évaluation			
1.3	Réaliser un programme simple	MTS	MS	MF	MI

Partie I : préparation

- 1) **Ouvre** le logiciel scratch 3.
- 2) **Enregistre** le fichier dans ton répertoire personnel (dans un répertoire scratch que tu as déjà créé)
Donne comme nom à ton fichier : « *projet-scratch-fleur.sb3* »

Principe de construction de la fleur :

- Chaque pétale de la fleur est constitué de 11 triangles rectangles emboîtés les uns dans les autres.
- Il faudra créer dans le script un bloc qu'on appellera « triangle » qui se chargera de construire un triangle.
- On répètera 11 fois ce bloc pour construire un pétale.
- Puis il faudra répéter 10 fois cette construction pour obtenir les 10 pétales de la fleur

Défi n°1 Changer l'arrière-plan

Explications :

- Il faut changer l'arrière-plan par l'image qui convient.

Programmation :

1) **Positionne la souris** sur « Choisir un arrière-plan » dans la zone « scène »

Puis **clique** sur « Importer un arrière plan ».



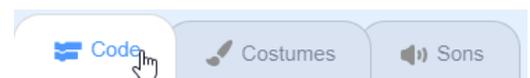
2) **Sélectionne** le fichier « *fond-fleur.png* » dans le répertoire « *Commun* » du réseau. **Valide**.

3) **Clique** sur « Convertir en vecteur ».



4) **Clique** sur l'image dans la zone de dessin puis **agrandis** la pour qu'elle prenne tout l'espace.

5) Pour terminer, **clique** sur le lutin chat puis **clique** sur « code ».



Défi n°2

Construire un triangle



Explications :

- Chaque pétale de la fleur est constitué, de triangles rectangles de plus en plus grand.
- Chaque triangle sera construit par un « bloc » dans le script. Puis on répètera autant de fois que l'on veut ce bloc de construction d'un triangle rectangle et en lui indiquant les mesures nécessaires.
- La construction de tous les triangles commencera au centre de la fleur c'est-à-dire au point de coordonnées (0 ;0)

Programmation :

1) **Crée** 2 variables que tu nommeras « *c1* » et « *c2* ». Elles représenteront la taille des deux côtés de l'angle droit du triangle.

2) **Crée** une variable « *angle* ».

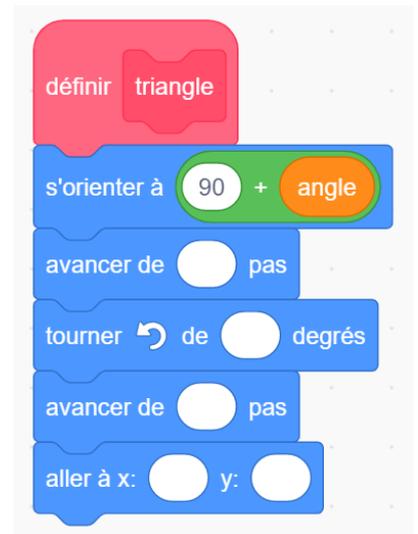
3) Dans la partie « *Mes Blocs* », **crée** un bloc que tu appelles « triangle »

4) La construction du triangle va se dérouler ainsi :

- Départ au centre de la fleur.
- Construction du 1^{er} côté de longueur « *c1* ».
- Construction de l'angle droit.
- Construction du second côté de longueur « *c2* ».
- Retour au point de départ.

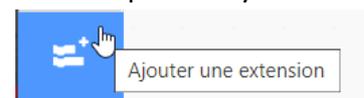
Complète les instructions ci-contre pour obtenir la construction demandée :

5) **Recopie** le script dans scratch.



6) Il s'agit maintenant de tester le script. Pour cela tu as besoin dans la bibliothèque du stylo.

Si stylo n'apparaît pas dans la bibliothèque **clique** sur « ajouter une extension » qui se trouve tout en bas à gauche de la fenêtre du logiciel.



Cherche « stylo » et **ajoute-le**.

7) **Recopie** le script suivant et **teste** le.

8) **Appelle** le professeur pour validation.



Défi n°3**Construire un pétale**Explications :

Pour construire le pétale de la fleur tu vas devoir répéter un certain nombre de fois le bloc de construction d'un triangle.

Il faudra agrandir les côtés de l'angle droit en les multipliant par 1,2.

Programmation :

1) Quelle instruction (boucle) vas-tu utiliser pour répéter la construction du bloc triangle 10 fois de suite ?

2) Pour réaliser les 10 triangles tu vas devoir respecter les consignes suivantes :

- La première valeur de c1 doit être 20.
- La première valeur de c2 doit être 10.
- Après chaque construction d'un triangle il faut agrandir les côtés comme indiqué dans les explications.

Complète les instructions suivantes à l'aide des conditions données ci-dessus.

3) **Recopie** le script dans scratch en le plaçant à la suite du script précédent à la place de l'instruction :



4) **Teste** le script. Le dessin obtenu correspond-il à ce que l'on souhaitait ?

5) Si oui tu peux continuer.

Si non **modifie** ton script.

6) **Appelle** le professeur pour validation.

Explications :

Tu as réussi à construire un pétale de la fleur.

Tu dois maintenant construire 10 pétales qui seront répartis équitablement autour du centre pour former la fleur.

Programmation :

1) Si on construit 10 pétales, quel sera l'angle en degrés entre chaque pétale pour que les 10 pétales forment un tour complet ?

2) Si on souhaitait construire 12 pétales, quel serait l'angle en degrés entre chaque pétale pour que les 12 pétales forment un tour complet ?

3) **Modifie** ton script précédent pour réaliser la construction de 10 pétales.

Tu utiliseras la boucle suivante :



A la fin de cette boucle, il faut changer l'angle de départ pour la construction pour réaliser le nouveau pétale.

Pour cela tu ajouteras et complèteras l'instruction suivante à l'endroit nécessaire :



4) **Teste** le script. Le dessin obtenu correspond-il à ce que l'on souhaitait ?

5) Si oui tu peux continuer. Si non **modifie** ton script.

6) **Appelle** le professeur pour validation.

Défi n°5**(Facultatif) Personnaliser la fleur**Explications :

Vous allez pouvoir personnaliser la fleur :

- En modifiant la taille du stylo.
- En modifiant les couleurs.
- En modifiant le nombre de pétales (plus difficile).

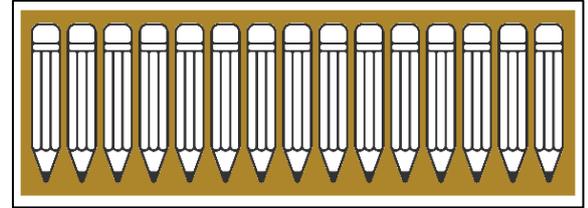
Programmation :

Modifie ton script pour essayer de réaliser les modifications précédentes.

Appelle le professeur pour validation lorsque tu as réussi.



Le jeu de Nim



Principe du jeu :

Le jeu se joue à 2 joueurs. Il y a 15 crayons disposés sur la table. Les joueurs jouent à tour de rôle. Chaque joueur peut choisir de retirer un, deux ou trois crayons. Le joueur qui retire le dernier crayon a perdu.

	Compétence évaluée	Évaluation			
1.3	Réaliser un programme simple	MTS	MS	MF	MI

Partie I : préparation

- 1) **Ouvre** le logiciel scratch 3.
- 2) **Ouvre** le fichier « projet-nim-départ.sb3 » qui se trouve sur le disque réseau.
- 3) **Clique** sur « *fichier* » puis sur « *sauvegarder sur votre ordinateur* ».
Enregistre le fichier dans ton répertoire personnel. (Tu créeras un répertoire scratch s'il n'existe pas)

4) Combien de lutins sont déjà créés ?

5) **Clique** sur le lutin crayon. Que se passera-t-il lorsqu'on cliquera sur le drapeau vert ?

6) **Clique** sur le lutin chat. Que se passera-t-il lorsqu'on cliquera sur le drapeau vert ?

7) Combien de variables sont déjà créés ?

Partie II : création des variables

Voici la liste des variables nécessaires pour le jeu :

Nom	Rôle
Nomjoueur	Enregistre le nom du joueur
Nbcrayons	Enregistre le nombre de crayons restant après chaque coup
Ordi	Enregistre le nombre de crayons que l'ordinateur retire
Réponse	Enregistre le nombre de crayons que le joueur retire

Crée dans le logiciel les variables manquantes.

Jeu de Nim : Création du jeu

Défi n°1

Création des scripts pour les crayons

Explications :

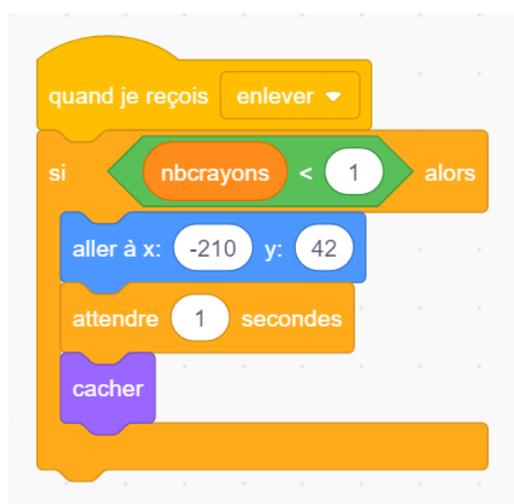
- Il faudra créer 15 crayons qui seront numérotés de 1 à 15
 - A chaque fois qu'un joueur ou que l'ordinateur retire des crayons il faudra envoyer un message à chaque crayon lui indiquant qu'on retire un ou des crayons.
 - Par exemple si on retire 2 crayons on compte le nombre de crayons restants et on efface ceux en trop. Imaginons qu'il reste 10 crayons. On en retire deux. Donc il en reste huit. Cela veut dire qu'on ne garde que les huit premiers crayons et qu'on retire le 9 et le 10.
- Donc si le nb de crayons restants est inférieur au numéro du crayon on retire le crayon

Programmation :

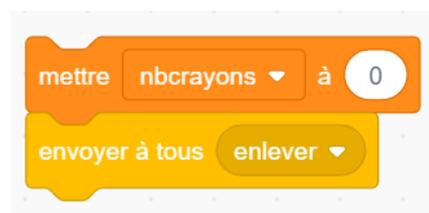
1) **Recopie** le script suivant pour le crayon n° 1.

Aide :

dans l'instruction « *quand je reçois* » il faut créer un nouveau message et le nommer « *enlever* »



2) **Ajoute** les instructions suivantes au script du lutin chat :



3) **Teste** le script en cliquant sur le drapeau vert.
Que se passe-t-il ?

.....

.....

4) **Supprime** les deux instructions que tu viens d'ajouter au script du lutin chat.
(C'était juste pour faire le test)

Jeu de Nim : Création du jeu

Défi n°2

Dupliquer les crayons

Explications :

- Il faut maintenant créer les crayons sur le modèle du crayon 1.
- Il faudra adapter le script de chaque crayon en modifiant les coordonnées du crayon pour les placer les uns à côté des autres.
- Il faudra modifier l'instruction qui indique en fonction de quel nombre de crayons, le crayon doit être caché.

Programmation :

1) **Clique** avec le bouton droit sur le lutin « *crayon1* » puis clique sur « *dupliquer* ».



2) Il faut placer le crayon 2 juste à côté du crayon 1. Pour cela on doit le décaler de 30 pixels vers la droite.

Sachant que pour le crayon 1 l'instruction est :

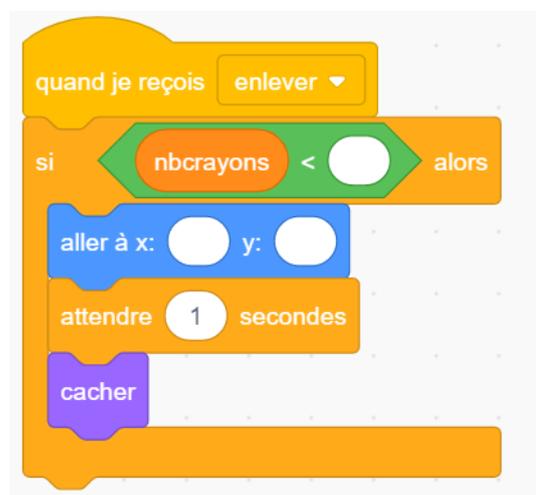


Quelles seront les valeurs pour le crayon 2 ?



3) Pour le crayon 1, on le cache si le nombre de crayons restants est inférieur à 1.

Complète les instructions suivantes pour le crayon 2 :



4) **Duplique** le crayon 2 de la même façon pour obtenir le crayon 3.

Modifie les scripts du crayon 3 comme tu as pu le faire pour le crayon 2.

5) **Continue** à dupliquer les crayons jusqu'à en obtenir 15.

Jeu de Nim : Création du jeu

Défi n°3

Script principal (tour du joueur)

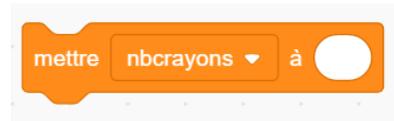
Explications :

- Il faut maintenant créer le script principal qui va demander au joueur combien de crayons il souhaite retirer.
- A chaque fois qu'il retire des crayons, il faut calculer le nombre de crayons restants.
- Tant qu'il reste au moins un crayon, on continue de demander au joueur de retirer des crayons.

Programmation :

1) Lorsqu'on clique sur le drapeau vert, il faut initialiser les variables.

Ajoute au script principal du chat l'instruction suivante en initialisant correctement le nombre de crayons.



2) Puis **ajoute** l'instruction suivante :

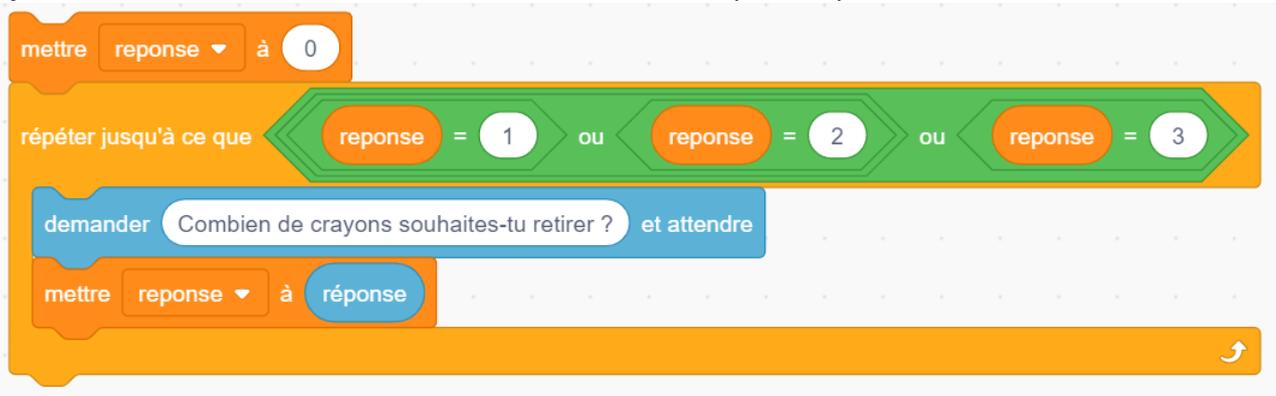


A l'intérieur de ce bloc on va demander au joueur combien de crayons il souhaite retirer.

Attention la réponse devra être 1 ou 2 ou 3.

Si le joueur répond autre chose on lui repose la question jusqu'à ce qu'il réponde correctement.

Ajoute les instructions suivantes à l'intérieur du bloc « *répéter* » précédent.



3) **Teste** le programme.

Est-ce que les crayons se retirent lorsque tu réponds 1 ou 2 ou 3 ?

Peux-tu expliquer pourquoi et ce qu'il faut rajouter :

.....

.....

.....

.....

Jeu de Nim : Création du jeu

Défi n°4 Script principal (tour du joueur) suite

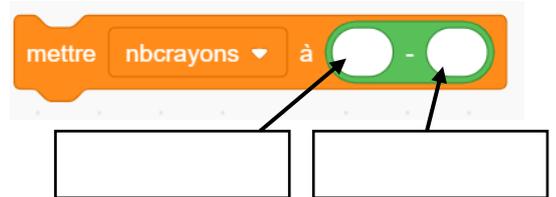
Explications :

- Dans le défi 3, nous n'avons pas calculer le nombre de crayons restants. A chaque fois que le joueur retire des crayons il faut mettre à jour la variable « *nbcrayons* » en calculant le nombre de crayons restants.
- Pour que les crayons se retirent, il faut leur envoyer le message « *enlever* » pour lancer l'instruction « *quand je reçois enlever* » dans le script de chaque crayon. Ainsi chaque crayon se mettra à jour et se cachera si nécessaire.

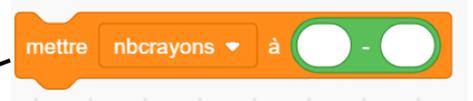
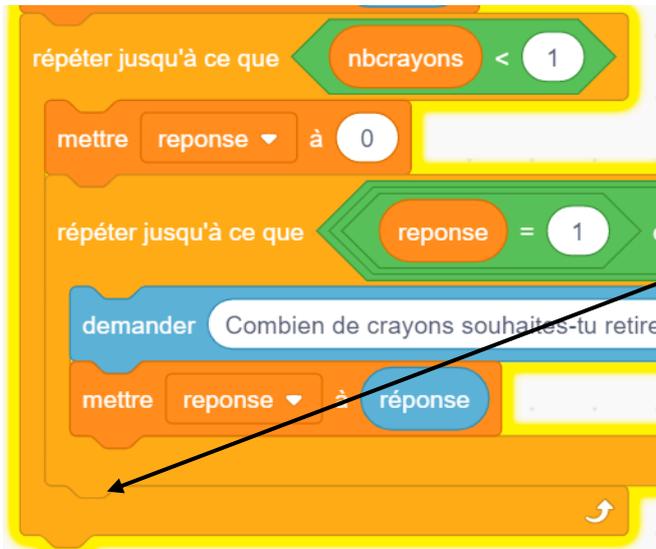
Programmation :

La suite se fait dans le script du lutin principal, le chat.

1) Pour mettre à jour le nombre de crayons restants une fois que le joueur a répondu à la question pour en retirer, par quelles variables doit-on compléter l'instruction suivante ?



2) **Appelle** le professeur pour validation. Ajoute cette instruction à la suite de la réponse donnée par le joueur. (Indiqué par la flèche ci-dessous)



3) Il faut envoyer le message « *enlever* » à tous les crayons. **Ajoute** l'instruction suivante sous l'instruction précédente.



4) **Teste** le programme.

Est-ce que les crayons se retirent lorsque tu réponds 1 ou 2 ou 3 ?

Jeu de Nim : Création du jeu

Défi n°5

Script principal (tour du joueur : fin du jeu et défaite)

Explications :

- Pour le moment il n'y a qu'un seul joueur qui joue. Lorsque le dernier crayon sera retiré par le joueur, cela signifiera qu'il aura perdu. On affichera un message et on arrêtera tous les scripts. Donc cela signifie que si le nombre de crayons **devient inférieur à 1**, alors le jeu s'arrête et c'est perdu car le joueur vient de retirer le dernier crayon.

Programmation :

La suite se fait dans le script du lutin principal, le chat.

1) Tu vas devoir ajouter les instructions suivantes à la suite de l'envoi du message « enlever »

Quelle instruction dois-tu indiquer ici ?



Appelle le professeur pour validation.

2) **Ajoute** les instructions suivantes dans le script principal à la suite de l'envoi du message « enlever ».

3) **Teste** le script jusqu'à enlever tous les crayons.

Le message de défaite s'affiche-t-il correctement à la fin ?

4) Facultatif : si tu as du temps, tu peux personnaliser le message de défaite en affichant le nom du joueur qui a perdu.

Jeu de Nim : Création du jeu

Défi n°6 Script principal (tour de l'ordinateur)

Explications :

- Pour le moment il n'y a qu'un seul joueur qui joue. Nous allons ajouter un tour pour l'ordinateur. Il jouera après le joueur.

Dans la boucle répéter il y aura donc un tour pour le joueur et un tour pour l'ordinateur.

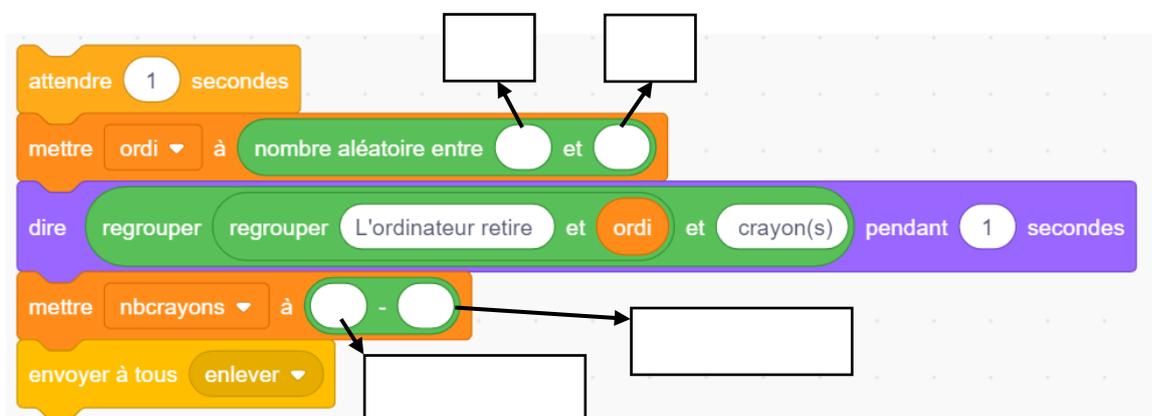
- L'ordinateur choisira un nombre au hasard entre 1 et 3 pour retirer les crayons.

Programmation :

La suite se fait dans le script du lutin principal, le chat.

1) Il faudra ajouter les instructions suivantes pour le tour de l'ordinateur.

Indique par quelles informations doivent être complétés les éléments manquants.



2) **Appelle** le professeur pour validation.

3) **Ajoute** les instructions ci-dessus à l'endroit suivant :
(après le tour du joueur)



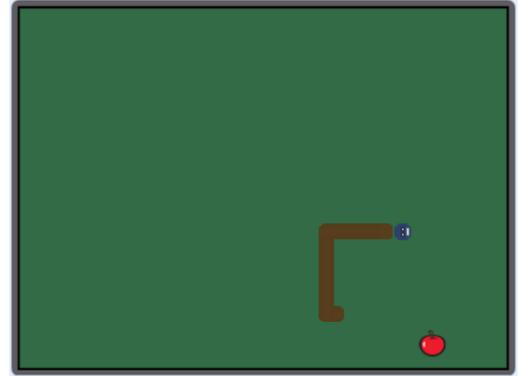
4) A la suite des instructions que tu viens de créer (pour le tour de l'ordinateur) il faut vérifier si l'ordinateur vient de retirer le dernier crayon.

En prenant modèle sur ce qui a été fait pour le joueur lorsqu'il retire le dernier crayon, faire de même pour l'ordinateur.

Attention le message sera cette fois-ci la victoire du joueur.



Le jeu du Serpent



Principe du jeu :

Vous dirigez un serpent sur l'aire de jeu. Le but est de manger un maximum de pommes. Attention si vous vous mordez la queue vous perdez. Si vous touchez le bord de l'écran vous perdez également.

	Compétence évaluée	Évaluation			
1.3	Réaliser un programme simple	MTS	MS	MF	MI

Partie I : préparation

- 1) Ouvre le logiciel Scratch.
- 2) Dans le logiciel scratch, **clique** sur « fichier » puis « importer » et sélectionne le fichier « projet-serpent.sb3 » qui se trouve dans le répertoire « commun » du réseau.
- 3) Combien de lutins sont déjà créés ?
- 4) Combien y a t-i d'arrière plans différents ?
- 5) D'après toi, à quoi servira l'arrière plan n°2 ?

Jeu du serpent : Création du jeu

Défi n°1	Placement aléatoire de la pomme en début de jeu	<input type="checkbox"/>
<u>Explications :</u> - Au début du jeu, la pomme doit se placer aléatoirement à un endroit de l'espace de jeu. - Il y aura besoin de 4 variables qui enregistreront le score, le meilleur score, la vitesse et la taille du serpent.		
<u>Programmation :</u> 1) Sélectionne le lutin « pomme » 2) En utilisant les instructions ci-dessous, crée un script qui permet de : <ul style="list-style-type: none">- Lorsqu'on clique sur le drapeau vert la taille de la pomme se met à 40 %- Puis la pomme se place à un endroit au hasard sur l'aire de jeu avec x compris entre -210 et 210 et y compris entre -155 et 155		
<div style="display: flex; justify-content: space-around;"><div style="border: 1px solid green; border-radius: 15px; padding: 5px; background-color: #90EE90;">nombre aléatoire entre <input type="text"/> et <input type="text"/></div><div style="border: 1px solid purple; border-radius: 15px; padding: 5px; background-color: #8A2BE2;">mettre la taille à <input type="text"/> % de la taille initiale</div><div style="border: 1px solid blue; border-radius: 15px; padding: 5px; background-color: #4169E1;">aller à x: <input type="text"/> y: <input type="text"/></div></div>		
3) Crée 4 variables que tu appelleras « score », « highscore », « vitesse » et « longueur »		
4) Ajoute à la suite de ton script une boucle « répéter indéfiniment »		
5) Appelle le professeur pour validation.		

Défi n°2**Placement aléatoire de la pomme en cours de jeu**Explications :

- A chaque fois que la tête du serpent touchera la pomme, cela signifiera qu'il la mange.
- A chaque fois qu'elle est mangée par le serpent, elle doit disparaître et une nouvelle pomme apparaît aléatoirement à un endroit.

Programmation :

- 1) Dans la boucle « *répéter indéfiniment* », **ajoute** l'instruction ci-contre :



- 2) Dans ce bloc d'instruction « *si ... alors* », **ajoute** les instructions nécessaires pour :

- Ajouter 1 à la variable score et 1 à la variable highscore.
- Ajouter 0,2 à variable longueur.
- Aller à une position aléatoire comme celle donnée en début de script.

- 3) Que se passera-t-il pour la pomme lorsqu'elle sera touchée par la tête du serpent ?

- 3) **Appelle** le professeur pour validation.

Défi n°3**Faire avancer le serpent automatiquement**Explications :

- Dès le démarrage du jeu le serpent doit avancer.
- Au début du jeu, la longueur du serpent doit être la plus petite possible.
- Au début du jeu le score doit être initialisé à 0.
- La vitesse doit également être initialisée à une valeur assez petite.

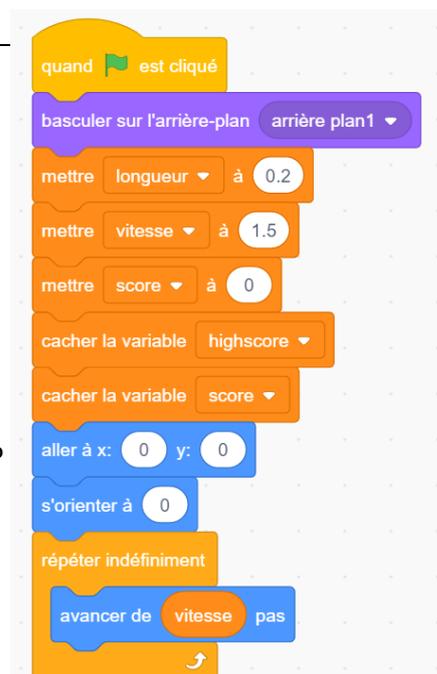
Programmation :

- 1) **Sélectionne** le lutin « *tête* »

- 2) **Recopie** le script ci-contre :

- 3) **Teste** le script. **Décris** ce qu'il se passe :

- 4) Que faudra-t-il ajouter comme mouvement au serpent maintenant ?



Défi n°4**Déplacement de la tête du serpent**Explications :

- Lorsque que le joueur appuie sur la touche « *flèche droite* » du clavier la tête du serpent doit s'orienter vers la droite pour aller à droite.
- Il faudra créer les instructions pour les touches « *flèche gauche* », « *flèche haut* » et « *flèche bas* ».

Programmation :

1) **Sélectionne** le lutin « *tête* »

2) **Recopie** le script ci-contre :



3) **Complète** le script par 3 autres blocs d'instructions pour les flèches « *haut* », « *bas* » et « *gauche* ».

4) **Teste** le script.

Défi n°6**Positionner le corps du serpent**Explications :

La queue du serpent doit se positionner derrière la tête. La queue doit suivre le déplacement de la tête. A chaque fois que le serpent mange une pomme il faut agrandir sa queue.

Programmation :

1) **Sélectionne** le lutin serpent.

2) **Recopie** le script ci-contre et choisis parmi les 4 possibilités données ou doit aller (se positionner) la queue du serpent.

2) **Teste** le script.

Est-ce que le corps du serpent suit la tête ?

Si non, modifie ton script.

Si oui continue les questions.

3) Est-ce que le corps du serpent grandit lorsqu'il touche une pomme ?



Défi n°7**Augmenter la taille du corps du serpent**Explications :

A chaque fois que le serpent mange une pomme il faut agrandir sa queue.

Programmation :

1) **Sélectionne** le lutin serpent.

2) **Recopie** le script ci-contre.

2) **Teste** le script. Que fais ce script ?



3) Est-ce que le serpent grandit que lorsqu'il mange une pomme ?

4) Il faut maintenant que le serpent que lorsque le serpent mange une pomme et qu'il ne grandisse pas entre temps. Pour cela recopie le script suivant :

Ce script permet d'effacer le reste de la queue lorsque le serpent a atteint sa taille. (Il ne grandit pas lorsqu'il ne mange pas de pommes)

Lorsqu'il mange une pomme, on augmente la variable « longueur » ce qui explique que le serpent va grandir car on attend un peu plus longtemps avant d'effacer le bout de son corps.



5) **Teste** le script.

6) Est-ce que le serpent grandit que lorsqu'il mange une pomme ?

Défi n°8**Arrêt du jeu en cas de défaite**Explications :

- Si la tête du serpent touche le bord de l'écran le jeu doit s'arrêter.
 - Si la tête du serpent touche sa queue le jeu doit s'arrêter.
- Dans ces cas, il faut afficher le score et la fin de partie.

Programmation :

1) **Sélectionne** le lutin « tête ».

2) Dans la boucle « *répéter indéfiniment* », après « *avancer de vitesse pas* » **ajoute** l'instruction suivante :



Les couleurs à choisir sont :

- Le marron pour indiquer que la tête a touché la queue marron.
- Le noir pour indiquer que la tête a touché le bord de la surface de jeu.

3) A l'intérieur de la boucle « si...alors » **ajoute** des instructions pour que :

- l'arrière bascule sur l'arrière-plan n°2 qui indique la fin du jeu.
- afficher les variables « *score* » et « *highscore* ».

(tu placeras les variables scores sur l'arrière-plan, où tu veux)

- stopper tous les scripts.

4) Tu peux maintenant profiter de ton jeu qui est terminé.

Pense à l'enregistrer dans ton répertoire personnel.

Défi n°9**Facultatif : ajouter d'un lutin piège**Explications :

Tu peux ajouter un petit lutin qui sera un piège.

Le serpent devra l'éviter sinon la partie s'arrêtera.

Ce lutin piège apparaîtra de façon aléatoire à un endroit de la surface de jeu.

Il restera à la même place durant 3 secondes puis changera de place.

Programmation :

1) **Choisis** un nouveau lutin qui sera un lutin piège.

2) **Mets** le à une taille proche de celle de la pomme.

3) **Programme** les instructions nécessaires pour répondre aux explications données ci-dessus.

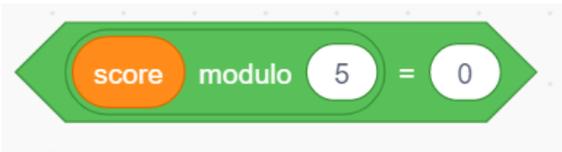
4) **Appelle** le professeur pour validation lorsque tu as terminé.

Explications :

- Tu peux augmenter légèrement la vitesse toutes les 5 pommes mangées (ou 10 pommes ou tout autre choix)
- A la fin du jeu, lors de l'affichage des scores, on peut cacher tous les lutins.

Programmation :

- 1) **Sélectionne** le lutin « pomme ».
- 2) Nous allons rajouter une instruction de contrôle qui va vérifier si le serpent a mangé 5 pommes. Pour cela nous allons utiliser la condition suivante :



Cette condition teste la division du nombre score par 5. Si le reste est égal à 0, cela signifie que le nombre score est un multiple de 5, donc que le serpent vient de manger 5 pommes.

A la fin de la boucle « *si touche la tête ? alors* » **ajoute** une boucle « *si ... alors* » avec la condition précédente.

Puis dans cette boucle, augmente la vitesse (par exemple de 0,5)

- 3) Lors de la fin de la partie, nous allons cacher tous les lutins. Pour cela dès que la partie est finie on envoie un message « perdu » à tous les lutins.

Détermine à quel endroit tu dois ajouter l'instruction suivante et **fais-le** :



Dans le script de chaque lutin, lorsqu'il reçoit le message perdu on le cache.

Ajoute les instructions correspondantes pour chaque lutin.

- 4) **Appelle** le professeur pour validation.



Dessiner une rosace

But du script :

L'objectif de ce TP est de réaliser une rosace.

Le travail n'est pas guidé : ce sera à toi de construire le script qui permettra de construire la rosace. Pour cela tu pourras t'aider du TP précédent avec la fleur géométrique

	Compétence évaluée	Évaluation			
1.3	Réaliser un programme simple	MTS	MS	MF	MI

Partie I : préparation

1) Ouvre le logiciel scratch 3.

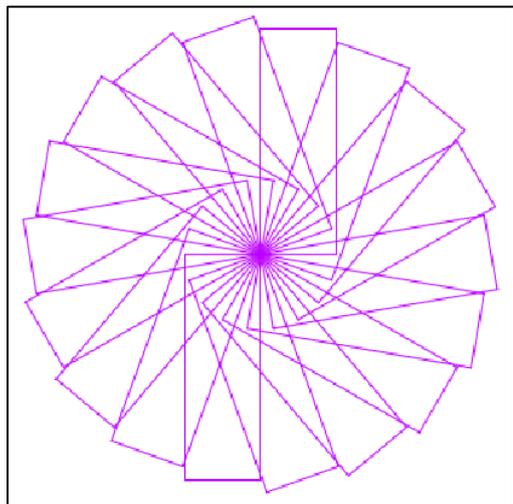
2) Enregistre le fichier dans ton répertoire personnel (dans un répertoire scratch que tu as déjà créé)

Donne comme nom à ton fichier : « *projet-scratch-rosace.sb3* »

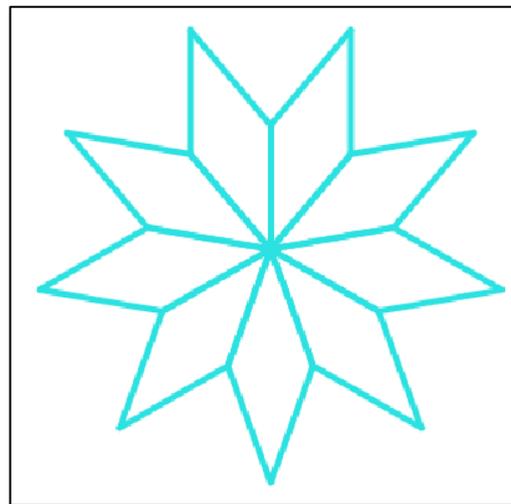
Objectifs à atteindre :

Tu vas devoir essayer de réaliser la construction d'une des deux rosaces ci-dessous :

Rosace n°1 (plus simple)



Rosace n°2 (plus difficile)



Aides

Explications :

- Pour la rosace n°1, la figure de base est un rectangle.
- Pour la rosace n°2, la figure de base est un losange (angles de 40° et de 140°)
- Pour chaque rosace, tu auras besoin d'une variable angle qui sera l'angle d'orientation pour le dessin du rectangle ou du losange.
- Tu auras besoin de créer un bloc : ce bloc construira un rectangle ou un losange.
- A toi de voir combien de fois tu devras répéter la construction de ce bloc et dans avec quel angle tu t'orienteras.

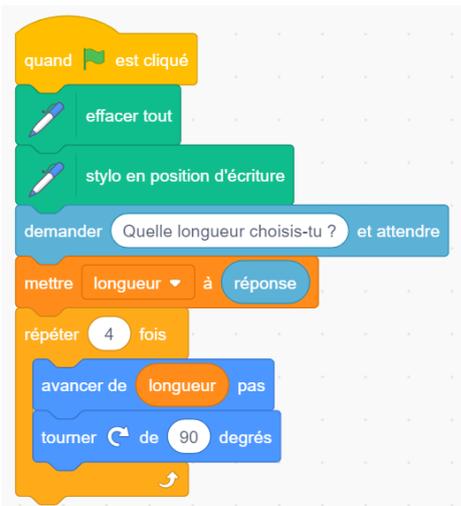
Exercices d'entraînement



Ces exercices que tu feras toute l'année dans ton cahier vont te permettre de progresser en programmation et dans l'utilisation de scratch. Ils te permettront d'être prêt pour le brevet car il s'agira d'exercices de ce type que tu retrouveras à l'épreuve de maths du DNB.

Exercice 1 :

Que va réaliser le programme scratch ci-dessous lorsqu'on cliquera sur le drapeau vert ?



Exercice 2 :

1) Quelle figure dessine le script suivant ?



2) Calcule les dimensions de chaque côté de la figure.

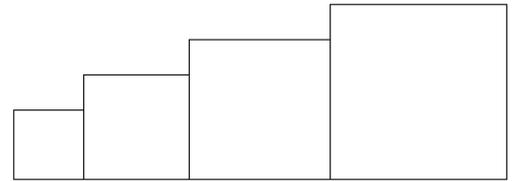
Exercice 3 :

Simon travaille sur un programme. Voici des copies de son écran :

Script principal	Bloc Carré
	<p>Information</p> <p>L'instruction signifie qu'on se dirige vers la droite.</p>

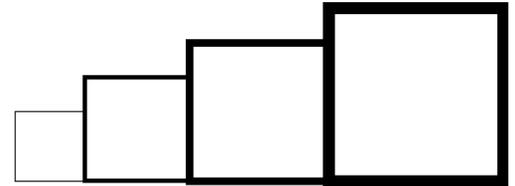
1) Il obtient le dessin ci-contre.

- D'après le script principal, quelle est la longueur du côté du plus petit carré dessiné ?
- D'après le script principal, quelle est la longueur du côté du plus grand carré dessiné ?



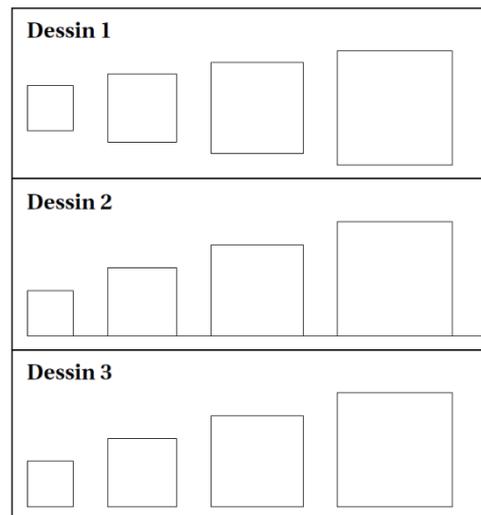
2) Dans le script principal, où peut-on insérer l'instruction ci-dessous de façon à obtenir le dessin ci-contre ?

ajouter 2 à la taille du stylo



3) On modifie maintenant le script principal pour obtenir celui qui est présenté ci-contre :

Parmi les dessins ci-contre, lequel obtient-on ?



```

quand [drapeau] est cliqué
aller à x : -200 y : 0
s'orienter à 90
effacer tout
mettre la taille du stylo à 1
mettre côté à 40
répéter 4 fois
  carré
  avancer de côté + 30
  ajouter à côté 20
  
```

Exercice 4 :

Voici un script saisi par Alice dans un logiciel d'algorithmique.

```

quand [drapeau] est cliqué
demander "Choisissez un nombre ?" et attendre
envoyer à tous "le nombre a été saisi"
mettre Nombre à réponse
mettre Résultat 1 à 2 * Nombre + 3
mettre Résultat 1 à Résultat 1 * Résultat 1
dire [regroupe] "le résultat 1 est Résultat 1" pendant 2 secondes
  
```

```

quand je reçois "le nombre a été saisi"
mettre Résultat 2 à Nombre * Nombre
mettre Résultat 2 à Résultat 2 * 4
mettre Résultat 2 à Résultat 2 + 12 * Nombre
mettre Résultat 2 à Résultat 2 + 9
attendre 3 seconde
dire [regroupe] "le résultat 2 est Résultat 2"
  
```

1) Alice a choisi 3 comme nombre, **calcule** les valeurs de Résultat 1 et de Résultat 2.

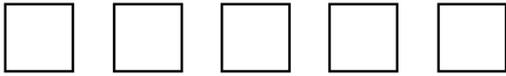
Justifier en faisant apparaître les calculs réalisés.

Modélisation :

- En appelant x le nombre choisi dans l'algorithme, donner une expression littérale traduisant la première partie de l'algorithme correspondant à Résultat 1.
- En appelant x le nombre choisi dans l'algorithme, donner une expression littérale traduisant la deuxième partie de l'algorithme correspondant à Résultat 2.

Exercice 5 :

Léna et Youri travaillent sur un programme. Ils ont obtenu le dessin suivant :



Ils ont ensuite effacé une donnée par erreur dans le script principal. Voici les copies d'écran de leur travail :

Programme		Pour information
<p>Script principal</p>	<p>Bloc du motif</p>	<p>L'instruction</p> <p>signifie qu'on se dirige vers la droite.</p>

Valeur effacée

- 1) Quelle était la valeur effacée dans le script principal ? (20, 40, 60 ou 80)
- 2) **Dessine** sur ton cahier ce qu'on aurait obtenu avec 40.
On représentera l'instruction « avancer de 20 » par un segment de longueur 1 cm.

3) Léna et Youri souhaitent maintenant obtenir un triangle équilatéral comme motif.

Par quelles valeurs doit-on remplacer les lettres a, b, c ?

Exercice 6 :

On donne le programme suivant qui traduit un programme de calcul :

- 1) Si on choisit le nombre 8 au départ, quel sera le résultat ?
- 2) Ecrire en français, sur votre cahier, le programme de calcul correspondant à ce script :
 - Choisir un nombre.
 -

3) Si on choisit comme nombre de départ x , donne l'expression littérale correspondant à ce programme de calcul.

4) Quel nombre doit-on choisir au départ pour obtenir comme résultat 6 ?