

---

# Scratch au collège

**Introduction du code : un apport pour tous les élèves.**

vincent.huvelle@ac-grenoble.fr - 2014-2015

---

**Projet réalisé dans le cadre des "heures numériques" de l'académie de Grenoble.**

---

Des documents et les programmes Scratch se trouvent sur le site Planète maths :  
[lien](#)<sup>1</sup>

---



---

<sup>1</sup> <http://www.ac-grenoble.fr/disciplines/maths/pages/PM/Affichage/Recherche.php?faire=voir&ChoixNumero=470>

---

## Sommaire

1. Introduction
2. L'outil Scratch
3. Le projet
4. La progression adoptée
5. Bilan

---

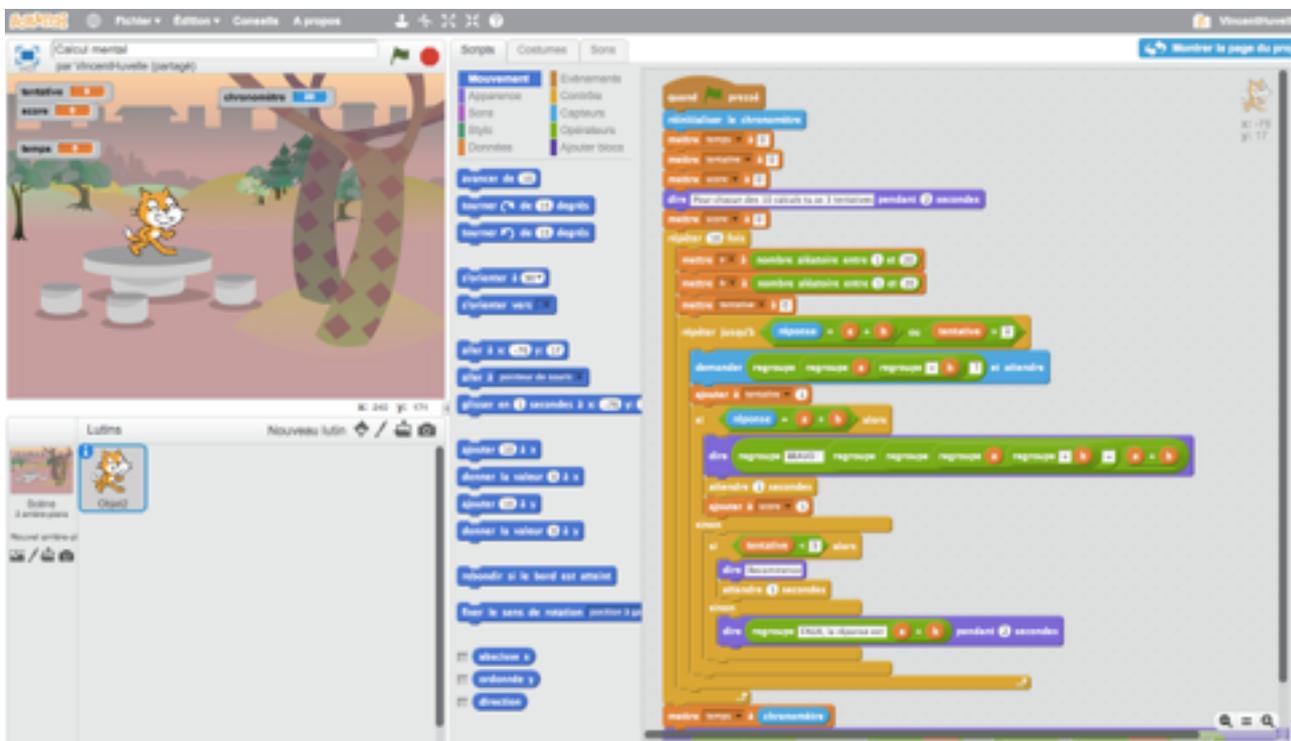
## Introduction

J'ai toujours pensé que la programmation était propice à une attitude de recherche, qu'elle pouvait aider à structurer sa pensée, guider sa réflexion, travailler sa logique.

Je suis également convaincu, comme **Seymour Papert** l'explique dans *le jaillissement de l'esprit*, de l'importance de l'affectif dans les apprentissages : un projet, un exposé va permettre à l'élève de s'investir totalement.

Malheureusement la rigueur d'un langage de programmation<sup>2</sup> est un vrai handicap pour de nombreux élèves. Mon but n'étant pas de faire un club informatique pour quelques élèves mais bien d'initier tous les élèves et notamment ceux en difficultés.

Et c'est là que **Scratch** apparaît



---

<sup>2</sup> Même des plus « simple » comme le python.

---

## L'outil Scratch

Scratch est un environnement de programmation visuel adapté pour les plus jeunes.

Il a été développé par un groupe de recherche du laboratoire Média du MIT (Massachusetts Institute of Technology, USA) dans le cadre du développement de l'apprentissage du code informatique dès le plus jeune âge.

La programmation se fait en assemblant des blocs, ce qui permet de se dégager des contraintes d'écriture d'un langage de programmation.

Malgré cela, on peut travailler rapidement tous les concepts de programmation et créer des programmes attirants pour les élèves.

On peut utiliser Scratch en ligne : <https://scratch.mit.edu> ou télécharger l'application <https://scratch.mit.edu/scratch2download/>

À noter qu'on peut créer un compte sur le site de [scratch.mit.edu](https://scratch.mit.edu), créer des projets partagés, des studios, commenter différents projets.

---

# Le projet

## Mise en place

- \* Une classe de cinquième.
- \* Une heure par semaine<sup>3</sup> (J'ai une demi-heure de plus avec eux dans leur emploi du temps).
- \* Co-animation sur certaines séances avec un professeur de technologie.
- \* Séance en salle informatique, dans laquelle nous disposons également de tables pour être en configuration « classique ».
- \* Travail en groupe de 2.

## Forme du projet

- \* Une progression sur plusieurs TP pour comprendre les concepts de bases de la programmation.
- \* Un projet « mathématique » sur la constructibilité des triangles.
- \* La création d'un jeu dans la dernière période.

## Apports pour les élèves

- \* Faire des mathématiques autrement.
- \* Développer la rigueur, la logique.
- \* Travailler l'esprit de synthèse.<sup>4</sup>
- \* Avoir de l'estime de soi. Ce ne sont pas forcément les mêmes élèves qui réussissent. De plus en programmation, ce n'est pas tout bon ou tout faux, on débogue petit à petit.
- \* On peut faire de la pédagogie différenciée. On part d'une tâche simple qui peut être généralisée pour les meilleurs.

---

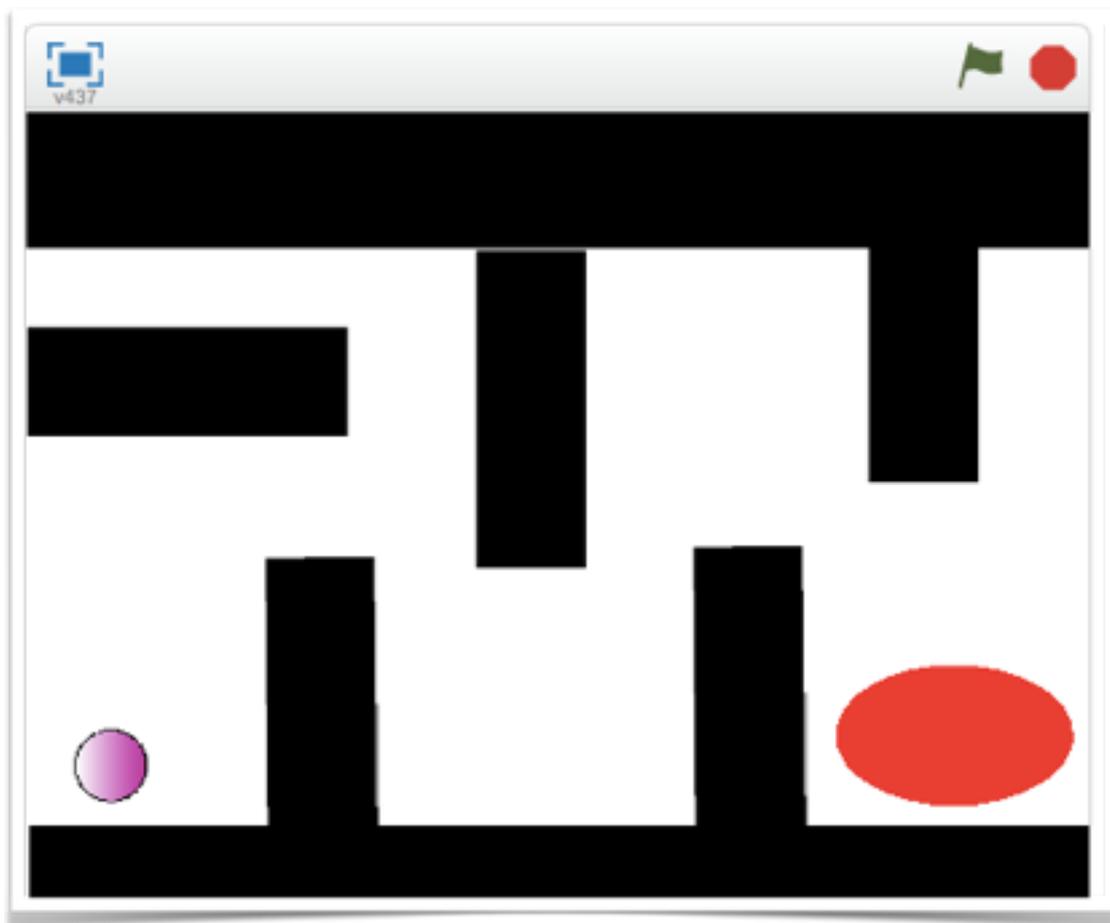
<sup>3</sup> Au maximum, cette heure peut être une heure de cours de mathématiques ou être remplacée pour d'autres projets.

<sup>4</sup> Rédiger une notice est un excellent exercice.

---

## La progression adoptée

### TP1 : Découverte de l'interface. Création d'un labyrinthe.



- \* Notions de lutins (scripts et costumes) et de scène (arrière plan).
- \* Exploration des menus **Mouvement** et **Événements** afin de déplacer le lutin à l'aide des touches du clavier.
- \* Utilisation de la palette graphique pour dessiner un nouveau lutin et des obstacles pour le labyrinthe.
- \* Notion de repérage pour les objets et la scène.
- \* Exploration des menus **Capteurs** et **Apparences** pour les obstacles et la fin du labyrinthe.
- \* Création d'un compte en ligne pour chaque élève et d'un studio « labyrinthe » afin qu'ils puissent tester les labyrinthes de leurs camarades et même de les améliorer.



---

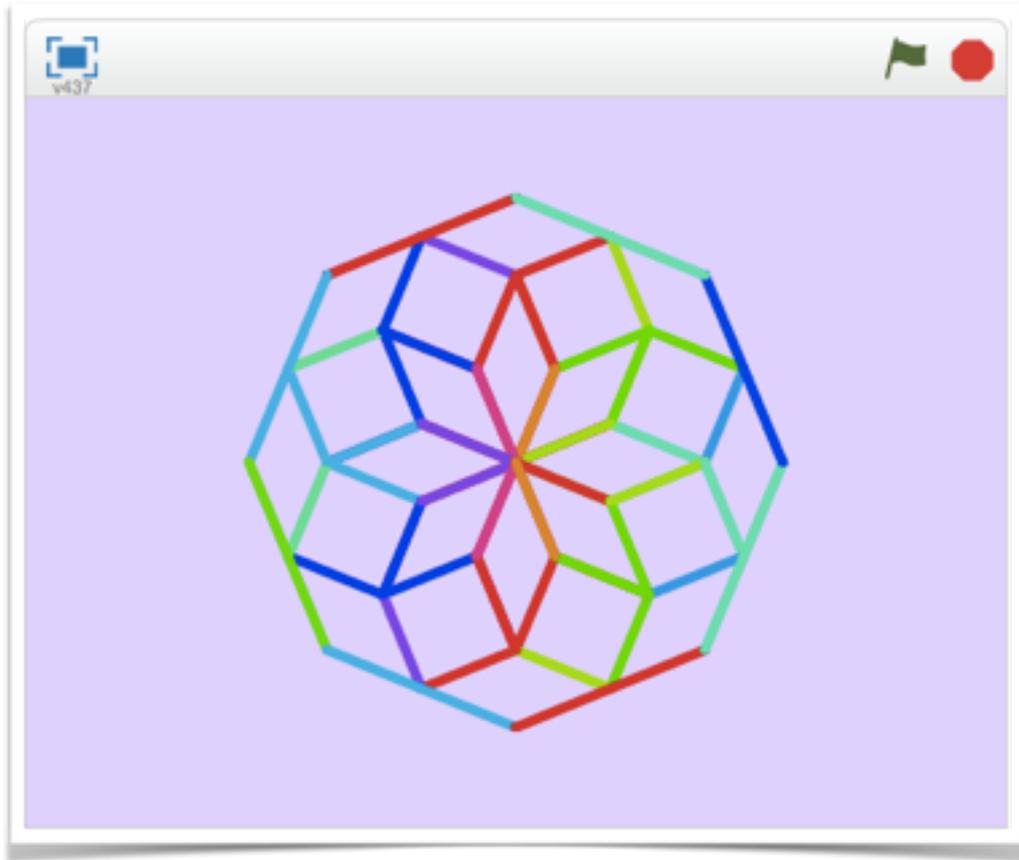
## TP3 : Geotortue

L'IREM de Paris a fait un travail remarquable de fiches pour Geotortue :

[http://www-irem.univ-paris13.fr/site\\_spip/spip.php?article3](http://www-irem.univ-paris13.fr/site_spip/spip.php?article3)

On peut donc en profiter pleinement avec Scratch !

Je présente la première figure sous forme de défi :



- \* Distribution de la figure.
- \* Analyse sur table de la figure, premières idées de construction qu'on partage.
- \* Création d'un script.

Remarque : Pour aider les élèves qui sont plus en difficultés, je dépose des fichiers « étapes » du script<sup>6</sup> dans leur espace personnel sur le réseau. De plus, les élèves qui réussissent sont souvent motivés pour apporter de l'aide à leurs camarades.

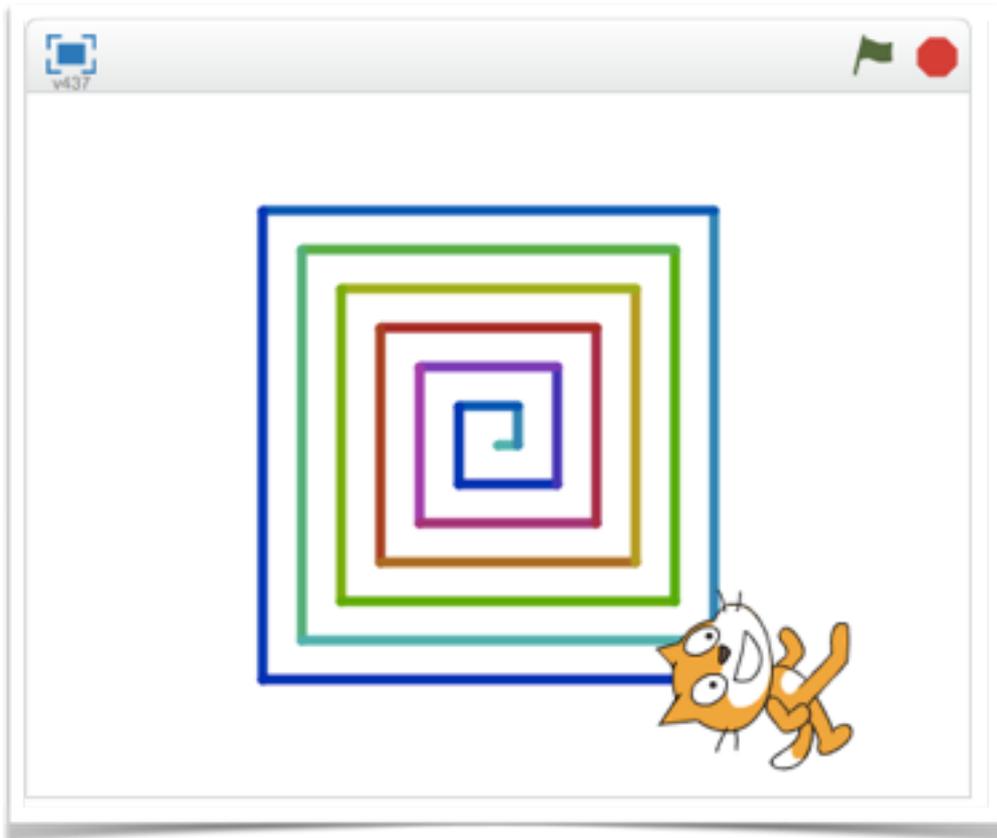
---

<sup>6</sup>Que je peux volontairement bugger pour permettre à l'élève d'obtenir la satisfaction du résultat !

---

## TP4 : Introduction de variables. Création de spirales.

On continue à utiliser les possibilités du stylo et de la notion de défi en proposant la création de spirales :



- \* Analyse de la construction d'une spirale à l'aide d'une construction « papier ».
- \* Notion de variables, analyse du menu **Données**.
- \* Défi : échanger 2 variables.

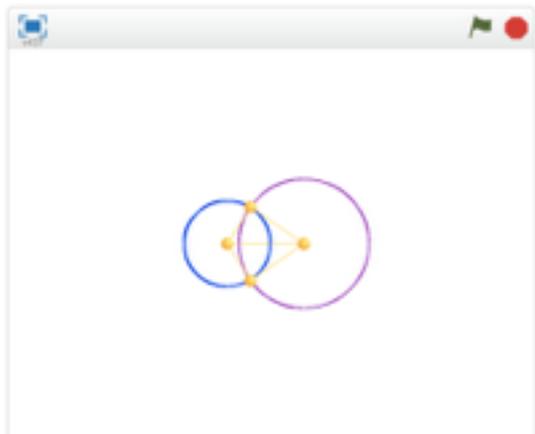
## TP5 : Création d'une feuille de calcul mental. Découverte des nombres aléatoires et des opérateurs logiques.



- \* Je présente l'objectif aux élèves, on analyse les besoins pour le script.
- \* Découverte des menus **Contrôles** et **Opérateur**.
- \* Les plus avancés ont de nombreuses possibilités de généralisations : score, chronomètre...

---

## Projet : Constructibilité d'un triangle.



Les élèves sont maintenant prêts pour leur premier projet.

En résumé, la page d'accueil affiche 2 lutins qui donnent le choix entre le critère somme des angles ou longueurs des côtés.

L'utilisateur entre les informations, les élèves créent les scripts qui analysent et qui déterminent la constructibilité ou non du triangle (la partie longueurs est très riche).

À chaque construction possible, les élèves tracent le triangle.

- \* Demander la valeur des 3 angles du triangle (répéter jusqu'à ce que la somme soit égale à 180 degrés).

- \* Construction du triangle : étant donnés 3 angles dont la somme de leur mesure est 180 degrés, construire le triangle.

Pour cela, on introduit la notion de liste puis de tri, afin de classer les angles dans l'ordre croissant (on utilisera les 2 angles aigus pour la construction).

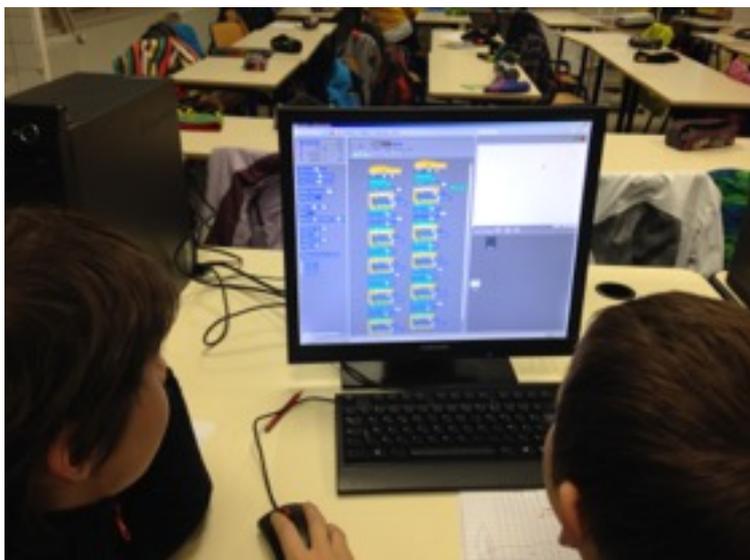
- \* Longueur des côtés : comparaison de la longueur du plus long côté avec la somme des deux autres : décision de la constructibilité du triangle.

- \* Construction d'un triangle connaissant la longueur des côtés...

- \* Travail sur l'ergonomie et le graphisme de l'application.

---

## Un premier bilan



Très positif !

Scratch est devenu un compagnon du quotidien qui permet d'illustrer, d'expliquer, d'approfondir beaucoup de notions qui restaient jusqu'alors « abstraites ».

Par exemple la notion de variables, ou encore la logique : **si condition alors conclusion...**

Mais le plus important reste :

- \* Les élèves cherchent, ils sont motivés.
- \* Les élèves n'ont plus d'appréhension pour certaines notions mathématiques.
- \* Les élèves se sont appropriés l'environnement et les concepts de base.
- \* Les élèves créent, apportent une touche personnelle aux problèmes.
- \* Il y a des échanges entre les élèves sur leurs projets.
- \* La relation élève enseignant est complètement différente, c'est l'ordinateur qui « déclare » que le programme fonctionne ou pas ; le professeur est alors là pour aider, guider mais ne juge pas, il encourage.
- \* Des élèves en échec depuis longtemps en mathématiques parviennent à utiliser des notions mathématiques et réussissent de belles choses.