

# ScratchMaths

# Maths&Scratch

## Présentation du projet

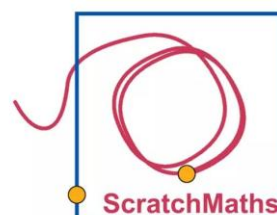


**UCL Institute of Education**

Une équipe d'enseignants et de spécialistes en pédagogie des mathématiques de l'UCL (University collège London) a développé un programme ambitieux et passionnant. Son but : exercer les compétences mathématiques travaillées en fin de cycle primaire avec Scratch.

Le programme prévoit 6 modules à travailler sur deux ans avec des enfants de 9 et 10 ans, l'équivalent de nos 4<sup>ième</sup>, 5<sup>ième</sup> et 6<sup>ième</sup> primaire et ce, en une vingtaine d'heures chaque année.

Si vous avez quelques notions d'anglais, n'hésitez pas à y jeter un coup d'œil. Vous pourrez apprécier l'énorme travail qui a été réalisé.



Présentation du projet et des ressources :

<http://www.ucl.ac.uk/scratchmaths>

Ce diaporama vous donne un aperçu du programme :

[https://www.ucl.ac.uk/ioe/sites/ioe/files/ScratchMaths\\_Curriculum\\_Overview.pdf](https://www.ucl.ac.uk/ioe/sites/ioe/files/ScratchMaths_Curriculum_Overview.pdf)

L'ensemble des ressources peuvent être téléchargées ici :

<https://discovery.ucl.ac.uk/id/eprint/10204339/1/UCL%20SM%203.0.zip>

Que trouve-t-on dans ces ressources ?

- **Teacher materials** (PDF) : le document qui explique en détails à l'enseignant comment procéder. Entre 50 et 90 pages pour chacun des 6 modules !
- **Classroom presentation** (PDF) : les diapositives à projeter en classe et sur base des quelles l'enseignant met ses élèves au travail
- **Les fichiers Scratch au format SB3**. L'élève charge ces fichiers dans lesquels des blocs et des sprites sont prêts à l'emploi. Parfois, une courte vidéo de présentation.
- **Des posters à afficher en classe** et des listes de vocabulaire.

## Traduire ou adapter ?

J'étais parti sur l'idée de traduire cette ressource. Elle présente en effet un objectif auquel je souscris totalement : mettre la programmation et le codage au service des apprentissages en Mathématiques.

Après mûre réflexion, j'ai pris la décision de proposer une adaptation plutôt qu'une traduction tout en restant fidèle à la teneur des exercices, à la planification et à la méthodologie.

La dénomination **ScratchMaths** devient **Maths & Scratch**.

Pour la présentation des exercices aux élèves, l'équipe de l'UCL propose des diaporamas qui mettent l'enseignant en position de devoir « rattraper » régulièrement son groupe pour présenter les challenges les uns après les autres, faire le point ou expliquer différentes notions.

Je préfère la méthode des cartes de programmation (voir nos rubriques Scratch+ et FMTTN) qui permet aux élèves de progresser chacun à leur rythme, en autonomie encadrée et à faire en sorte que l'enseignant sois moins souvent au tableau et plus souvent aux côtés des apprenants.

J'ai travaillé le premier module (sur 6....) afin de pouvoir recueillir vos impressions, vos remarques et suggestions, et in fine, évaluer si cela vaut la peine de continuer... c'est un gros boulot, heureusement que je suis à la retraite...

### **Les cartes**

- Un jeu par élève ou par groupe de deux élèves.
- À imprimer en A5 sur papier fort
- Elles restent en classe et serviront plusieurs fois
- Elles présentent les tâches aux élèves en donnant un minimum d'explications. En principe et si ils lisent attentivement, ils peuvent avancer en autonomie...

Important ! **Les cartes n'ont pas vocation à apprendre à utiliser Scratch !** Il revient à l'enseignant de présenter Scratch à ses élèves, les différents espaces de travail et la terminologie propre à Scratch. Une petite démonstration de 10-15 minutes avant de commencer et des mises au point de temps en temps sont à prévoir.

### **Le carnet de l'élève**

- Peut être imprimé sur simples feuilles A4 en couleur ou en nuances de gris.
- Contient :
  - o Des exercices en mode « déconnecté ». Ils peuvent être utilisés comme tâches à faire à domicile ou en classe
  - o Les définitions des nouveaux termes et des blocs Scratch
  - o Une check-list des savoir-faire exercés

### **Le carnet de l'enseignant (à construire...)**





La traduction des « teacher materials » de ScratchMaths n'est pas prévue. Ce serait très fastidieux et je ne suis pas sûr que les enseignants sont prêts à potasser des modes d'emploi de 50 à 100 pages pour chaque module....

L'enseignant devrait d'abord s'appuyer sur le carnet de l'élève et les cartes, s'en imprégner et les tester.

Si vous pensez qu'un carnet pour l'enseignant serait néanmoins utile, je vous propose de me dire ce que vous aimeriez y trouver...

## Module 1,

Les 4 séquences

1. Déplacer, tourner et estampiller 	2. Répéter, alterner 	3. Rosaces et motifs circulaires 	4. Travailler avec des blocs personnalisés 
--	---	--	---

## Terminologie et concepts couverts par le module 1

Scratch	Programmation	Mathématiques
Sprite Scène Bloc Bloc <b>estampiller</b> Bloc de tête (bloc événement) Bloc <b>Tourner de ...</b> Script – assemblage de blocs Bloc <b>Avancer de ...</b> Bloc <b>répéter ... fois</b> (boucle) Changement de <b>Costumes</b> Blocs personnalisés	Instruction Programme, programmation Procédure Répétition ou boucle Algorithme Script événement	Opérations fondamentales Symétries Translations Angles Rotation Transformation Nombres positifs et négatifs  Et un tout premier contact avec les Coordonnées dans un repère X Y

## Durée recommandée : 4 x 2 périodes (minimum...)

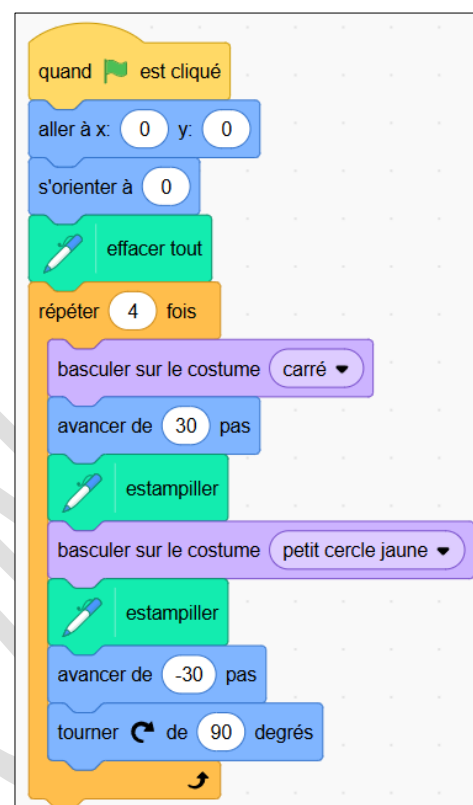
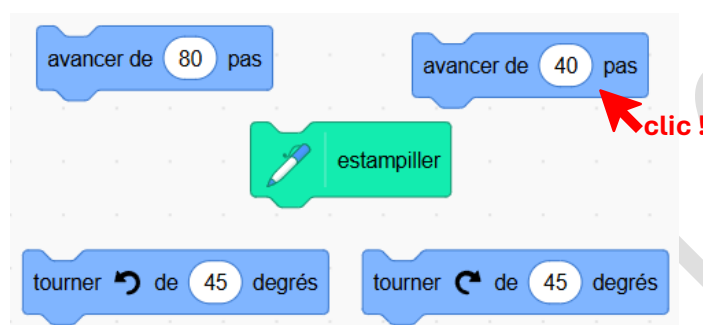
À prendre sur les périodes de Mathématiques et de FMTTN.

À mon avis, il en faudra un peu plus...

## La thématique

Ce module propose aux élèves de dessiner des rosaces en imprimant des motifs simples à l'aide du bloc **estampiller**.

On commence avec les blocs de Scratch utilisés comme des outils actionnés un par un pour arriver en fin de module à concevoir de vrais programmes qui font le job avec un seul clic de souris.



## FMTTN ???

Est-ce que ces séquences couvrent les attendus de la FMTTN en termes de programmation et d'algorithmique ?

Rien que le premier module les couvre largement ! Pour être tout à fait en phase avec le référentiel, il ne manque que deux ou trois logigrammes. Les programmes qui seront développés dans le cadre de ces activités s'y prêtent très bien. Mais il faut bien reconnaître que ces logigrammes risquent de ressembler comme deux gouttes d'eau aux enchaînements de blocs Scratch...

## Quelques remarques sur la méthodologie

### Tester et essayer les blocs Scratch avant de les assembler.

La démarche proposée est particulièrement intéressante car elle apprend aux élèves que les instructions (les blocs...) peuvent être testées avant d'être assemblées pour former un programme.

Quand ils rencontreront des difficultés ou feront face à un bug, ils devront bien souvent démonter leur code et tester les blocs pour comprendre où est le problème.

### Des projets prêts à l'emploi

Pour gagner du temps, les élèves seront invités à charger des programmes dans lesquels des sprites et même un peu de code sont proposés et prêts à l'emploi.

Deux méthodes sont possibles pour s'approprier ces projets:

1. Si vous utilisez la version de Scratch en ligne et que les élèves ont un compte Scratch (<https://scratch.mit.edu>), indiquez aux élèves comment
  - accéder à ce studio : <https://scratch.mit.edu/studios/51335090>,

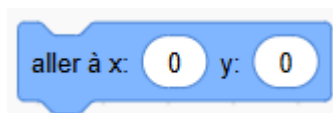
- ouvrir le projet renseigné,
  - le remixer et le renommer.
2. **Si vous utilisez la version Scratch hors ligne**, vous devrez fournir aux élèves les fichier Scratch (extension SB3) en les copiant dans un dossier de leur ordinateur et leur expliquer comment :
- y accéder,
  - charger le fichier renseigné avec Scratch et le renommer
  - et..... ne pas oublier de le sauvegarder de temps en temps....
  -

## Learning by doing .....Ne pas vouloir tout expliquer !

L'équipe de l'UCL propose de travailler de façon très progressive et de donner l'occasion aux élèves de s'exercer à chaque étape pour qu'ils puissent le mieux possible fixer, comprendre les concepts et acquérir les savoir-faire attendus.

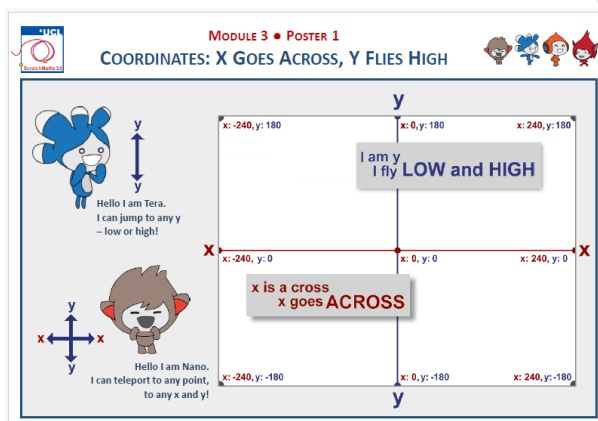
Mais ils nous rappellent par ailleurs que les Mathématiques et la programmation, cela s'apprend aussi par petites explorations successives, en acceptant qu'on ne peut pas tout comprendre et maîtriser d'entrée de jeu... C'est le **Learning by doing**.

Par exemple. Dès le début, l'élève va être (passivement...) confronté à ce bloc :



Si l'enseignant, d'entrée de jeu, décide d'expliquer à ses élèves ce qu'est un repère orthonormé, les axes X et Y, les coordonnées, abscisses et ordonnées.... il n'arrivera jamais au bout de la séquence. Il faut pouvoir dire à l'élève :

1. Tu ne dois pas vouloir tout comprendre du premier coup !
2. Retiens que ce bloc à pour rôle de placer le sprite au milieu de la scène



En l'occurrence, des applications qui font appel au repère X-Y interviendront avec le module 3... et encore, d'une façon très intuitive et progressive. Avant cela, la notion sera explorée, essayée, utilisée pour des opérations concrètes. Le but est de faire en sorte qu'au moment où le concept sera formalisé, l'élève soit prêt à le cueillir, comme un fruit mûr.

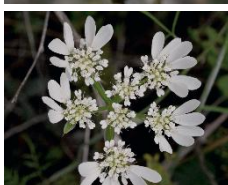
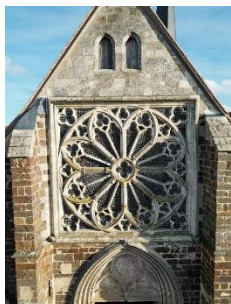
## La Récré Scratch

Dans les premières séquences, on utilise un nombre très limité d'instructions (de blocs). Les élèves seront tentés d'en essayer d'autres, de jouer avec d'autres sprites, de changer de décor. Il ne sera pas facile de les en empêcher. Mais il faudra bien... sinon, ils vont se disperser et perdre de vue les apprentissages.

C'est le même problème avec les cartes de programmation (Scratch+ et FMTTN). Généralement, la consigne que je donne est : « on suit à la lettre les instructions et le parcours proposé ». Et, pour que cela marche, je propose une carotte : « quand nous aurons fini la séquence, il y aura une Récré Scratch pendant laquelle ils pourront essayer autre chose, faire des rosaces avec d'autres sprites, etc...

## Des liens avec d'autres disciplines !

Une rosace n'est pas qu'une figure abstraite ! Des liens peuvent être faits avec d'autres disciplines : les arts, l'architecture, la biologie. Et pourquoi pas, en parallèle, une activité dans le cadre du PECA ?



# 1 Déplacer, tourner et estampiller

Fichiers SB3 utilisés :

- **SM10-Déplacer-Estampiller**
- **SM11-Tourner-Estampiller**
- **SM12-On code** (à créer sur base de SM11)

## Quelques conseils pour la mise en œuvre

### Ne pas brûler les étapes

De façon très progressive et sans chercher à brûler les étapes, les élèves sont invités à dessiner leurs premières rosaces en

- 1.2 Déplaçant le sprite avec la souris puis en cliquant sur **estampiller**.
- 1.3 Faisant tourner le sprite à l'aide des blocs **tourner de ...**
- 1.4 Déplaçant les blocs à l'aide d'un bloc **avancer de ...**

En fin de séquence 1.4, le premier script est réalisé !

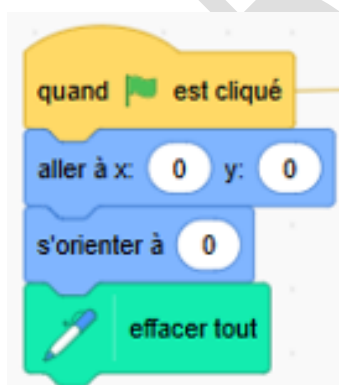
Un script, c'est un assemblage de quelques blocs que l'on peut tester en faisant un clic dessus.

### Maintenant , on programme code !

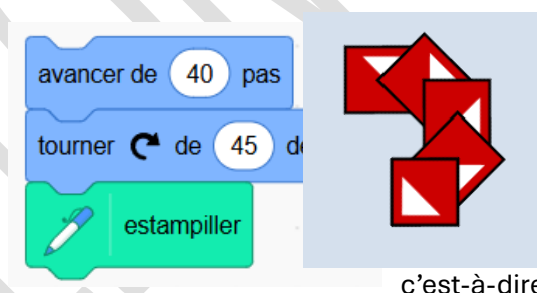
À ce stade, il faut éviter de parler de programme ou de programmation. Nous sommes plutôt au stade du codage, de la traduction d'actions simples (déplacer, tourner, imprimer) en langage informatique, c'est-à-dire en blocs dans le cas de Scratch.

On peut dire à ce stade que la rosace se compose en mode semi-automatique puisque le script exécute bien les trois actions **avancer-tourner-estampiller** mais l'élève doit cliquer plusieurs fois dessus pour que la rosace apparaisse.

### Le script d'initialisation



Ne pas oublier de rappeler aux élèves qu'ils peuvent exécuter le script d'initialisation chaque fois qu'ils souhaitent recommencer une rosace.



c'est-à-dire



## 2 Répéter, alterner

Fichier SB3 utilisé :

- **SM13-Repeter-alterner**

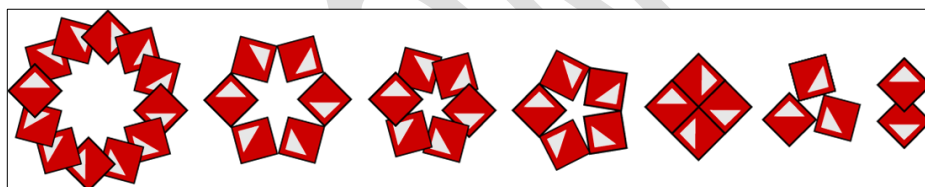
### Quelques conseils pour la mise en œuvre

Dans cette deuxième séquence, on fait le grand saut : l'utilisation d'une boucle.

Ici encore, veiller à ne pas brûler les étapes. L'élève va découvrir les avantages incontestables de l'utilisation des bloc **répéter...fois** après s'être livré à l'exercice fastidieux qui consiste à dupliquer 12 fois le même code.

Si vous en avez l'occasion, faites une petite pause quand les élèves arrivent à l'étape **2.2 Répéter**, et invitez-les à modifier la rosace en remplaçant tous les **avancer de 40 pas** par **avancer de 25 pas**...

À l'étape **2.3**, pour effectuer les rosaces suivantes, l'élève devrait se poser la question :  
« Quel angle, quelle rotation ? »

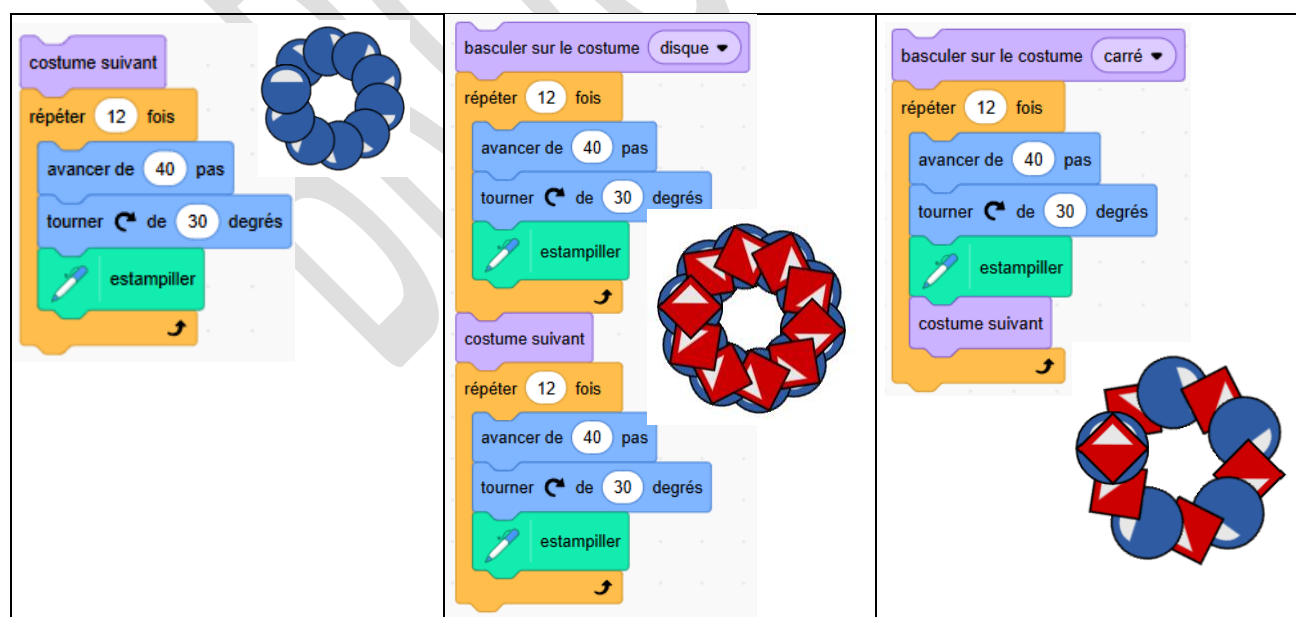


On propose de travailler cela en mode déconnecté dans le carnet de l'élève.

costume suivant

À l'étape **2.4.**, l'élève découvre l'existence des deux costumes et du bloc **costume suivant** qui permet de basculer de l'un à l'autre.

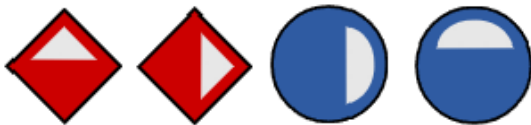
Veillez à insister auprès des élèves pour qu'ils réfléchissent au placement du bloc, avant la boucle, dans la boucle, après la boucle..... et bien sûr, ne pas leur interdire de dupliquer le script...



En fin d'étape **2.4**, « Il y a un problème ». Laisser l'élève chercher un peu... Il faut qu'il comprenne que sur la scène, il y a les pavés estampillés (imprimés) et le sprite. Les pavés estampillés ne peuvent plus être déplacés ou modifiés, on peut juste tout effacer (bloc **effacer tout**). Le pavé sprite peut lui être déplacé...

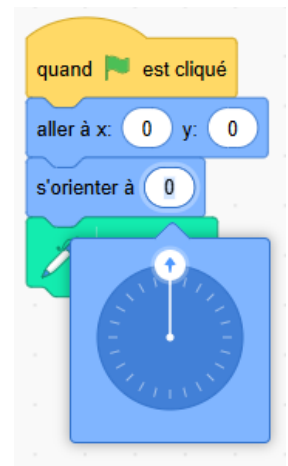
**Etape 2.5 par groupes de deux** . Pas de nouveaux blocs ni de nouvelles procédures. Mais l'élève va devoir tout remobiliser, modifier les scripts en ce compris le script d'initialisation.

Il vont notamment devoir modifier l'orientation du sprite au départ et donc le bloc s'orienter à .... Si nécessaire, on peut en profiter pour faire le point sur la notion d'orientation et sur l'utilisation du cadran d'orientation.



On peut aussi attirer l'attention des élèves sur le fait que la forme des deux costumes permet de détecter leur orientation :

Ici, dans l'ordre, les blocs sont orientés à 0°, 90°, 90°, 0°



---

## 3 Rosaces et motifs circulaires

Fichier SB3 utilisé :

- **SM14-Rosaces**

### Quelques conseils pour la mise en œuvre

#### Un nouvel algorithme

L'algorithme basé sur les trois instructions **tourner-avancer-estampiller** montre vite ses limites si on veut construire des rosaces plus complexes.

L'algorithme **avancer-estampiller-reculer-tourner** permet d'enrichir les rosaces tout en maintenant une symétrie centrale. Il permet de contrôler plus finement le positionnement des motifs. Les élèves le comprendront à l'usage, grâce aux défis que proposent cette séquence et grâce à l'activité déconnectée de l'étape 3.3. Il n'est pas nécessaire de le leur expliquer d'entrée de jeu.

#### De nouveaux costumes

**Etape 3.4** Les nouveaux costumes permettent de créer des motifs encore plus variés. Pour les élèves qui sont à la peine, nous avons prévu la conception d'une rosace simple : **Un exemple pas à pas.**

#### Un seul script pour une rosace en un seul clic

Il est temps de proposer aux élèves de prendre l'habitude d'assembler le script d'initialisation et celui qui réalise la rosace.

---

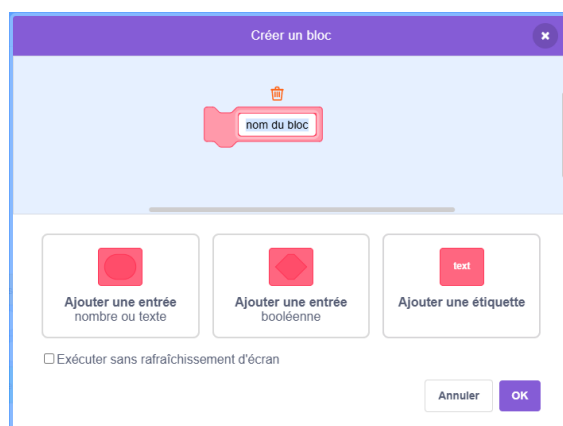
## 4 Travailler avec des blocs personnalisés

Fichier SB3 utilisé :

- **SM14-Rosaces** (à charger de nouveau)

Quelques conseils pour la mise en œuvre

### Les blocs personnalisés



L'équipe de l'UCL nous propose d'aborder les blocs personnalisés très tôt dans la progression. . D'aucuns pourraient penser que c'est trop tôt .... Je ne le pense pas..

Les élèves de fin de primaires comprennent assez facilement qu'une procédure peut et doit pouvoir être réutilisée.

La procédure implique de réaliser dans l'ordre plusieurs opérations mais n'a rien de compliqué. S'il le juge utile, l'enseignant peut commencer la séquence par une petite

démonstration.

Par contre, on évitera à ce stade d'utiliser les paramètres (les **entrées**) et la coche **Exécuter sans rafraîchissement d'écran**.

Les exemples et les explications ne manquent pas dans cette séquence et en plus, un exercice à faire dans le cahier de l'élève vient en renfort.

C'est par cette séquences et les blocs personnalisés que le premier module se termine.