Tout est algorithme, tout est fonction.

Première approche d'une pensée fonctionnelle
du programme de Première S

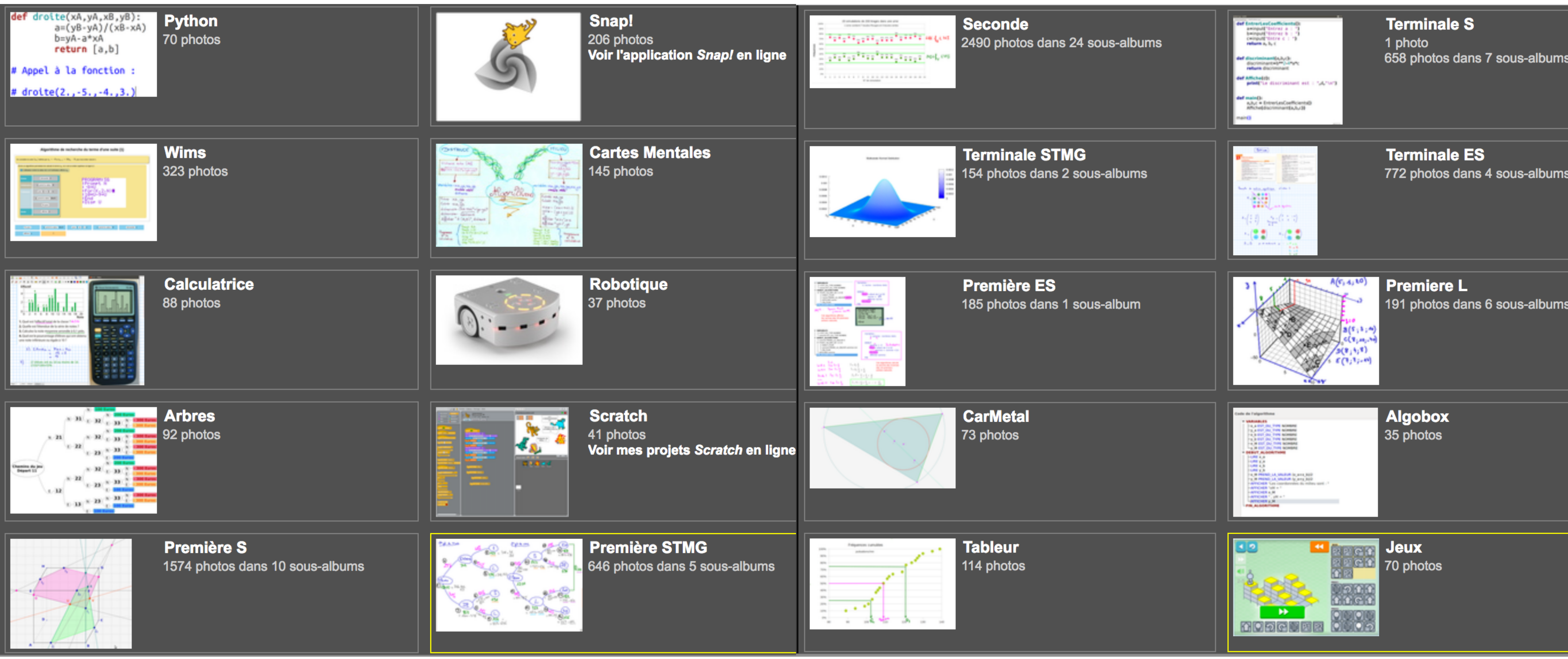
IREM de la Réunion, mercredi 13 février 2019

J'adore enseigner en classe de Première S et j'ai eu la chance d'en avoir une dès ma première année d'enseignement. Depuis, j'ai enseigné en Première S bien plus qu'un an sur deux. C'est à mon sens la seule classe où l'on peut réellement faire des mathématiques, éveiller la curiosité des élèves, montrer la beauté des mathématiques, faire dessiner les élèves et prendre le temps de les faire programmer. La raison en est évidente, pas de pression du baccalauréat au bout, nous n'avons que le bonheur de pratiquer des mathématiques, habituer les élèves à explorer, à raisonner.
Les algorithmes étaient déjà partout présents dans les programmes de lycée en 1991, lorsque j’ai débuté en tant qu’enseignante, après quelques années dans le privé. Il suffisait de lire entre les lignes... On pouvait programmer en Logo ou sur la calculatrice avec les élèves.
Venant du privé avec une formation d'ingénieur, j'ai toujours adoré programmer. D'autant que mon père m'avait offert ma première calculatrice à la boutique du MIT : une HP34C, lorsque nous y avions visité mon oncle, émigré Haïtien aux Etats-Unis, car il y avait une chaire en chimie.

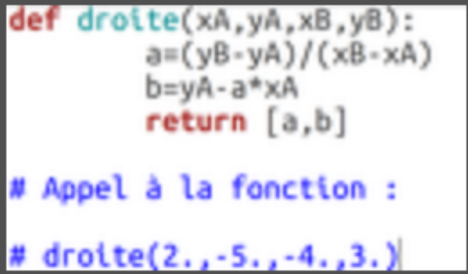
Pas étonnant que tout au long de ma carrière d'enseignante, j'ai essayé de transmettre ce goût pour la programmation à mes élèves…

Maths en classe en images

Vous trouverez de nombreux programmes sur ce site
Maths en classe en images nathalierun.net/lycee/piwigo/



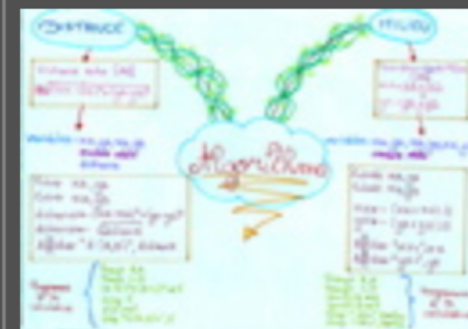
J'y mets en ligne ce qui est fait dans mes classes depuis 2008.



Python
70 photos

Snap!
206 photos
Voir l'application *Snap!* en ligne

Wims
323 photos



Cartes Mentales
145 photos

Calculatrice
88 photos

Robotique
37 photos

Arbres
92 photos

Scratch
41 photos
Voir mes projets *Scratch* en ligne

Première S
1574 photos dans 10 sous-albums

Première STMG
646 photos dans 5 sous-albums

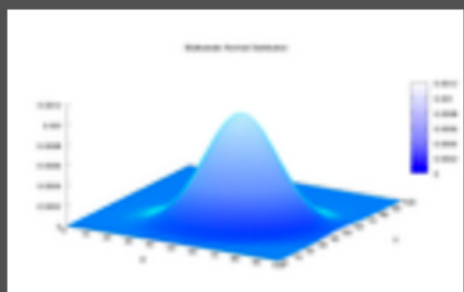
Seconde

2490 photos dans 24 sous-albums

Terminale S

1 photo

658 photos dans 7 sous-albums



Terminale STMG

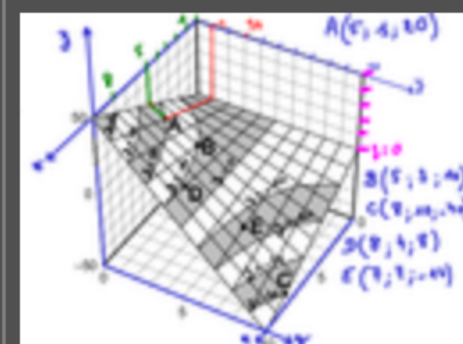
154 photos dans 2 sous-albums

Terminale ES

772 photos dans 4 sous-albums

Première ES

185 photos dans 1 sous-album



Premiere L

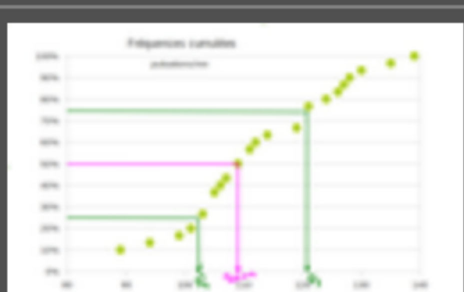
191 photos dans 6 sous-albums

CarMetal

73 photos

Algobox

35 photos



Tableur

114 photos

Jeux

70 photos

Maths en classe en images

En particulier, voici le lien obtenu en effectuant une recherche avec les mots-clé :
programme, algorithme, code, python, snap, scratch, algobox, calculatrice
nathalierun.net/lycee/piwigo/index.php?/search/683
Il faut penser à activer l'ordre de Tri Date d'ajout ancien -> récent.
Vous obtenez au moins 766 images, que vous pourrez alors faire défiler sous forme de diaporama.

Au moins 153 images illustrent des algorithmes (avec les mêmes mots-clé) en classe de première S ici :
nathalierun.net/lycee/piwigo/index.php?/search/684

C'est intéressant de voir d'une part la variété des algorithmes : on peut en voir dans toutes les branches des mathématiques abordées, mais aussi l'évolution des outils utilisés pour les mettre en oeuvre : la calculatrice, les logiciels utilisés en classe tels que :

AlgoBox

CarMetal

Python

Scratch

Snap!

Aseba pour Thymio

des jeux → LightBot, "The binary game", 2048

Snap!

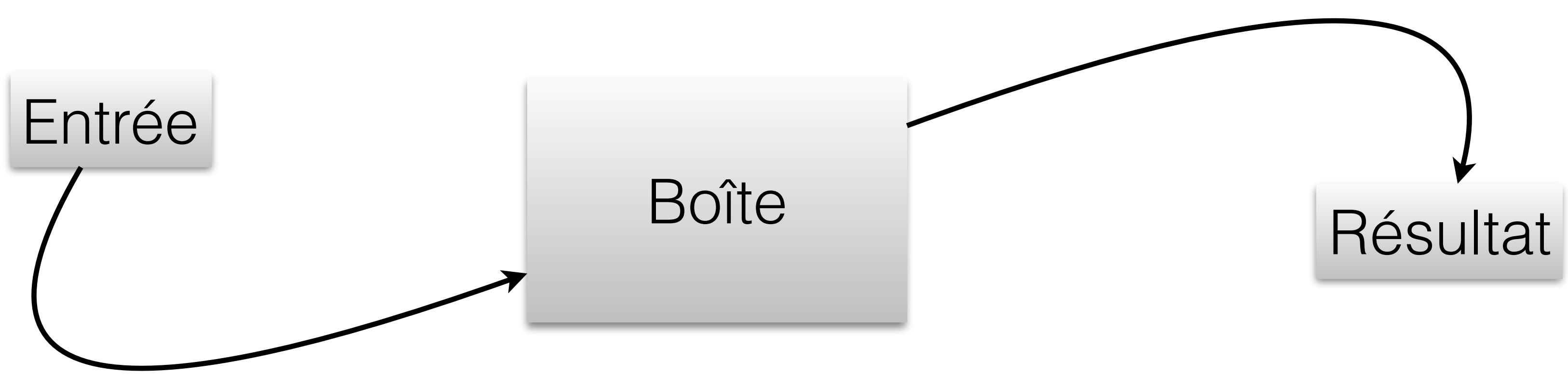
Le Scratch qui code du code…

Le cloud de Snap! héberge actuellement
- 2.5 millions de projets,
- dont environ 800,000 projets partagés,
- et quelques projets d'exemples .

Ce que je trouve particulièrement intéressant en naviguant sur mon cahier de textes en images, c'est de voir à quel point la découverte du logiciel *Snap!* a fait évoluer mes pratiques en classe et ma vision algorithmique des notions abordées. La première utilisation de Snap! en classe a eu lieu le 30 mars 2016 et depuis, on voit clairement mes pratiques évoluer.

Si vous ne connaissez pas encore *Snap!*, je vous invite à lire la page *"About"* de son site père : snap.berkeley.edu/about.html ,
et ce premier article que je lui ai consacré sur le site de l'IREM en avril 2017 :
Coder des algorithmes avec *Snap!* - Programmation visuelle au lycée

A partir de la première illustration de l'utilisation de *Snap!* comme bloc qui renvoie une valeur
nathalierun.net/lycee/piwigo/picture.php?/6380/category/104
tout est toujours codé grâce à *Snap!* comme une fonction.



Grâce à mes années d'enseignement en classe de première S, je pressens que tout est algorithme en Mathématiques, tout est fonction. Et *Snap!* m'a totalement convaincue de cela, tout en m'aidant à l'illustrer, à le confirmer.
images.math.cnrs.fr/Et-si-on-commencait-par-les.html

Dans cet article (difficile !), on montre qu'au lieu de fonder les mathématiques sur la notion d'ensemble comme le veut la tradition, **on peut mettre la notion de fonction au centre de l'édifice**. Et ce raisonnement trouve son origine dans les travaux d'Alonzo Church, connu principalement pour le développement du lambda-calcul qui a débouché sur la programmation fonctionnelle qui est un paradigme de programmation de type déclaratif qui considère le calcul en tant qu'évaluation de fonctions mathématiques.

Never had any mathematical conversations with anybody, because there was nobody else in my field.

Alonzo Church

Mathématicien, Logicien
(14 June, 1903 – 11 August, 1995)

D'ailleurs, on notera l'anecdote suivante : la mascotte de *Snap!* s'appelle Alonzo et ses cheveux sont gominés en forme de Lambda...

λ

Petit historique sur le Lambda-Calcul

D’après David Perrin

Le lambda-calcul
le reTour de Babel ?

… Nombreux algorithmes dans les papyrus égyptiens, les tablettes babyloniennes, les mathématiques indiennes et chinoises.

III^{e} s Éléments d’Euclide : nombreuses constructions géométriques. Algorithme d’Euclide.

IX^{e} s Al-Khwârizmî a classifié les algorithmes existants pour les problèmes du premier et du second degré. Il a aussi publié un traité d’algèbre. Son nom a été latinisé en Algorismus, qui a donné le mot « algorithme ».

XVII^{e} s Le philosophe Wilhelm Leibniz cherche « l'alphabet des pensées humaines ».

1900 David Hilbert propose d'axiomatiser et de formaliser les mathématiques.

1901 Edmund Husserl veut fonder les vérités logiques sur des processus psychiques avec la phénoménologie.

1932 Alonzo Church invente le lambda-calcul qui est un système formel (modélisation mathématique d'un langage). **En lambda-calcul, tout est fonction.**

λ

fr.wikipedia.org/wiki/Lambda-calcul

1944 John von Neumann conçoit le premier ordinateur.

1948 Haskell Curry voit dans le lambda-calcul la structure logique sous-jacente aux langues naturelles.

…

2015 *Snap!* Build Your Own Blocks

snap.berkeley.edu/about.html

La pensée algorithmique

Qu’est-ce que la pensée algorithmique ?

Jeanette Wing en donne cette définition en 2011 :
« computational thinking is the thought processes involved in formulating a problem and expressing its solution(s) in such a way that a computer—human or machine—can effectively carry out. » revue.sesamath.net/spip.php?article1096

professeure d’informatique née en 1956,
diplômée du MIT.

C’est le mode de pensée qui permet de concevoir ce qu’un calculateur - humain ou machine - peut exécuter.

Avoir une pensée algorithmique permanente peut induire une pensée mathématique plus claire.

En effet, observer une pensée algorithmique demande d'analyser un problème afin de le décomposer en nombreuses petites fonctions ayant des tâches très précises, très réduites.

La pensée algorithmique va développer chez les élèves l’habitude de raisonner en termes de fonctions.

Je m'attache donc à aborder toute notion en première S de manière algorithmique.

Et *Snap!* s'y prête bien. Ainsi, je pratique en classe en temps réel une pensée algorithmique que nous illustrons sans cesse à l'aide de scripts *Snap!* en français, proche d'un langage de script illustrant la langue naturelle.

recherche d'un algorithme en classe

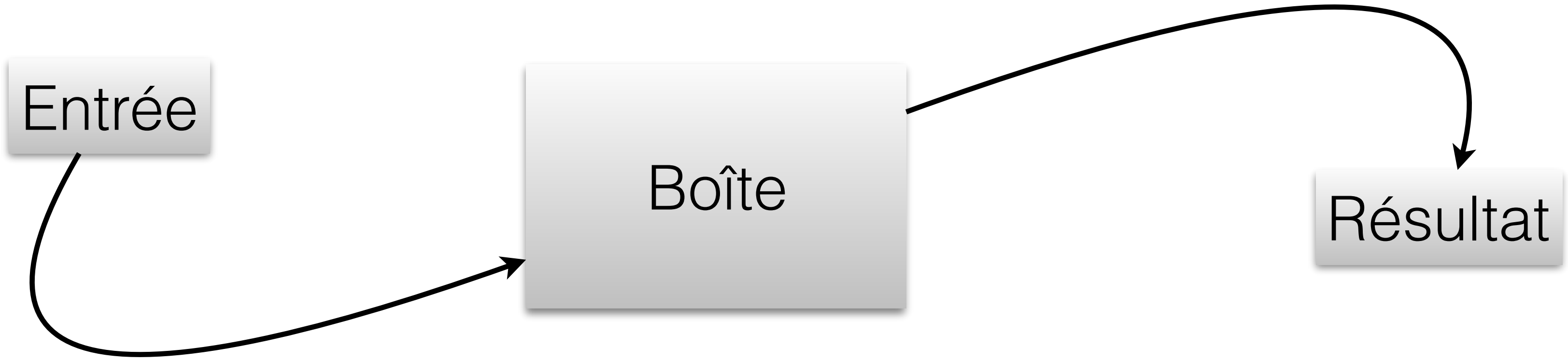
J'utilise *Snap!* en classe pour **écrire** un algorithme mais aussi pour le **coder** et le **tester**.

L'idée est d'ouvrir l'esprit des élèves à une pensée algorithmique indépendante d'un langage précis de programmation.

Bien sûr, j'utilise Python également comme langage de programmation textuel…

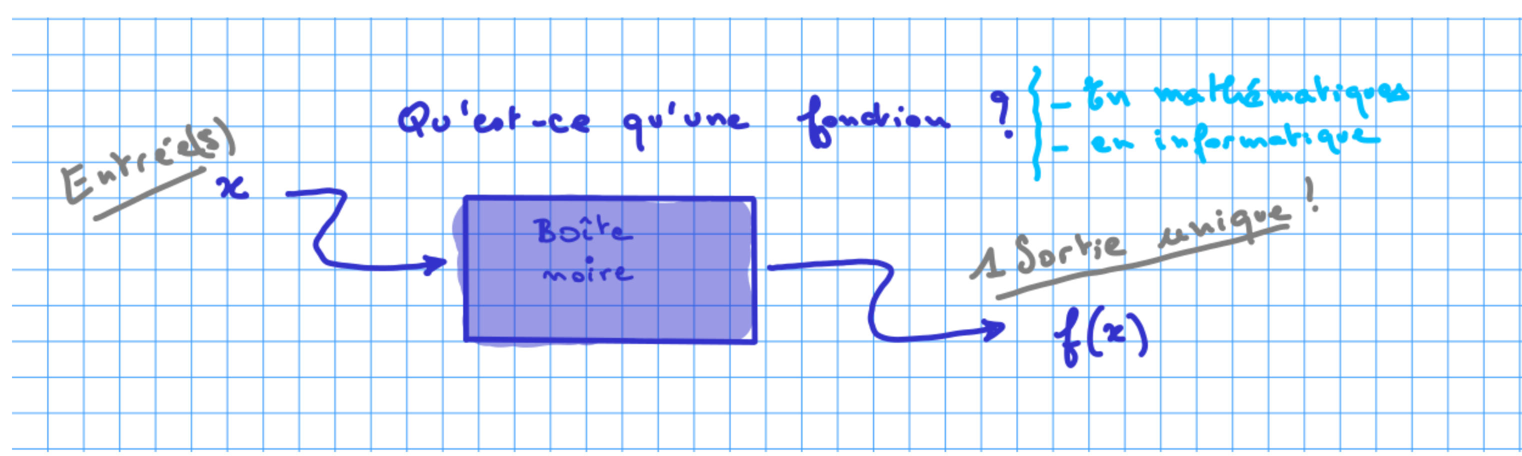
Toute pensée est un calcul…

J’essaie d'amener mes élèves à une pensée algorithmique, de les amener à penser fonctions en permanence.



On entre des objets, quelqu'ils soient. La boîte traite ces objets et renvoie un résultat, celui qui est cherché, attendu par l'élève. Cela oblige mes élèves à penser les Mathématiques en terme d'objets. De comprendre que les fonctions ne sont pas que numériques, les algorithmes non plus. Et qu'un résultat n'est pas forcément numérique. A partir de cet enseignement, leurs remarques me font grandir, je réalise à quel point je suis dans la bonne voie car ils m'apprennent des choses, me font réaliser la puissance de cette vision fonctionnelle de l'algorithmique.

Lorsque j'ai demandé aux élèves, lors d'une activité sur les fractales, quel était l'objet renvoyé par l'algorithme qui dessine un flocon de Von Koch, Aliyah m'a répondu comme une évidence *"Et bien, le dessin Madame !"*.
On en a déduit ensemble que cet algorithme était bien une fonction : il en a tous les ingrédients : des entrées traitées dans une boîte, et le dessin qui en résulte.



nathalierun.net/lycee/piwigo/picture.php?/7419/search/683

Document d'accompagnement du programme de seconde
Algorithmique et programmation

Quelques extraits…

Algorithmique_et_programmation_787733.pdf

Deux idées essentielles :
la **notion universelle de fonction** d'une part et la **programmation comme production d'un texte** dans un langage informatique d'autre part.

Les notions mathématique et informatique de fonction relèvent du même concept universel.

Notions d'entrées-sorties

On notera que les notions d'entrées-sorties (fonctions `input` et `print`) ne sont pas développées dans ce document : elles ne relèvent pas de la pensée algorithmique et l'accent mis par le programme sur la notion de fonction permet de s'en libérer complètement.

Évolution de l'écriture des algorithmes au baccalauréat

A partir de juin 2018

Dans un objectif de simplicité et de cohérence, il a été proposé une évolution de l'écriture des algorithmes dans les sujets de baccalauréat obéissant aux principes suivants :

- → suppression de la déclaration des variables, les hypothèses faites sur les variables étant précisées par ailleurs
- → suppression des entrées-sorties (en cohérence avec le nouveau document d'accompagnement en 2nde – p5)
- → simplification de la syntaxe, avec le symbole ← pour l'affectation.

$S \leftarrow 0$

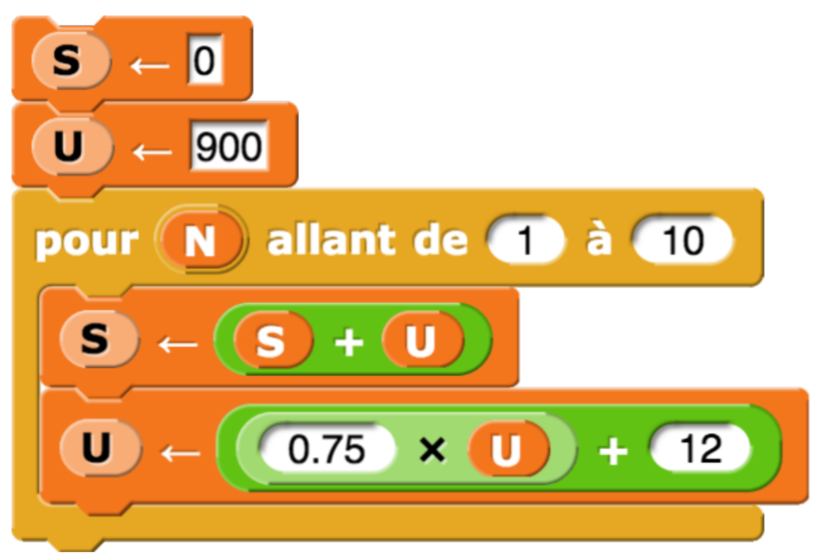
$U \leftarrow 900$

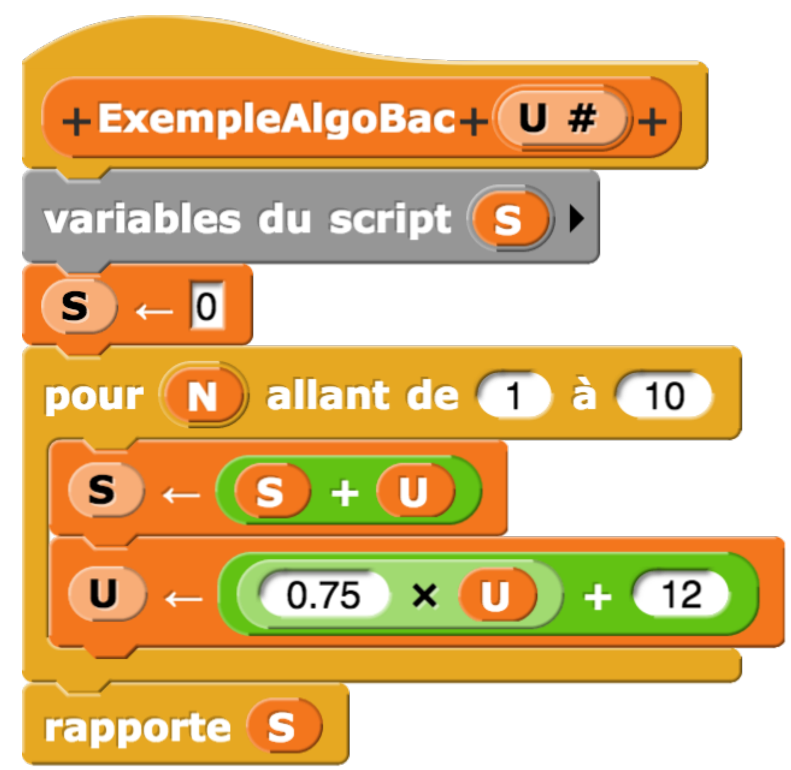
Pour N allant de 1 à 10

    $S \leftarrow \cdots$

    $U \leftarrow 0{,}75\,U + 12$

Fin Pour





Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

CONTENUS	CAPACITÉS ATTENDUES	COMMENTAIRES
Second degré Forme canonique d'une fonction polynôme de degré deux. Équation du second degré, discriminant. Signe du trinôme.	• Déterminer et utiliser la forme la plus adéquate d'une fonction polynôme de degré deux en vue de la résolution d'un problème : développée, factorisée, canonique.	On fait le lien avec les représentations graphiques étudiées en classe de seconde. ◇ Des activités algorithmiques doivent être réalisées dans ce cadre.
Étude de fonctions Fonctions de référence $x \mapsto \sqrt{x}$ et $x \mapsto \lvert x \rvert$.	• Connaître les variations de ces deux fonctions et leur représentation graphique. ⊡ Démontrer que la fonction racine carrée est croissante sur $[0;+\infty[$. ⊡ Justifier les positions relatives des courbes représentatives des fonctions $x \mapsto x$, $x \mapsto x^2$ et $x \mapsto \sqrt{x}$.	Aucune technicité dans l'utilisation de la valeur absolue n'est attendue.
Sens de variation des fonctions $u+k$, λu, $\sqrt{u}$ et $\frac{1}{u}$, la fonction *u* étant connue, *k* étant une fonction constante et λ un réel.	• Exploiter ces propriétés pour déterminer le sens de variation de fonctions simples.	⊡ On nourrit la diversité des raisonnements travaillés dans les classes précédentes en montrant à l'aide de contre-exemples qu'on ne peut pas énoncer de règle générale donnant le sens de variation de la somme ou du produit de deux fonctions. L'étude générale de la composée de deux fonctions est hors programme.

Relever des algorithmes relatif à cette page.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

Dérivation Nombre dérivé d'une fonction en un point.		Le nombre dérivé est défini comme limite du taux d'accroissement $\dfrac{f(a+h)-f(a)}{h}$ quand h tend vers 0. On ne donne pas de définition formelle de la limite.
Tangente à la courbe représentative d'une fonction dérivable en un point. Fonction dérivée.	• Tracer une tangente connaissant le nombre dérivé.	L'utilisation des outils logiciels facilite l'introduction du nombre dérivé.
Dérivée des fonctions usuelles : $x \mapsto \sqrt{x}$, $x \mapsto \dfrac{1}{x}$ et $x \mapsto x^n$ (n entier naturel non nul).	• Calculer la dérivée de fonctions.	On évite tout excès de technicité dans les calculs de dérivation. Si nécessaire, dans le cadre de la résolution de problèmes, le calcul de la dérivée d'une fonction est facilité par l'utilisation d'un logiciel de calcul formel.
Dérivée d'une somme, d'un produit et d'un quotient.		Il est intéressant de présenter le principe de démonstration de la dérivation d'un produit.
Lien entre signe de la dérivée et sens de variation. Extremum d'une fonction.	• Exploiter le sens de variation pour l'obtention d'inégalités.	Il n'est pas toujours utile de recourir à la dérivation pour étudier le sens de variation d'une fonction. On traite quelques problèmes d'optimisation.

Suites Modes de génération d'une suite numérique.	• Modéliser et étudier une situation à l'aide de suites. ◇ Mettre en œuvre des algorithmes permettant : - d'obtenir une liste de termes d'une suite ; - de calculer un terme de rang donné.	Il est important de varier les approches et les outils. L'utilisation du tableur et la mise en œuvre d'algorithmes sont l'occasion d'étudier en particulier des suites générées par une relation de récurrence.
Suites arithmétiques et suites géométriques.	▣ Établir et connaître les formules donnant $1+2+\ldots+n$ et $1+q+\ldots+q^n$.	
Sens de variation d'une suite numérique. Approche de la notion de limite d'une suite à partir d'exemples.	• Exploiter une représentation graphique des termes d'une suite.	◇ On peut utiliser un algorithme ou un tableur pour traiter des problèmes de comparaison d'évolutions et de seuils. Par exemple, dans le cas d'une suite croissante non majorée, on peut déterminer un rang à partir duquel tout terme de la suite est supérieur à un nombre donné. Le tableur, les logiciels de géométrie dynamique et de calcul sont des outils adaptés à l'étude des suites, en particulier pour l'approche expérimentale de la notion de limite. On ne donne pas de définition formelle de la limite.

Relever des algorithmes relatif à cette page.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

CONTENUS	CAPACITÉS ATTENDUES	COMMENTAIRES
Géométrie plane Condition de colinéarité de deux vecteurs : $xy' - yx' = 0$.		
Vecteur directeur d'une droite. Équation cartésienne d'une droite.	▣ Utiliser la condition de colinéarité pour obtenir une équation cartésienne de droite. • Déterminer une équation cartésienne de droite connaissant un vecteur directeur et un point. • Déterminer un vecteur directeur d'une droite définie par une équation cartésienne.	On fait le lien entre coefficient directeur et vecteur directeur. L'objectif est de rendre les élèves capables de déterminer efficacement une équation cartésienne de droite par la méthode de leur choix.
Expression d'un vecteur du plan en fonction de deux vecteurs non colinéaires.	• Choisir une décomposition pertinente dans le cadre de la résolution de problèmes.	On ne se limite pas au cadre de la géométrie repérée.

Trigonométrie Cercle trigonométrique. Radian. Mesure d'un angle orienté, mesure principale.	• Utiliser le cercle trigonométrique, notamment pour : - déterminer les cosinus et sinus d'angles associés ; - résoudre dans **R** les équations d'inconnue ***x*** : $\cos x = \cos a$ et $\sin x = \sin a$.	L'étude des fonctions cosinus et sinus n'est pas un attendu du programme.
Produit scalaire dans le plan Définition, propriétés.	• Calculer le produit scalaire de deux vecteurs par différentes méthodes : - projection orthogonale ; - analytiquement ; - à l'aide des normes et d'un angle ; - à l'aide des normes. • Choisir la méthode la plus adaptée en vue de la résolution d'un problème.	⊡ Il est intéressant de démontrer l'égalité des expressions attachées à chacune de ces méthodes. ⊡ La démonstration du théorème de la médiane fournit l'occasion de travailler le calcul vectoriel en lien avec le produit scalaire.
Vecteur normal à une droite.	• Déterminer une équation cartésienne de droite connaissant un point et un vecteur normal. • Déterminer un vecteur normal à une droite définie par une équation cartésienne.	
Applications du produit scalaire : calculs d'angles et de longueurs ; formules d'addition et de duplication des cosinus et sinus.	⊡ Déterminer une équation de cercle défini par son centre et son rayon ou par son diamètre. ⊡ Démontrer que : $\cos(a-b) = \cos a \cos b + \sin a \sin b$	La relation de Chasles pour les angles orientés est admise.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

CONTENUS	CAPACITÉS ATTENDUES	COMMENTAIRES
Statistique descriptive, analyse de données Caractéristiques de dispersion : variance, écart-type. Diagramme en boîte.	• Utiliser de façon appropriée les deux couples usuels qui permettent de résumer une série statistique : (moyenne, écart-type) et (médiane, écart interquartile). • Étudier une série statistique ou mener une comparaison pertinente de deux séries statistiques à l'aide d'un logiciel ou d'une calculatrice.	On utilise la calculatrice ou un logiciel pour déterminer la variance et l'écart-type d'une série statistique. Des travaux réalisés à l'aide d'un logiciel permettent de faire observer des exemples d'effets de structure lors du calcul de moyennes.
Probabilités Variable aléatoire discrète et loi de probabilité. Espérance, variance et écart-type.	• Déterminer et exploiter la loi d'une variable aléatoire. • Interpréter l'espérance comme valeur moyenne dans le cas d'un grand nombre de répétitions.	À l'aide de simulations et d'une approche heuristique de la loi des grands nombres, on fait le lien avec la moyenne et la variance d'une série de données. On exploite les fonctionnalités de la calculatrice ou d'un logiciel pour déterminer l'espérance, la variance et l'écart-type d'une variable aléatoire. ▣ On démontre les formules suivantes sur l'espérance et la variance : $E(aX+b)=aE(X)+b$ et $V(aX)=a^2V(X)$.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

Modèle de la répétition d'expériences identiques et indépendantes à deux ou trois issues.	• Représenter la répétition d'expériences identiques et indépendantes par un arbre pondéré. • Utiliser cette représentation pour déterminer la loi d'une variable aléatoire associée à une telle situation.	Pour la répétition d'expériences identiques et indépendantes, la probabilité d'une liste de résultats est le produit des probabilités de chaque résultat. La notion de probabilité conditionnelle est hors programme. On peut aussi traiter quelques situations autour de la loi géométrique tronquée. ◇ On peut simuler la loi géométrique tronquée avec un algorithme.
Épreuve de Bernoulli, loi de Bernoulli. Schéma de Bernoulli, loi binomiale (loi du nombre de succès). Coefficients binomiaux, triangle de Pascal.	• Reconnaître des situations relevant de la loi binomiale. • Calculer une probabilité dans le cadre de la loi binomiale.	La représentation à l'aide d'un arbre est privilégiée : il s'agit ici d'installer une représentation mentale efficace. On peut ainsi : faciliter la découverte de la loi binomiale pour des petites valeurs de *n* ($n \leq 4$) ; introduire le coefficient binomial $\binom{n}{k}$ comme nombre de chemins de l'arbre réalisant *k* succès pour *n* répétitions ; établir enfin la formule générale de la loi binomiale.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

	▣ Démontrer que $\binom{n}{k}+\binom{n}{k+1}=\binom{n+1}{k+1}$.	Cette égalité est établie en raisonnant sur le nombre de chemins réalisant $k+1$ succès pour $n+1$ répétitions. On établit également la propriété de symétrie des coefficients binomiaux.
	• Représenter graphiquement la loi binomiale.	L'utilisation des coefficients binomiaux dans des problèmes de dénombrement et leur écriture à l'aide des factorielles ne sont pas des attendus du programme. En pratique, on utilise une calculatrice ou un logiciel pour obtenir les valeurs des coefficients binomiaux, calculer directement des probabilités et représenter graphiquement la loi binomiale.
Espérance, variance et écart-type de la loi binomiale.	• Utiliser l'espérance d'une loi binomiale dans des contextes variés.	La formule donnant l'espérance de la loi binomiale est conjecturée puis admise, celle de la variance est admise. ◇ On peut simuler la loi binomiale avec un algorithme.

Programme officiel de mathématiques de la classe de première S, paru au BO le 30 septembre 2010

Relever des algorithmes relatif à cette page.

Échantillonnage Utilisation de la loi binomiale pour une prise de décision à partir d'une fréquence.	• Exploiter l'intervalle de fluctuation à un seuil donné, déterminé à l'aide de la loi binomiale, pour rejeter ou non une hypothèse sur une proportion.	L'objectif est d'amener les élèves à expérimenter la notion de « différence significative » par rapport à une valeur attendue et à remarquer que, pour une taille de l'échantillon importante, on conforte les résultats vus en classe de seconde. ◇ L'intervalle de fluctuation peut être déterminé à l'aide d'un tableur ou d'un algorithme. Le vocabulaire des tests (test d'hypothèse, hypothèse nulle, risque de première espèce) est hors programme.

Construisons le programme de Première S…

…sous une vision algorithmique

…par une approche fonctionnelle

Je cherche à libérer les élèves des entrées/sorties et à les faire raisonner indépendamment d’un langage de programmation.

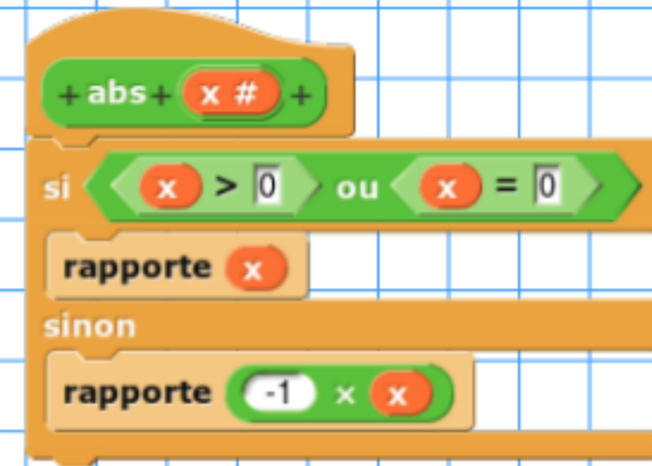
Se consacrer à l’essentiel : comprendre les mathématiques grâce aux algorithmes.

Les fonctions de référence

Fonction valeur absolue :

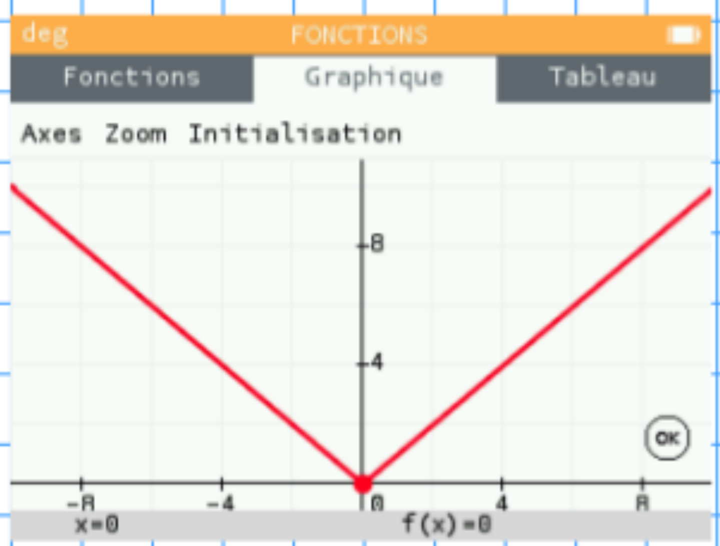
```
deg                PYTHON
from math import *

def abs(x):
  if (x>0 or x==0):
    return x
  else:
    return (-1)*x
```

$$|x| = \begin{cases} x & \text{si } x \geq 0 \\ -x & \text{sinon} \end{cases}$$



Script Python

```
def ValeurAbsolue(x):
    if (x>0 or x==0):
        return x
    else:
        return -1*x


y1=ValeurAbsolue(5)
y2=ValeurAbsolue(-5)
```



Vendredi 24

- Apporter le Pop
- p57 17-18

nathalierun.net/lycee/piwigo/picture.php?/7369/search/692

Enchaînement de fonctions

Comprendre comment se forme une fonction numérique est essentiel.

(3 / 5) × ((4 / 5) × x + 5)² + (5 / 3)

$$f(x)=\frac{3}{5}\left(\frac{4}{5}x+5\right)^2+\frac{5}{3}.$$

Cocher l'enchainement de fonctions de référence correspondant à f.

1. $x \xmapsto{\frac35x+\frac53} \cdots \xmapsto{x^2} \cdots \xmapsto{\frac45x+5} \cdots$
2. $x \xmapsto{\frac35x+\frac53} \cdots \xmapsto{\frac45x+5} \cdots \xmapsto{x^2} \cdots$
3. $x \xmapsto{x^2} \cdots \xmapsto{\frac35x+\frac53} \cdots \xmapsto{\frac45x+5} \cdots$
4. $x \xmapsto{\frac45x+5} \cdots \xmapsto{\frac35x+\frac53} \cdots \xmapsto{x^2} \cdots$
5. $x \xmapsto{\frac45x+5} \cdots \xmapsto{x^2} \cdots \xmapsto{\frac35x+\frac53} \cdots$
6. $x \xmapsto{x^2} \cdots \xmapsto{\frac45x+5} \cdots \xmapsto{\frac35x+\frac53} \cdots$

5

$$x \xmapsto{\frac45\square+5} \boxed{\frac45 x+5} \xmapsto{\square^2} \underbrace{\left(\frac45x+5\right)^2} \xmapsto{\frac35\square+\frac53} \underbrace{\frac35\left(\frac45x+5\right)^2+\frac53}_{=f(x)}$$

Exercice Wims illustré avec *Snap!*

(4 / 3) × (1 / ((3 / 4) × x + (5 / 3))) + 3

$$f(x)=\frac{4}{3\left(\frac34x+\frac53\right)}+3.$$

Cocher l'enchainement de fonctions de référence correspondant à f.

1. $x \xmapsto{\frac43x+3} \cdots \xmapsto{\frac1x} \cdots \xmapsto{\frac34x+\frac53} \cdots$
2. $x \xmapsto{\frac34x+\frac53} \cdots \xmapsto{\frac1x} \cdots \xmapsto{\frac43x+3} \cdots$
3. $x \xmapsto{\frac34x+\frac53} \cdots \xmapsto{\frac43x+3} \cdots \xmapsto{\frac1x} \cdots$
4. $x \xmapsto{\frac1x} \cdots \xmapsto{\frac34x+\frac53} \cdots \xmapsto{\frac43x+3} \cdots$
5. $x \xmapsto{\frac1x} \cdots \xmapsto{\frac43x+3} \cdots \xmapsto{\frac34x+\frac53} \cdots$
6. $x \xmapsto{\frac43x+3} \cdots \xmapsto{\frac34x+\frac53} \cdots \xmapsto{\frac1x} \cdots$

2

$$x \xmapsto{\frac34\square+\frac53} \frac34x+\frac53 \xmapsto{\frac{1}{\square}} \frac{1}{\frac34x+\frac53}$$

$$\xmapsto{\frac43\square+3} \frac43 \times \frac{1}{\frac34x+\frac53}+3 = \frac{4\times1}{3\left(\frac34x+\frac53\right)}+3$$

(3 / 5) × ((1 / 2) × x + 4)² + 1

$$f(x)=\frac{3}{5}\left(\frac{1}{2}x+\frac{1}{4}\right)^2+1.$$

Cocher l'enchainement de fonctions de référence correspondant à f.

1. $x \xmapsto{x^2} \cdots \xmapsto{\frac12x+\frac14} \cdots \xmapsto{\frac35x+1} \cdots$
2. $x \xmapsto{\frac35x+1} \cdots \xmapsto{x^2} \cdots \xmapsto{\frac12x+\frac14} \cdots$
3. $x \xmapsto{\frac12x+\frac14} \cdots \xmapsto{\frac35x+1} \cdots \xmapsto{x^2} \cdots$
4. $x \xmapsto{\frac35x+1} \cdots \xmapsto{\frac12x+\frac14} \cdots \xmapsto{x^2} \cdots$
5. $x \xmapsto{\frac12x+\frac14} \cdots \xmapsto{x^2} \cdots \xmapsto{\frac35x+1} \cdots$
6. $x \xmapsto{x^2} \cdots \xmapsto{\frac35x+1} \cdots \xmapsto{\frac12x+\frac14} \cdots$

$$x \xmapsto{\frac12\square+\frac14} \frac12x+\frac14 \xmapsto{\square^2} \left(\frac12x+\frac14\right)^2 \xmapsto{\frac35\square+1} \frac35\left(\frac12x+\frac14\right)^2+1$$

(5 / 2) × (1 / (5 × x² + (4 / 3))) + 3

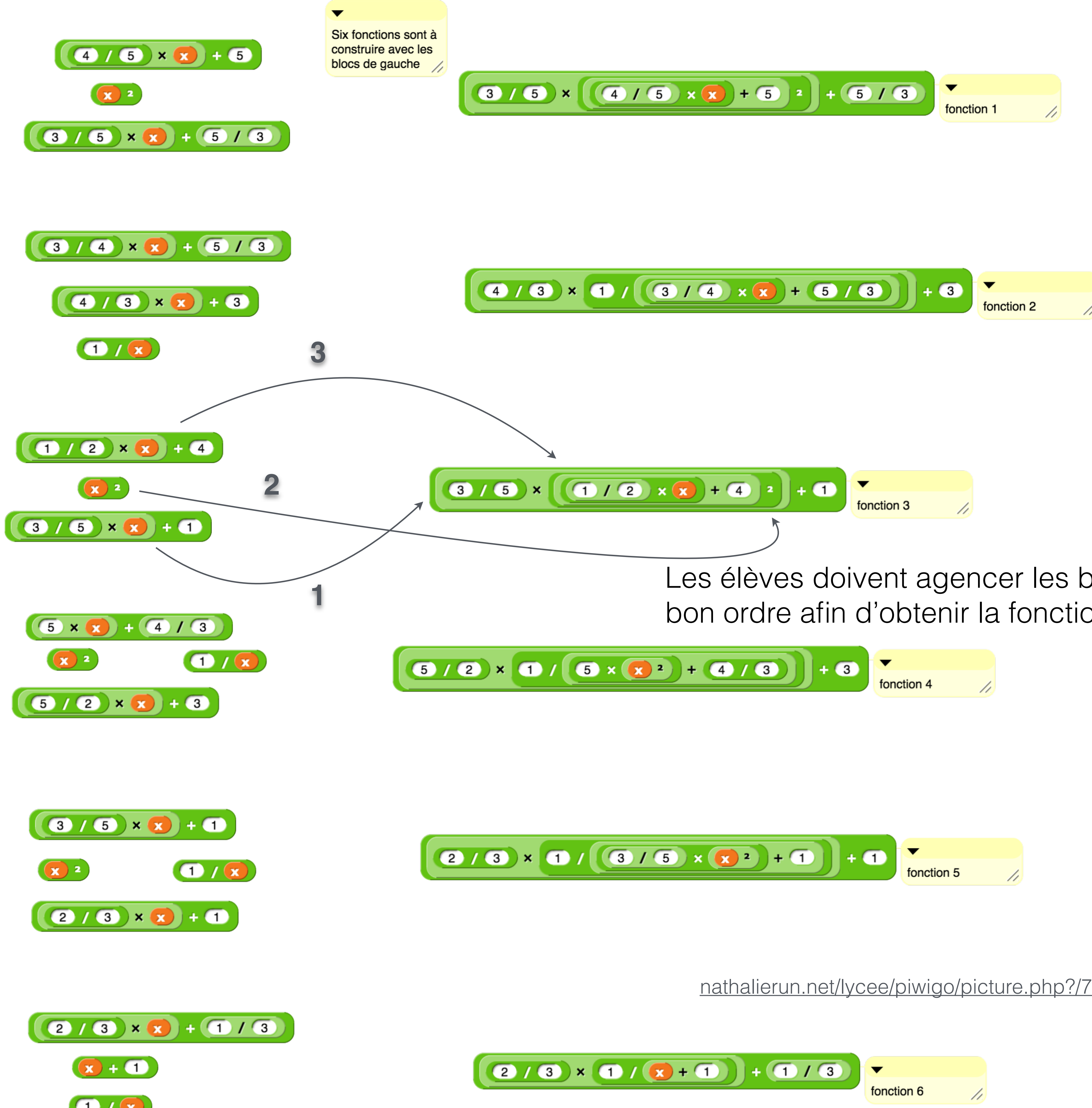
$$f(x)=\frac{5}{2\left(5x^2+\frac43\right)}+3.$$

Cocher l'enchainement de fonctions de référence correspondant à f.

1. $x \xmapsto{x^2} \cdots \xmapsto{\frac1x} \cdots \xmapsto{5x+\frac43} \cdots \xmapsto{\frac52x+3} \cdots$
2. $x \xmapsto{5x+\frac43} \cdots \xmapsto{x^2} \cdots \xmapsto{\frac52x+3} \cdots \xmapsto{\frac1x} \cdots$
3. $x \xmapsto{x^2} \cdots \xmapsto{5x+\frac43} \cdots \xmapsto{\frac1x} \cdots \xmapsto{\frac52x+3} \cdots$
4. $x \xmapsto{x^2} \cdots \xmapsto{5x+\frac43} \cdots \xmapsto{\frac52x+3} \cdots \xmapsto{\frac1x} \cdots$
5. $x \xmapsto{\frac1x} \cdots \xmapsto{5x+\frac43} \cdots \xmapsto{x^2} \cdots \xmapsto{\frac52x+3} \cdots$
6. $x \xmapsto{x^2} \cdots \xmapsto{\frac52x+3} \cdots \xmapsto{\frac1x} \cdots \xmapsto{5x+\frac43} \cdots$

Mardi 18 –
Finir les 3 fonctions qui restent.
(Détailler l'enchaînement)

nathalierun.net/lycee/piwigo/picture.php?/7412/category/125



Les élèves doivent agencer les blocs dans le bon ordre afin d'obtenir la fonction demandée.

nathalierun.net/lycee/piwigo/picture.php?/7416/category/125

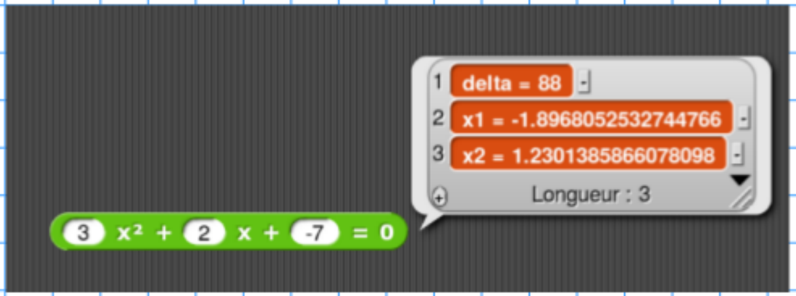
Réaliser l’appel d’une fonction avec *Snap!*

Le second degré

Le second degré

Logiciel Snap! de calcul sur le second degré. $f(x) = ax^2 + bx + c$

1) Une fonction qui entre l'équation et qui retourne Δ et les solutions si elles existent



Pour Vendredi 12/10

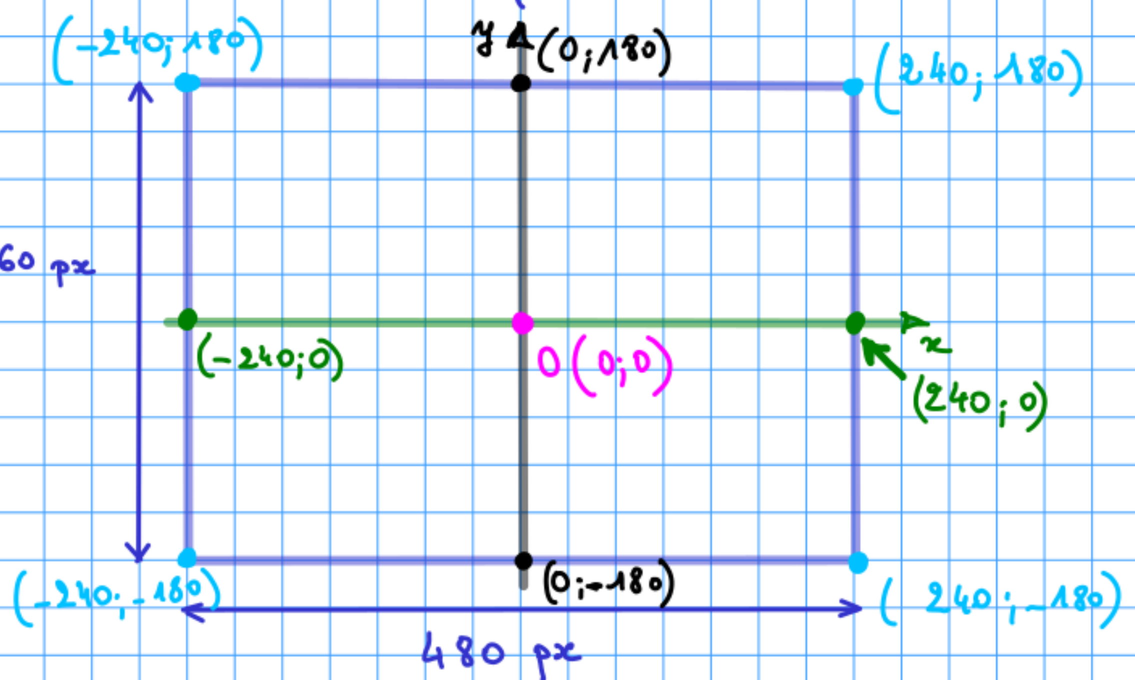
2) Une fonction qui renvoie les coordonnées du sommet de la parabole (en entrée $ax^2 + bx + c$)

3) Une fonction qui renvoie la factorisation

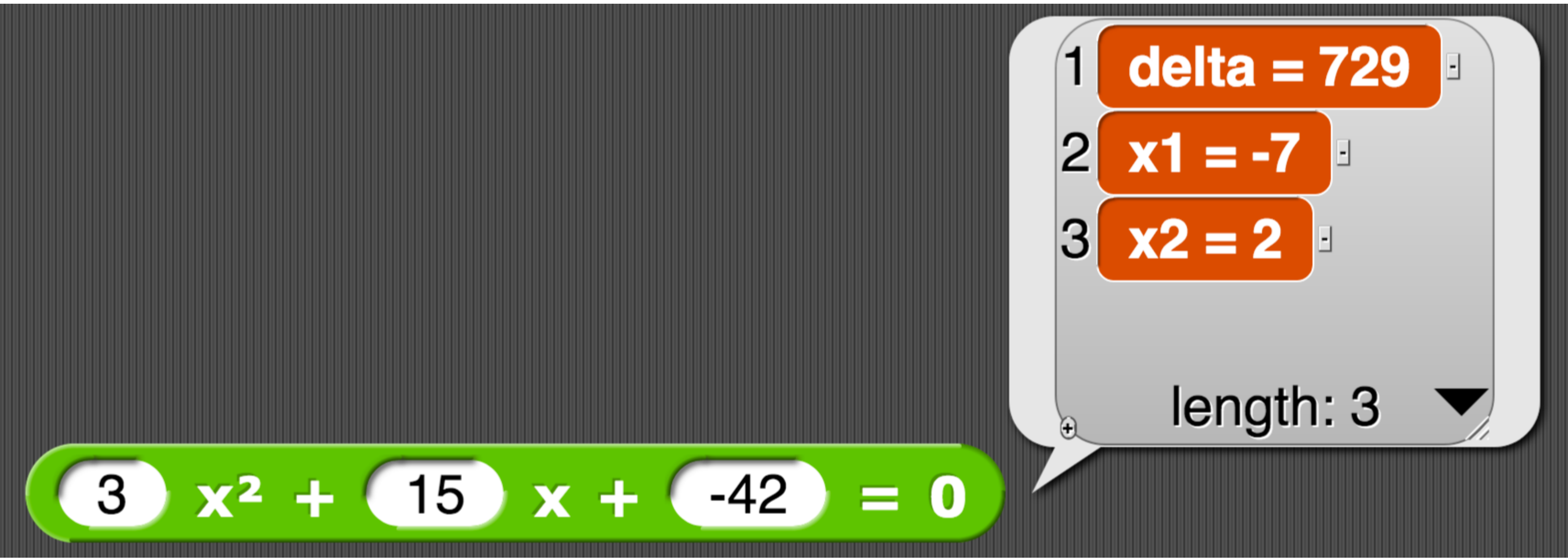
4) Une fonction qui renvoie la forme canonique

5) Une fonction qui renvoie son signe

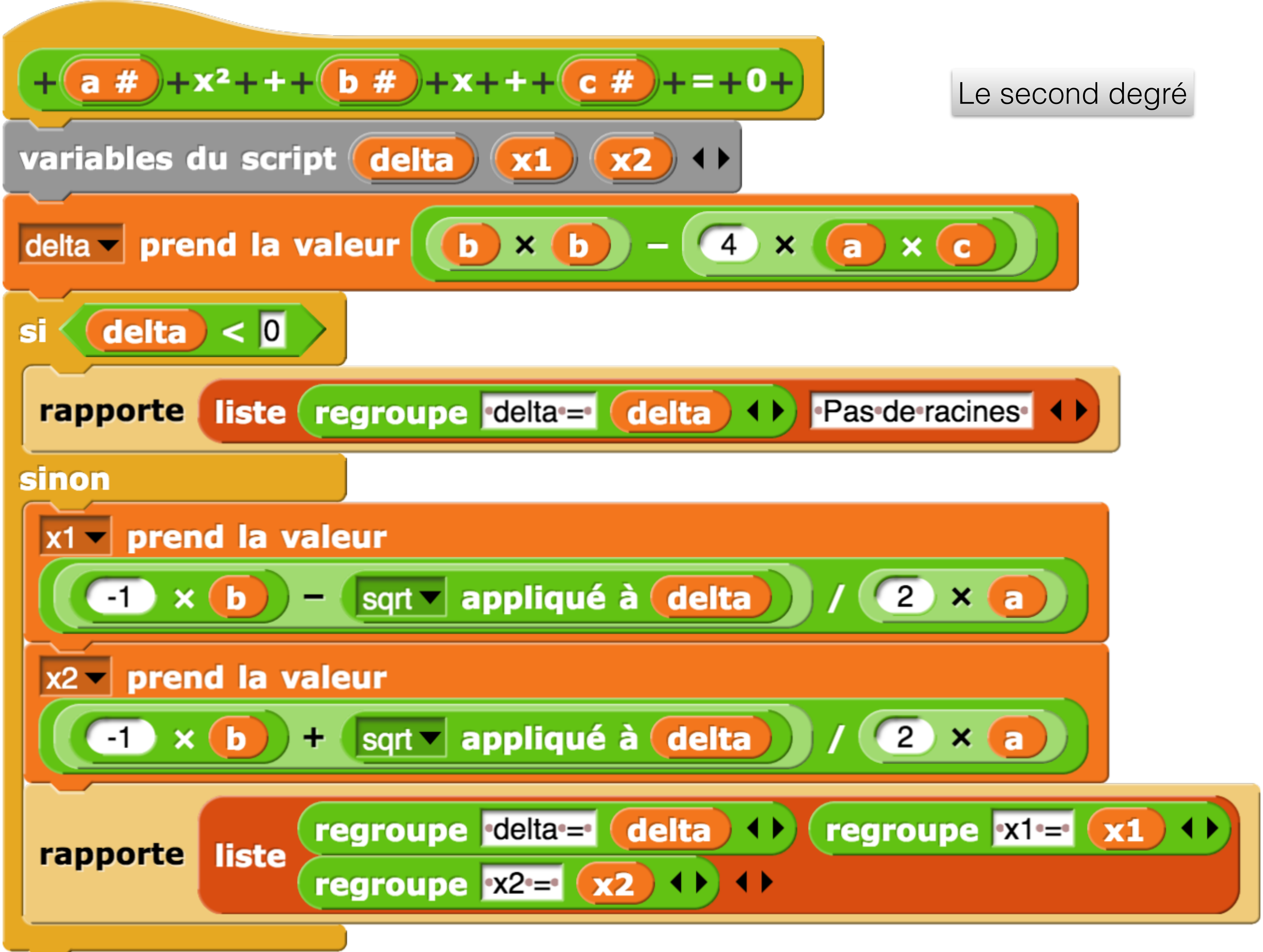
Scène de Snap!



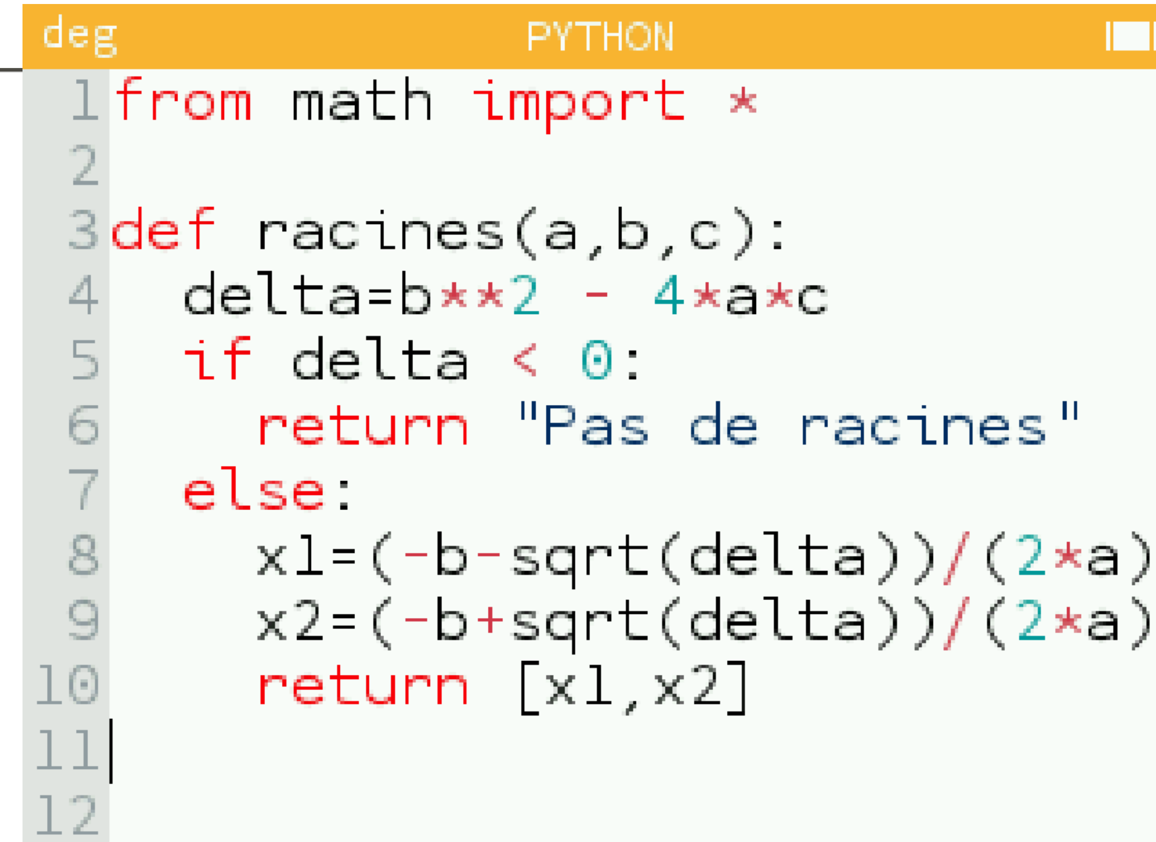
Le second degré



Le second degré



```python
from math import *

def delta(a,b,c):
        return b**2-4*a*c

def racines(a,b,c):
        delta = delta(a,b,c)
        if delta < 0:
                return "Pas de racines"
        else:
                x1 = (-b-sqrt(delta))/(2*a)
                x2 = (-b+sqrt(delta))/(2*a)
                return [x1,x2]

# appel aux fonctions delta et racines
delta(-3,2,1)
racines(-3,2,1)
```




```python
from math import *

def racines(a,b,c):
  delta=b**2 - 4*a*c
  if delta < 0:
    return "Pas de racines"
  else:
    x1=(-b-sqrt(delta))/(2*a)
    x2=(-b+sqrt(delta))/(2*a)
    return [x1,x2]
```

Le second degré

$$f(x) = ax^2 + bx + c$$

forme canonique $\quad f(x) = a(x-\alpha)^2 + \beta$

$$\alpha = -\frac{b}{2a} \qquad \beta = f(\alpha)$$

- Sommet de la parabole $\quad S(\alpha\,;\,\beta)$

Entrées : a, b, c $\rightarrow$ Calcul de α, β $\rightarrow$ Résultat $[\alpha, \beta]$

- Factorisation de f :

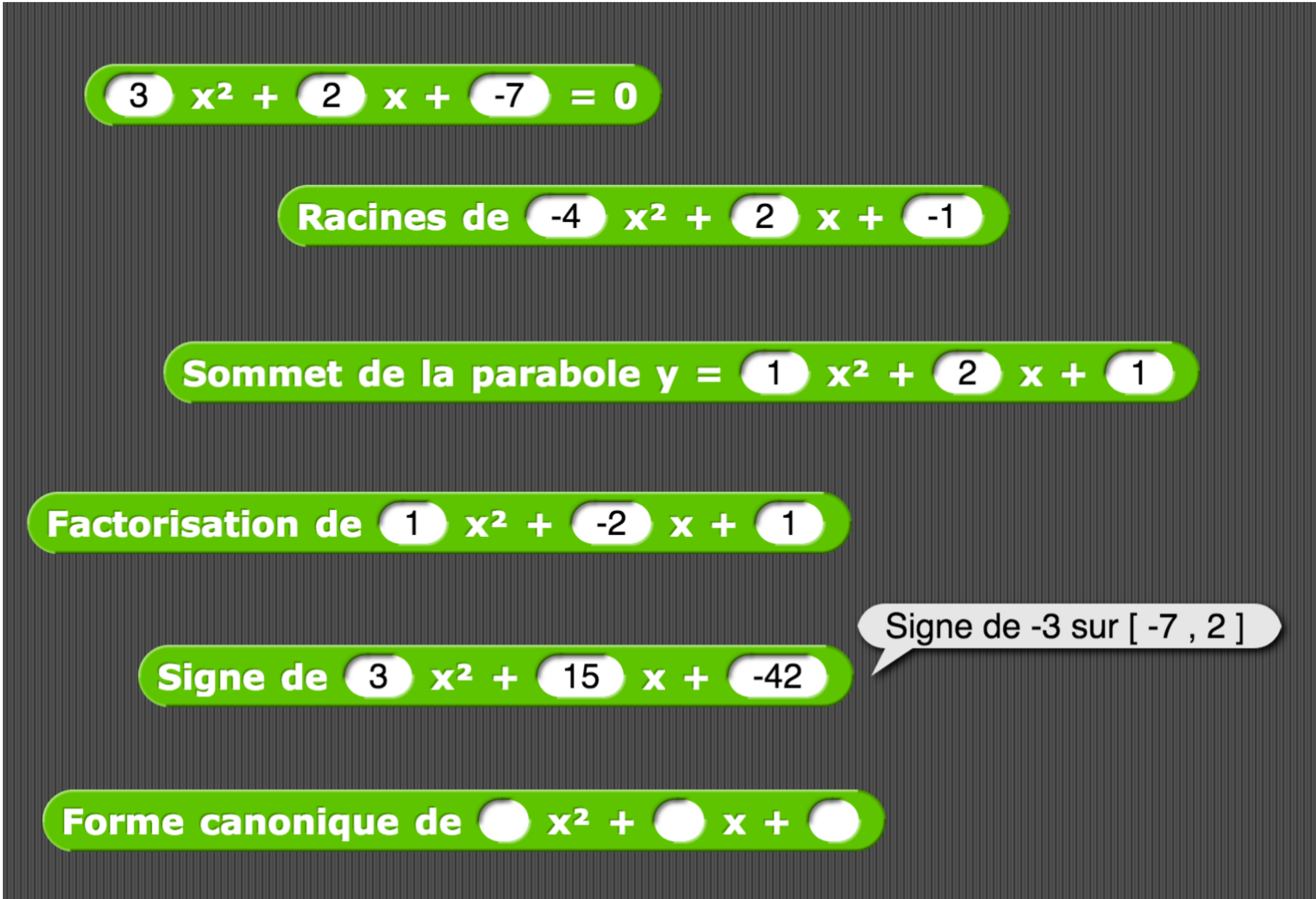
$\rightarrow$ si $\Delta < 0$ $\quad$ Pas de factorisation

$\rightarrow$ si $\Delta \geq 0$ $\quad$ x_1, x_2 (avec éventuellement $x_1 = x_2$)

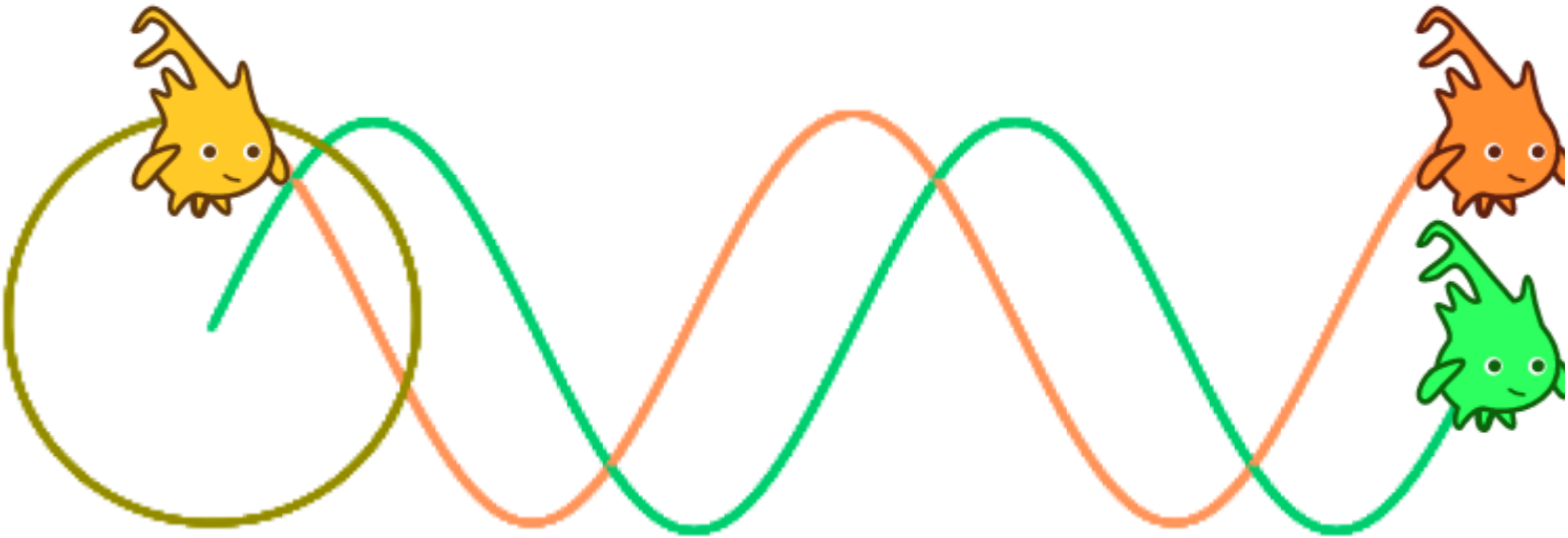
$$f(x) = a(x-x_1)(x-x_2)$$

Entrées : a, b, c $\rightarrow$ Factorisation de ax^2+bx+c $\rightarrow$ Résultat $a(x-x_1)(x-x_2)$ (qd $\Delta \geq 0$)

Le second degré

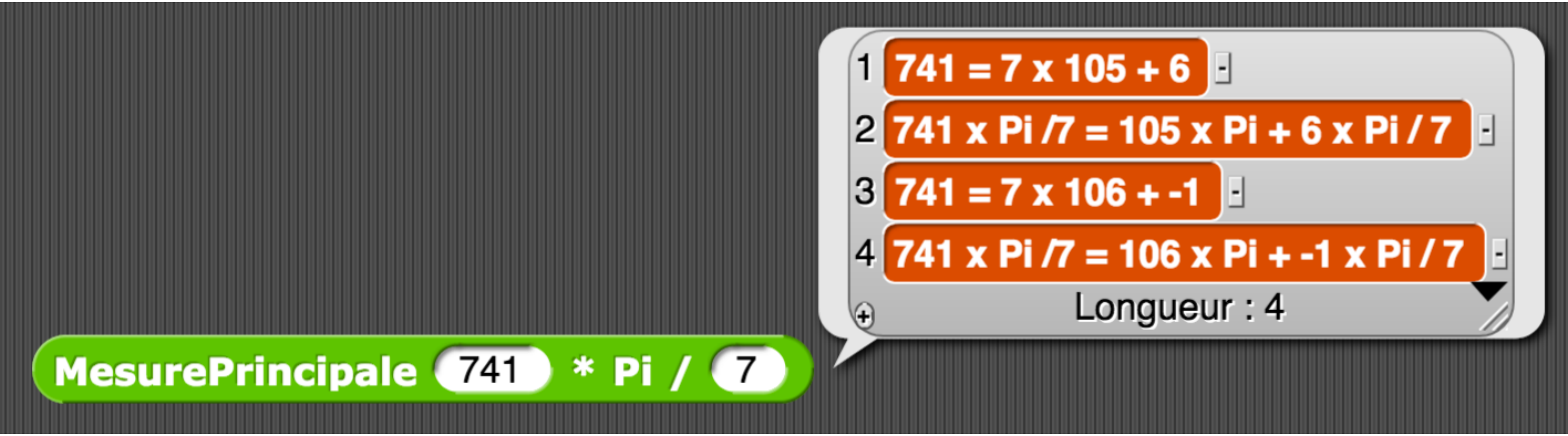


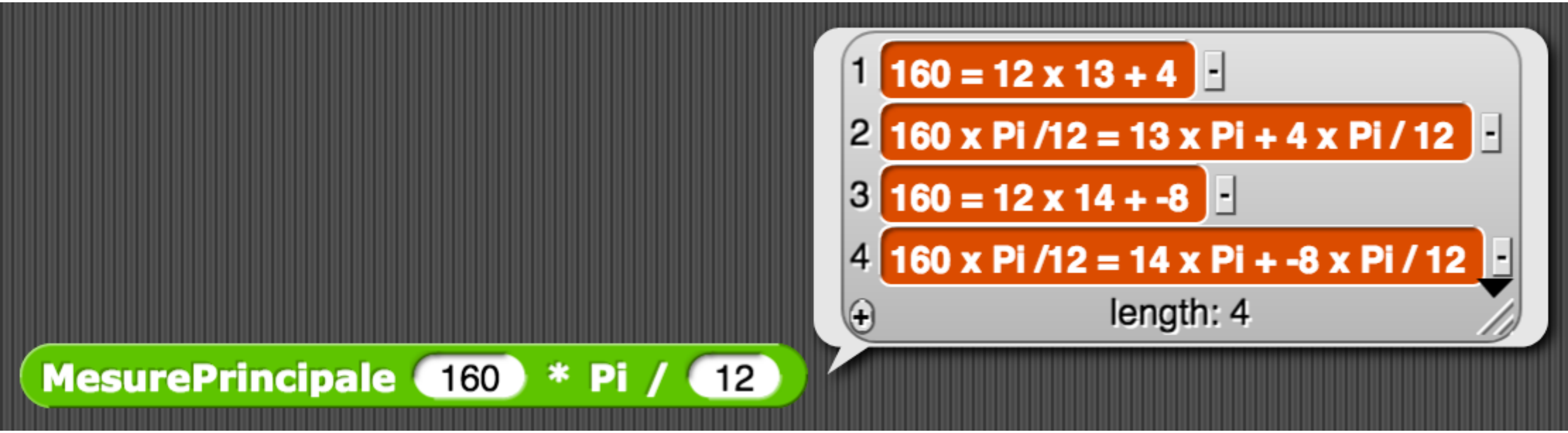
Trigonométrie



Lien vers l'animation *Snap!*

https://snap.berkeley.edu/snapsource/snap.html#present:Username=nathalierun&ProjectName=2018-09-21-SinusCosinus2


1 741 = 7 x 105 + 6
2 741 x Pi /7 = 105 x Pi + 6 x Pi / 7
3 741 = 7 x 106 + -1
4 741 x Pi /7 = 106 x Pi + -1 x Pi / 7
Longueur : 4
MesurePrincipale 741 * Pi / 7


1 160 = 12 x 13 + 4
2 160 x Pi /12 = 13 x Pi + 4 x Pi / 12
3 160 = 12 x 14 + -8
4 160 x Pi /12 = 14 x Pi + -8 x Pi / 12
length: 4
MesurePrincipale 160 * Pi / 12

Amener les algorithmes aux données : essence de la programmation fonctionnelle…

Amener les opérateurs aux données : essence de la programmation fonctionnelle…

Amener les opérateurs aux données : essence de la programmation fonctionnelle…

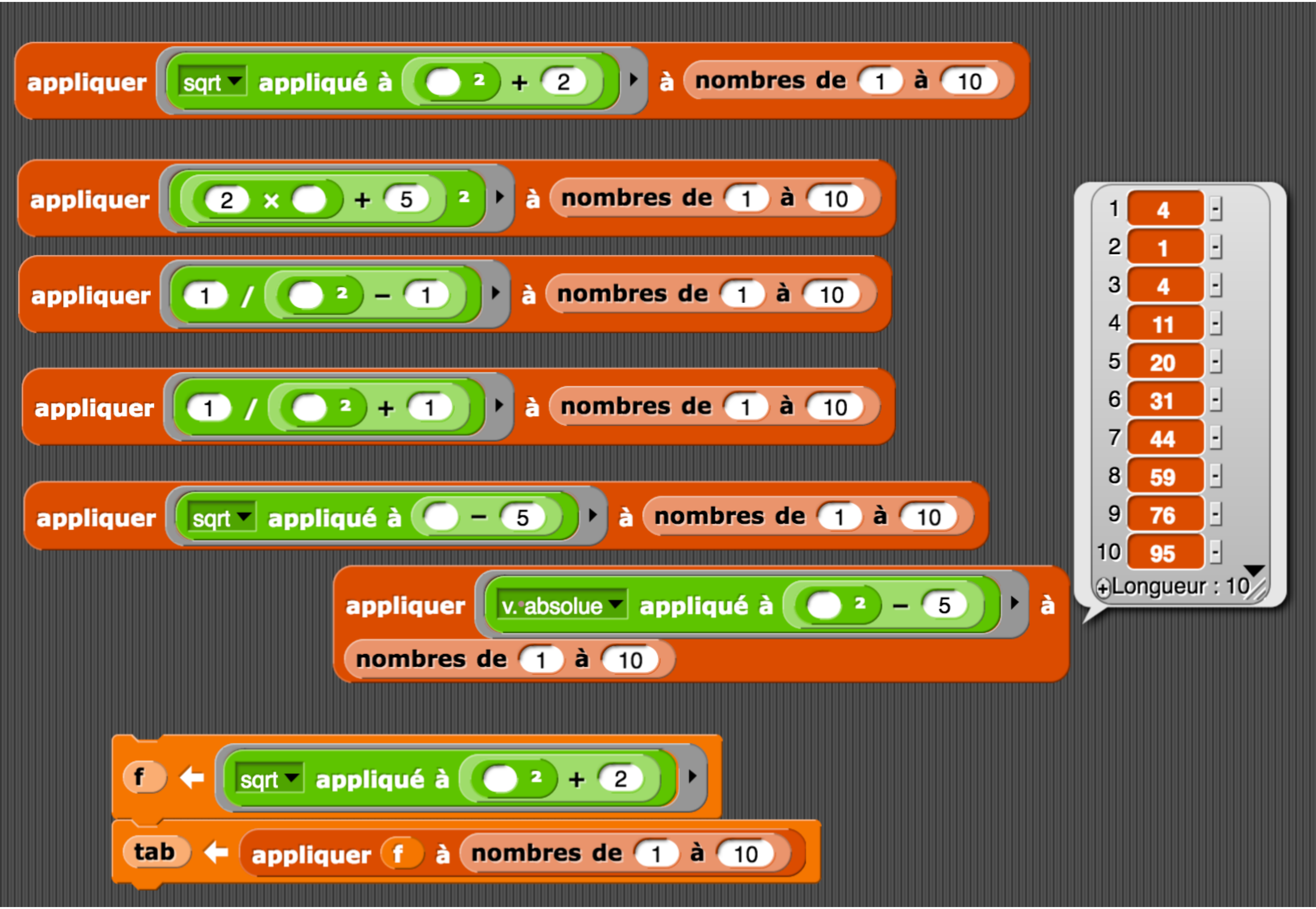
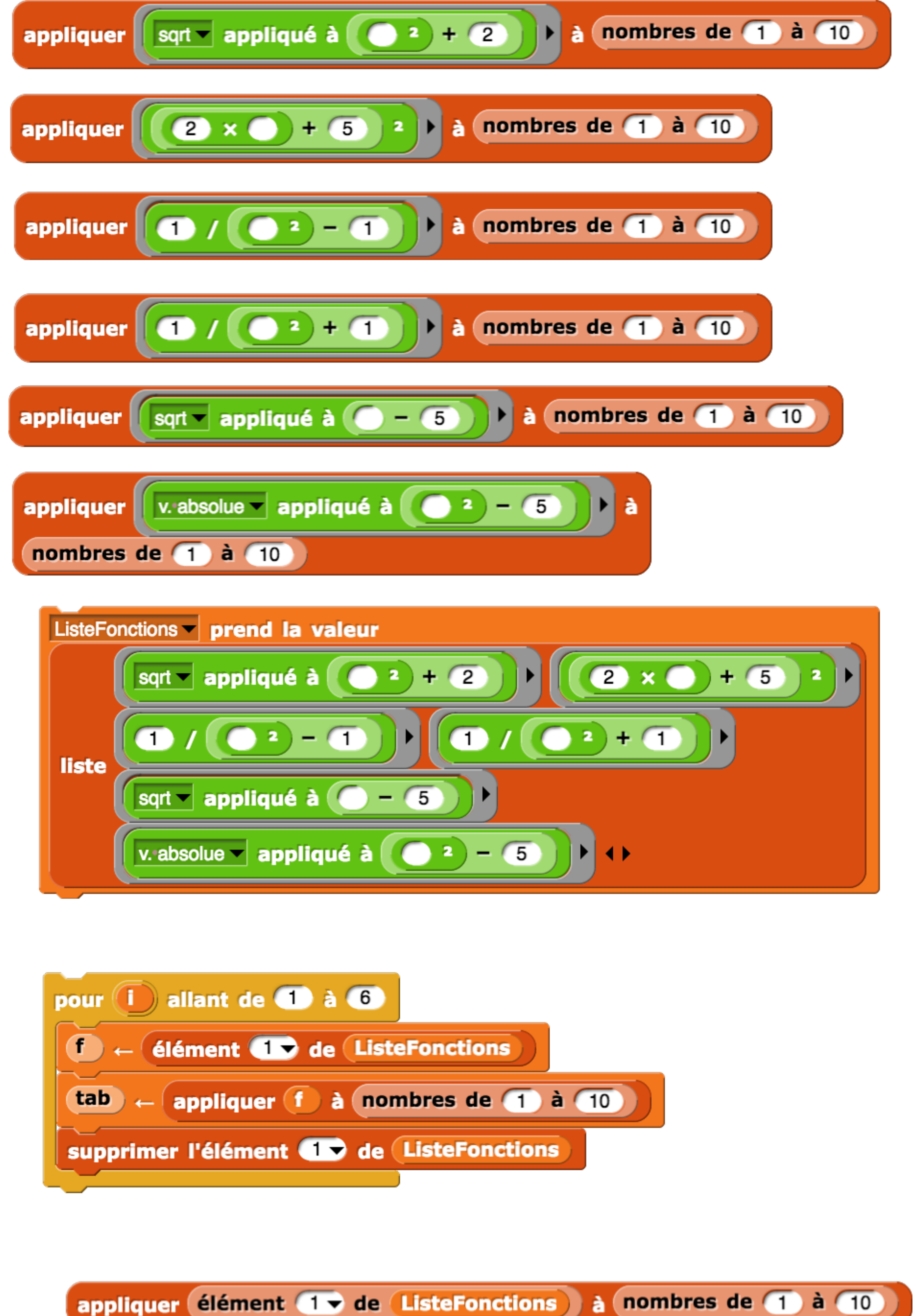


Tableau de valeurs
Fonctions associées

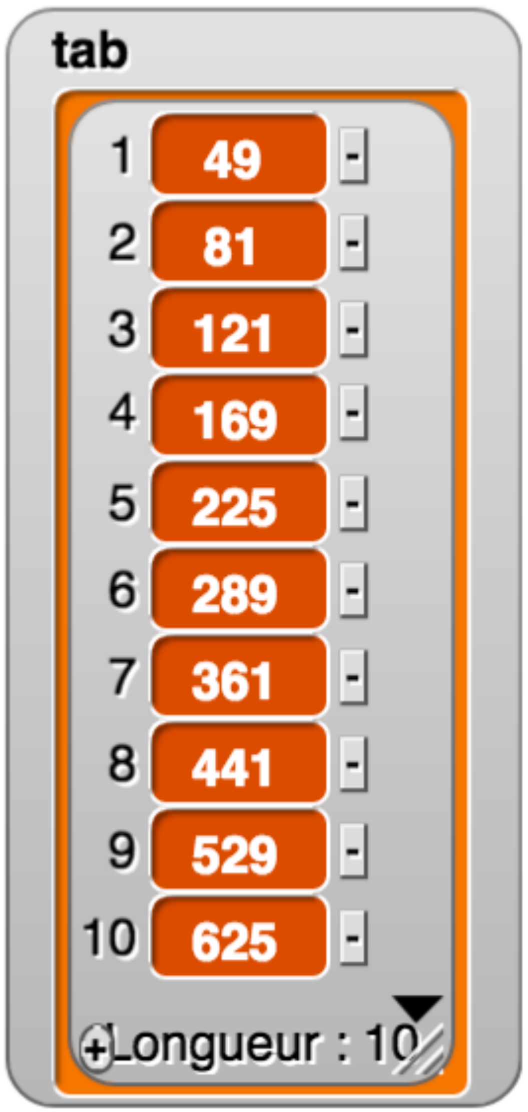
Faire un tableau de valeurs devient un jeu d'enfants avec les listes de Snap!

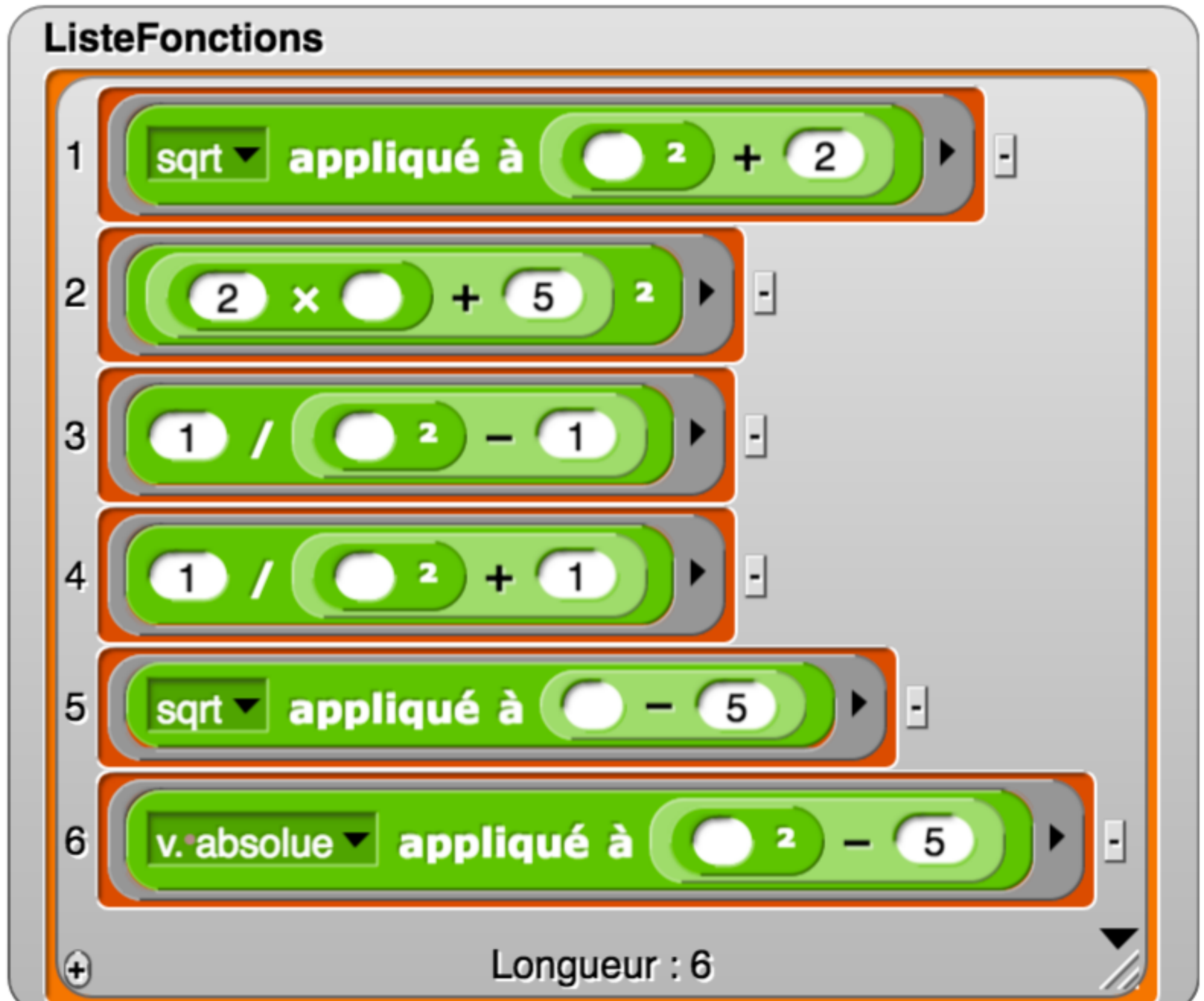
Tableau de valeurs
Fonctions associées

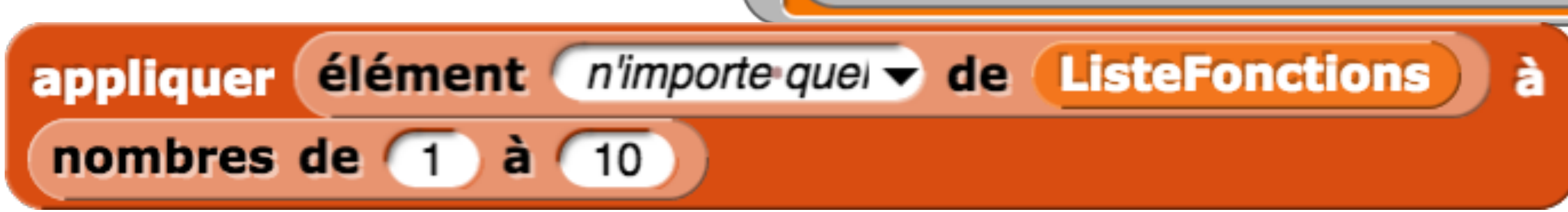


De quelle fonction s'agit-il ?

Tableau de valeurs
Fonctions associées







Lien vers le programme *Snap!*

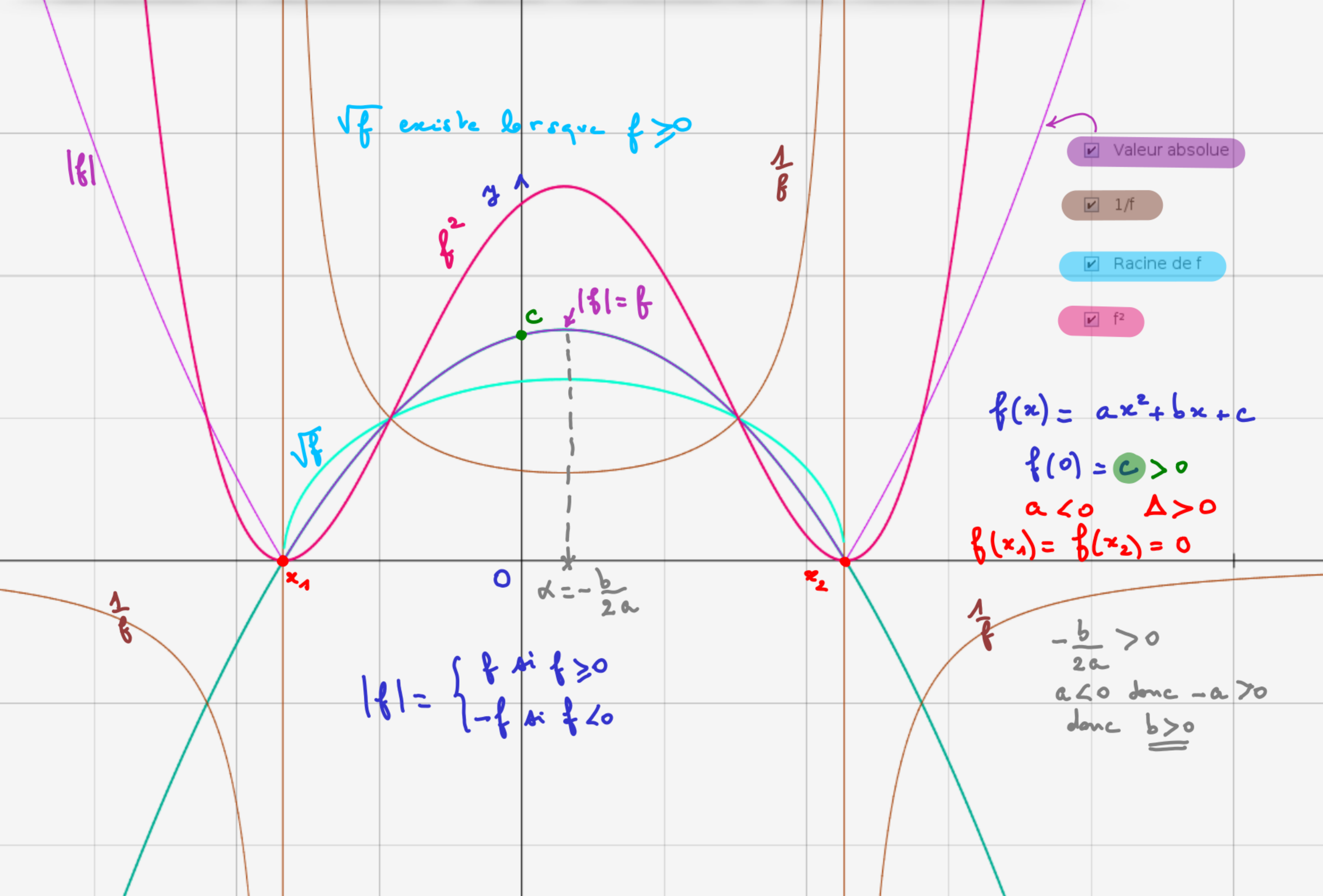
FonctionsComposees.py

```python
from random import *

a = 2.
b = 5.

def u1(x):
    return x*x

def u2(x):
    return 1/x

def u3(x):
    return sqrt(x)

def u4(x):
    return a*x+b

def u5(x):
    return abs(x)

liste_fonctions = [u1,u2,u3,u4,u5]

def f(x):
    u = choice(liste_fonctions)
    v = choice(liste_fonctions)
    return u(v(x))

liste = [1,2,3,4,5,6,7,8,9,10]
liste2 = map(f,liste)
```

Jeu sur les fonctions avec Python

```
                Jeu avec Python

On applique le script Python suivant dans un terminal :

=====================================================
from math import *
from random import *

a = 2.
b = 5.

def u1(x):
        return x*x

def u2(x):
        return 1/x

def u3(x):
        return sqrt(x)

def u4(x):
        return a*x+b

def u5(x):
        return abs(x)

liste_fonctions = [u1,u2,u3,u4,u5]

def f(x):
        u = choice(liste_fonctions)
        v = choice(liste_fonctions)
        return u(v(x))

liste = [1,2,3,4,5,6,7,8,9,10]
liste2 = map(f,liste)
=====================================================

Avec le résultat obtenu suivant :

>>> liste2
[49.0, 2, 3.3166247903554, 5.0, 0, 0, 19.0, 0.0, 23.0, 3.1622776601683795]


Quelles ont été les fonctions u et v tirées au hasard ?
```

Fonctions déduites des fonctions de références



Devoir sur les fractales

pour allier

Dessin papier

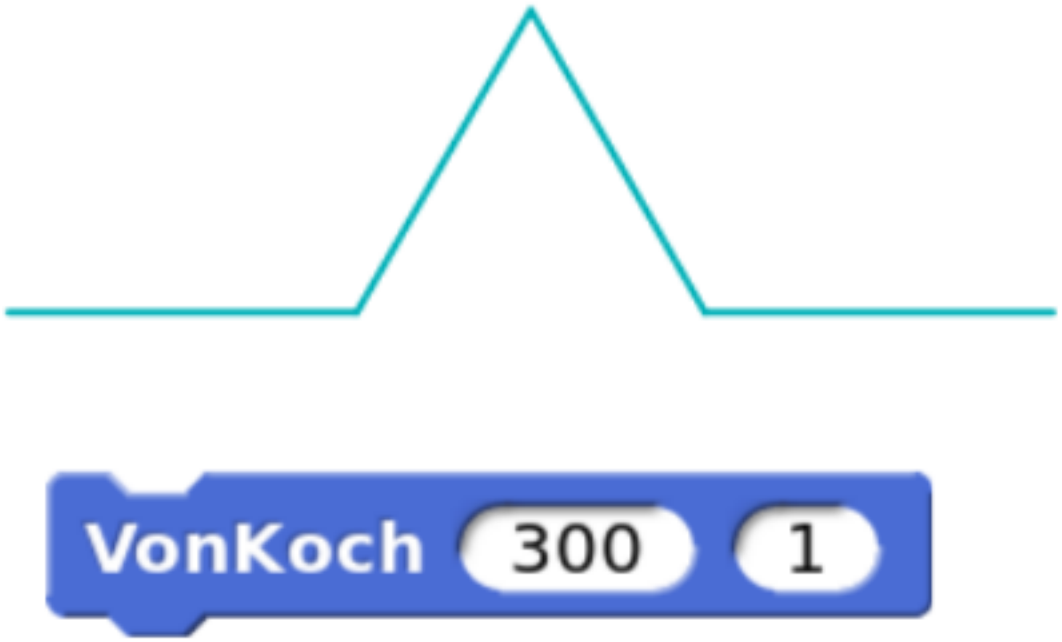
Algorithmes et programmation

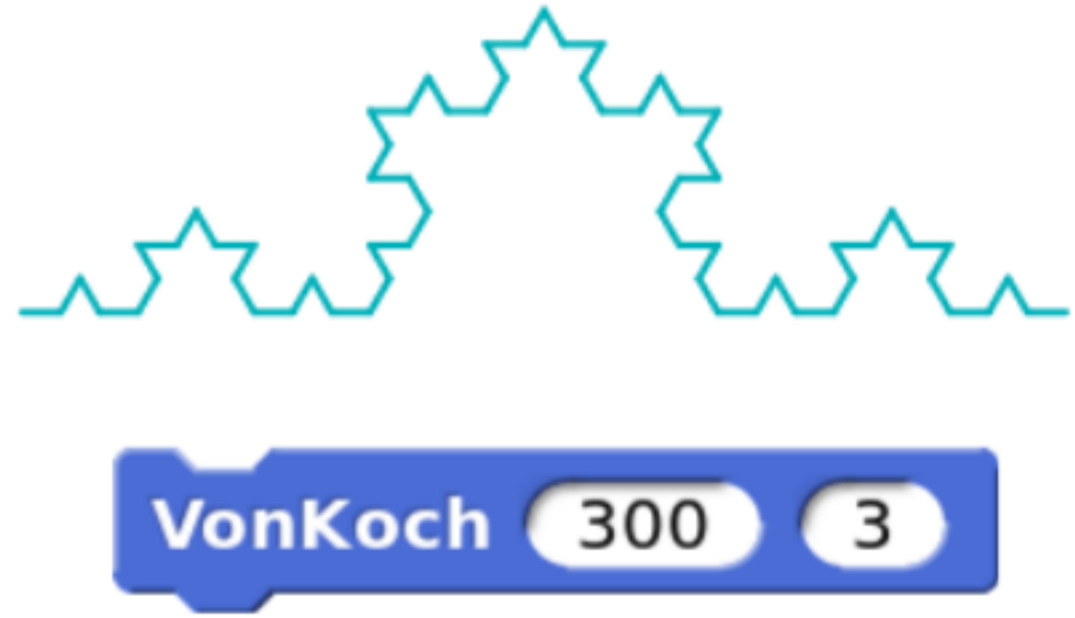
Géométrie, orientation dans le plan

Calculs et suites numériques

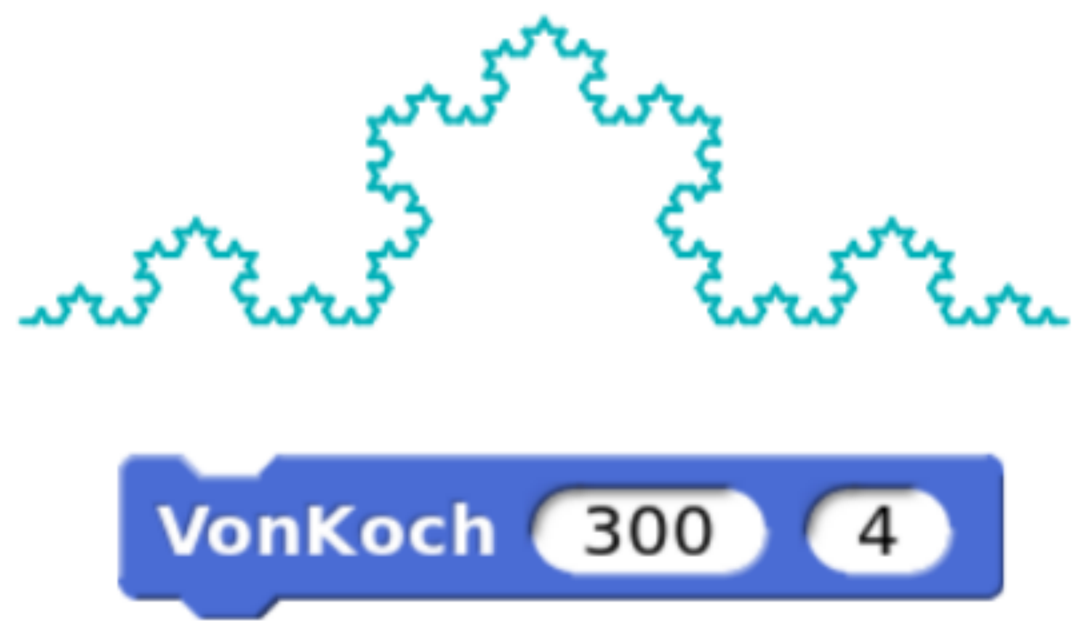
Plan du devoir

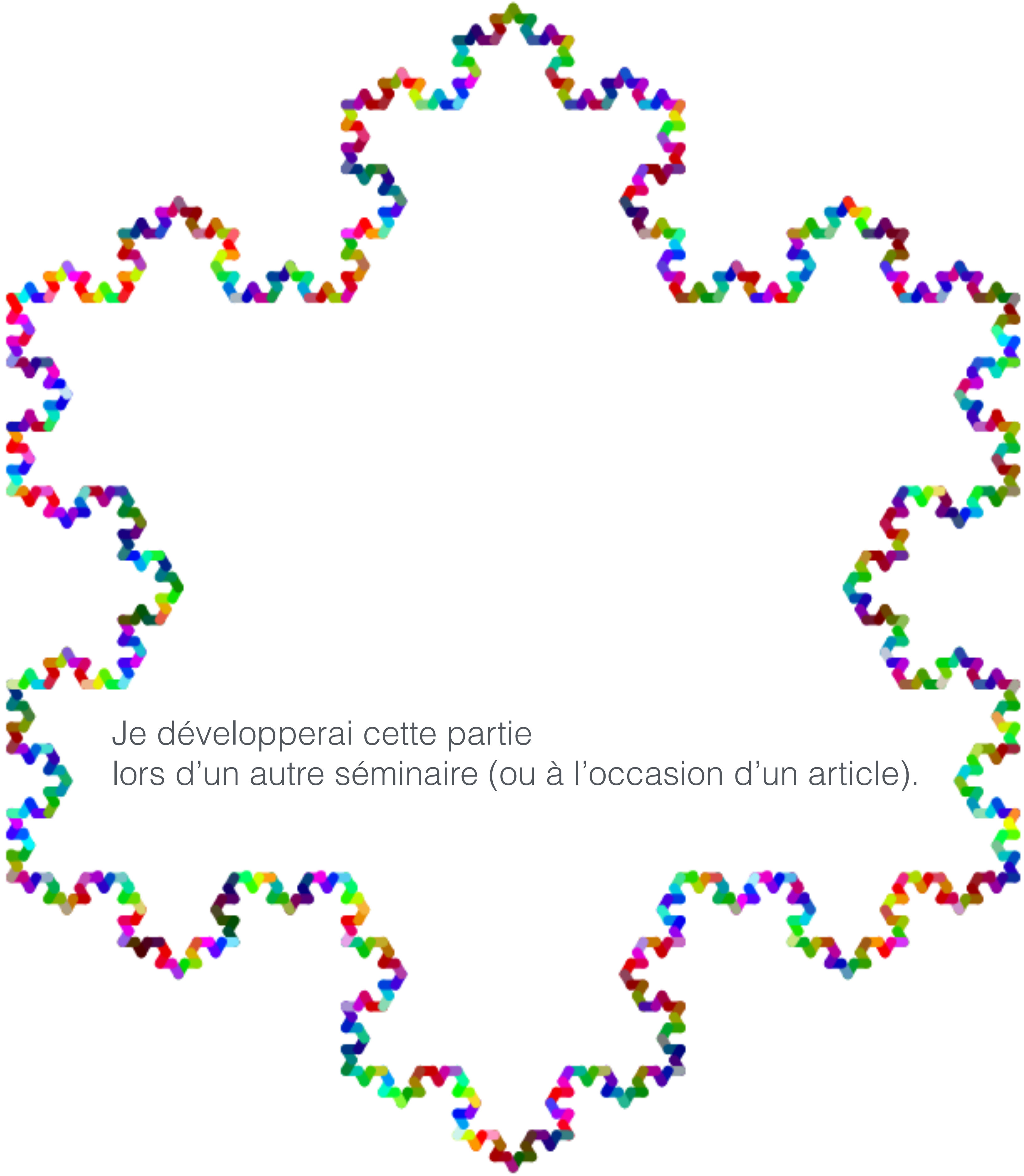
Données, initialisation
PARTIE DESSIN PAPIER
PREMIERS CALCULS
PARTIE ALGORITHMIQUE
PROGRAMMATION
RETOUR SUR LA PARTIE CALCULS





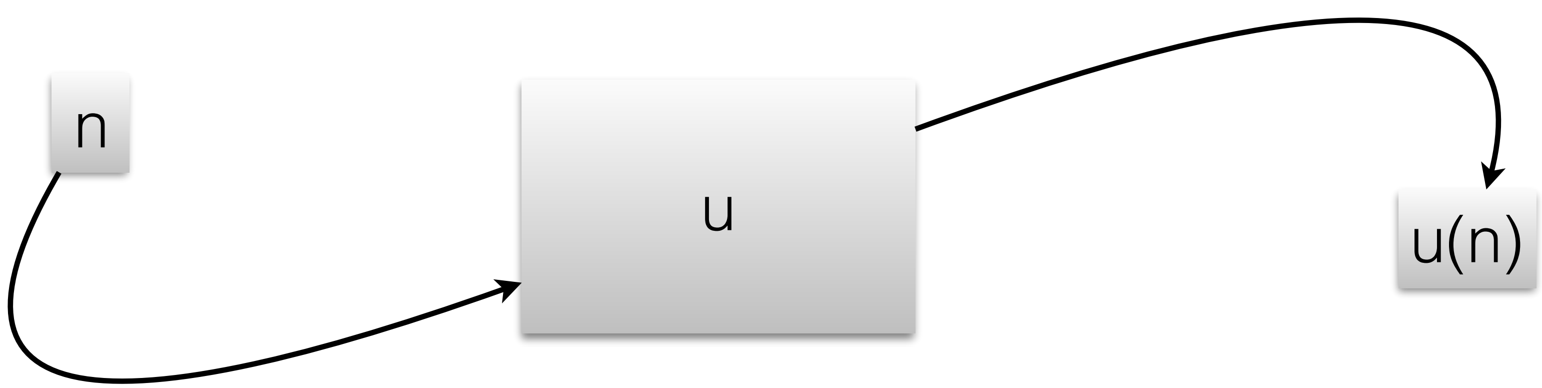
Faire découvrir aux élèves la notion de récursivité…



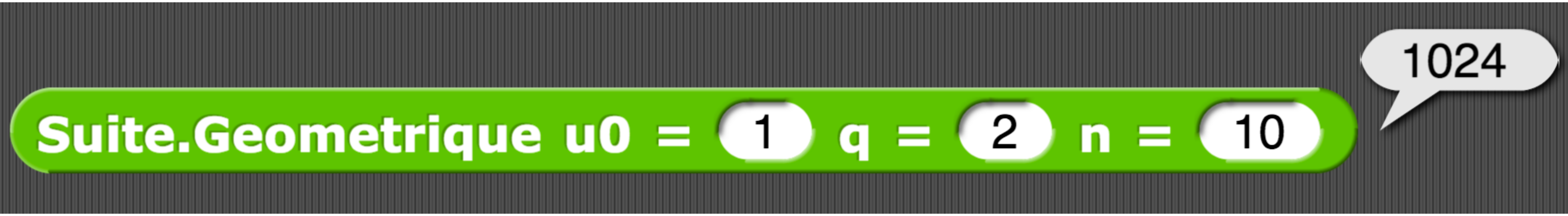


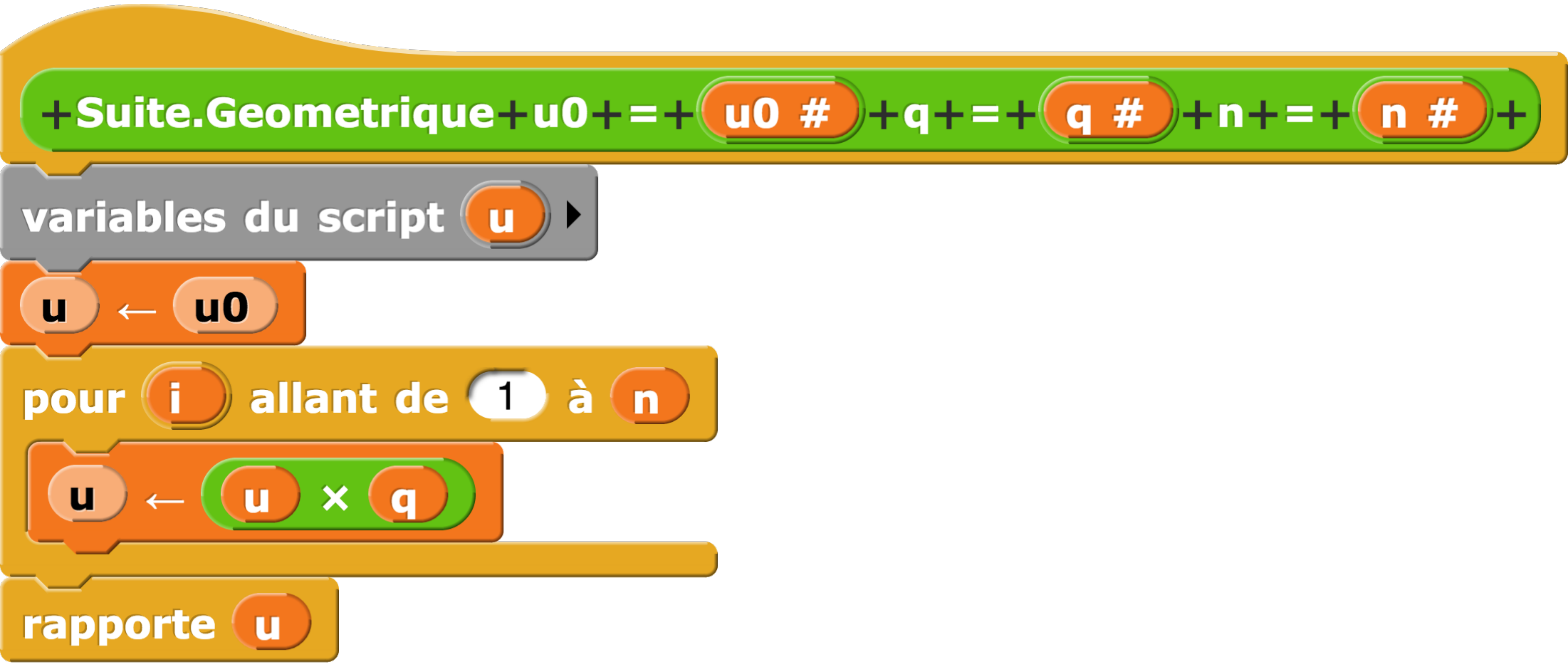
Je développerai cette partie
lors d'un autre séminaire (ou à l'occasion d'un article).

Exemples de suites numériques

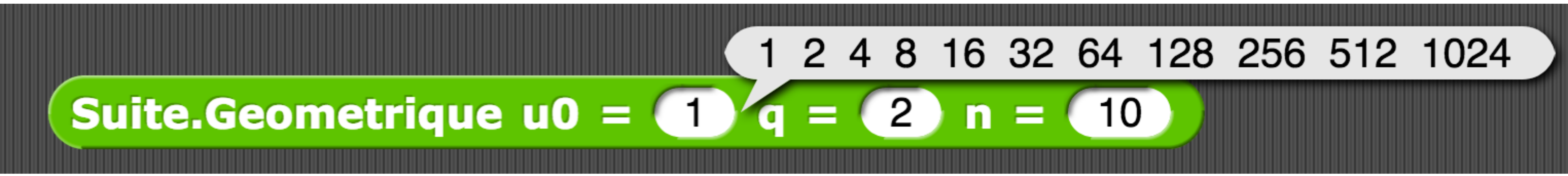


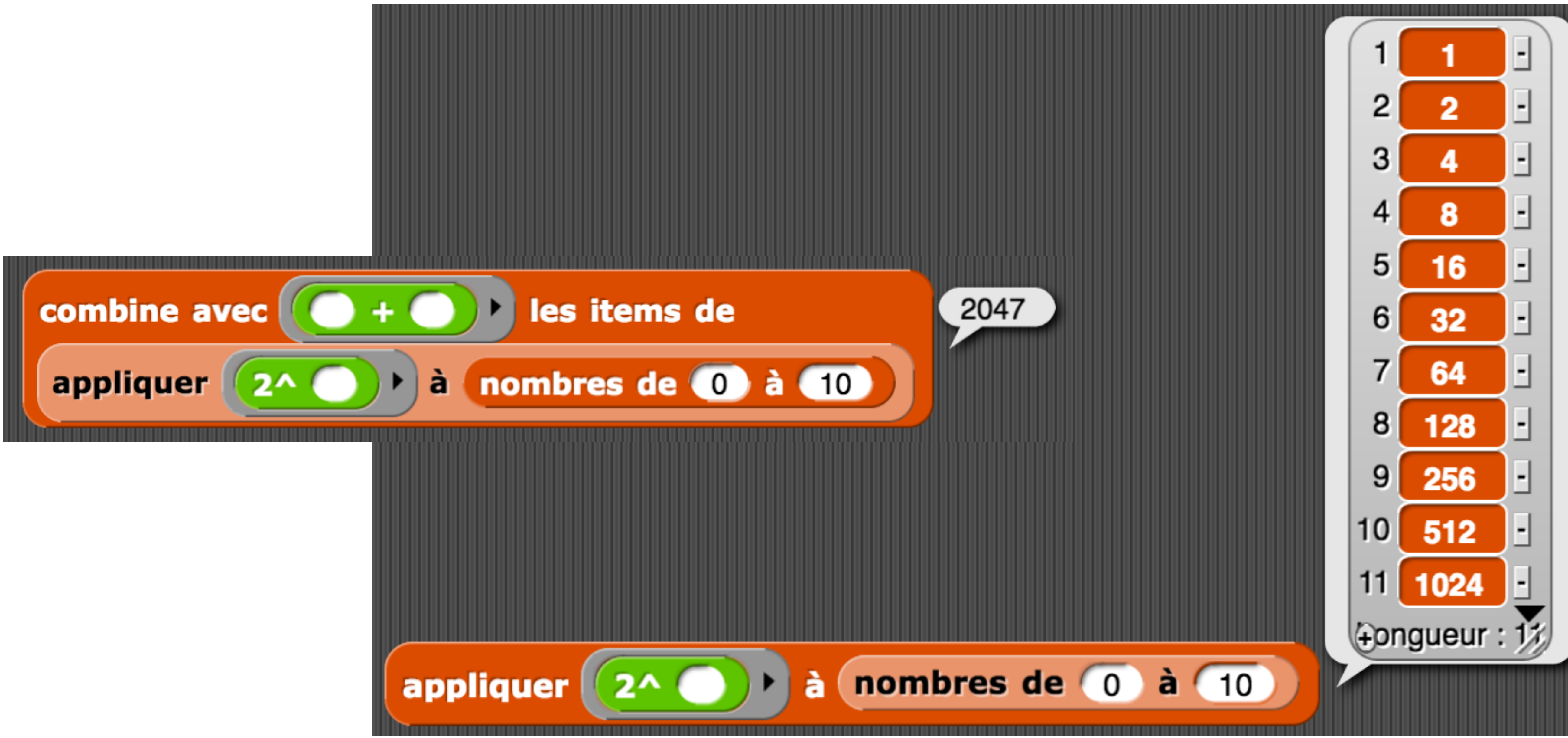
Exemples de suites numériques





Exemples de suites numériques





Exemples de suites numériques

$$\sum_{k=1}^{n} k$$

Exemples de suites numériques

$$\sum_{k=1}^{n} k^2$$

$$\sum_{k=1}^{n} k^3$$

Travail en géométrie analytique

En seconde a été vu :

Milieu de 2 points

Distance entre 2 points

Relation de Chasles

Coordonnées du symétrique d'un point par rapport à un autre.

Équation d'une droite qui passe par 2 points

En première S

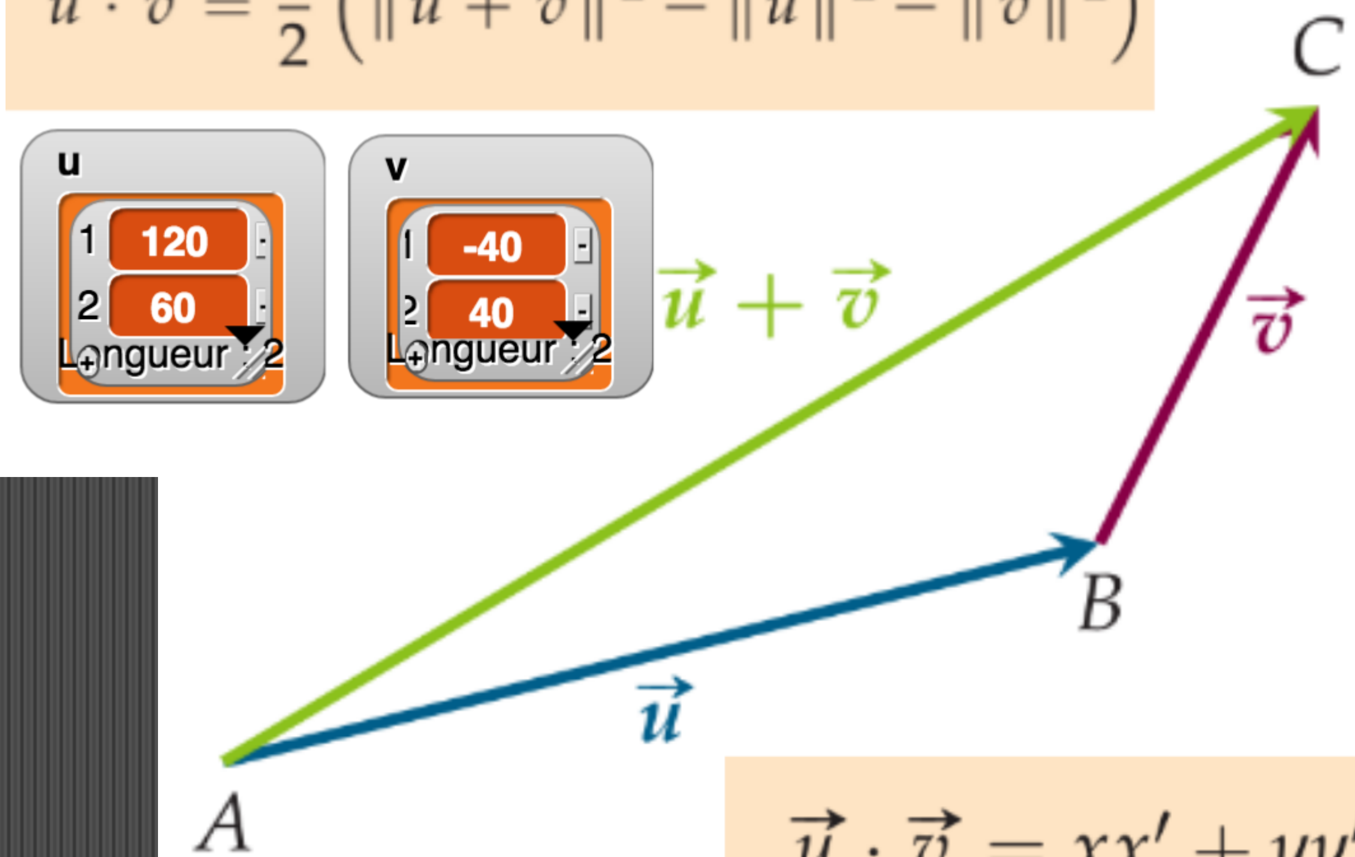
Produit scalaire de 2 vecteurs

Colinéarité de 2 vecteurs

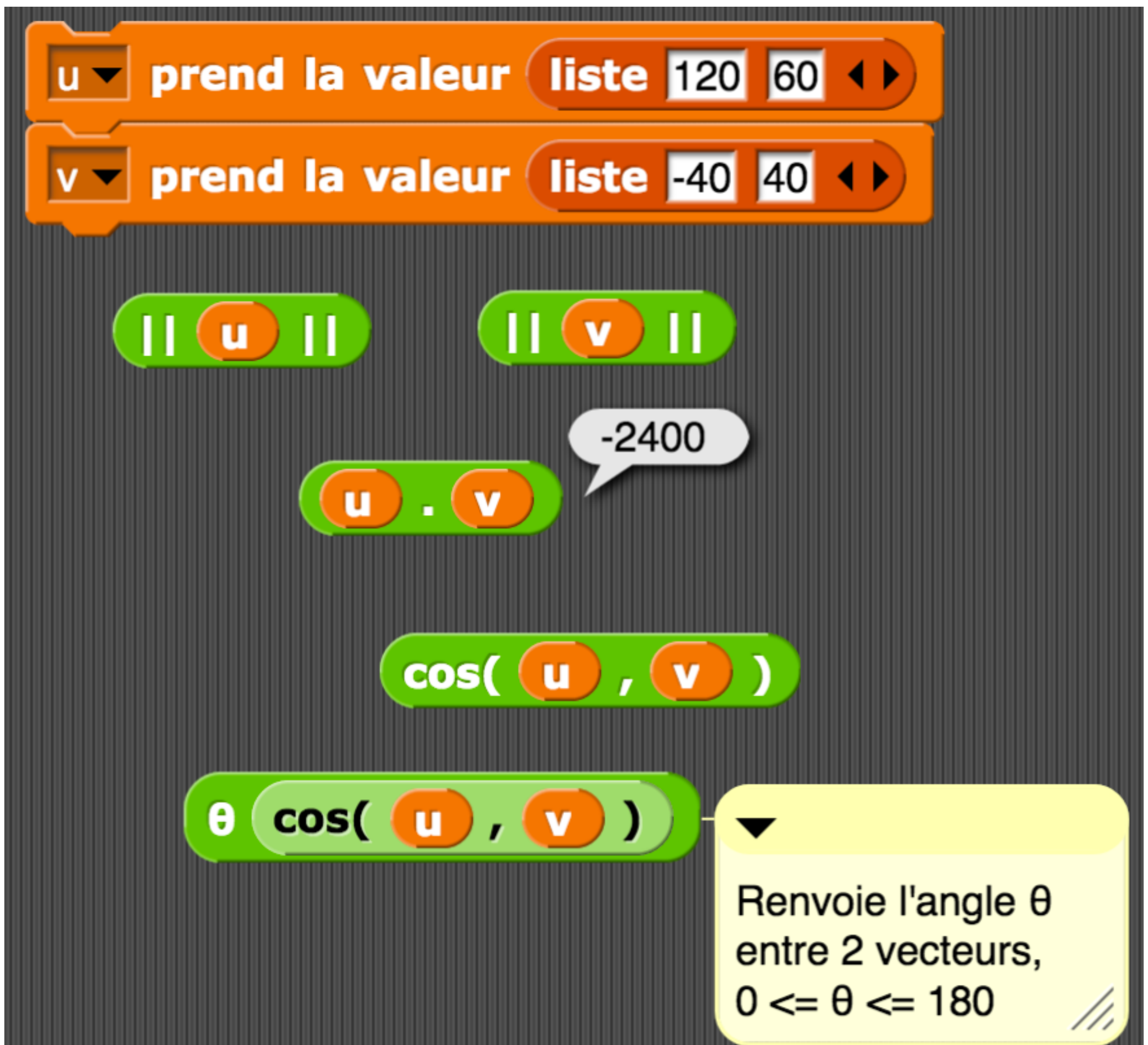
Équation d'une droite normale à une direction et passant par 1 point

Produit scalaire

$$\vec{u} \cdot \vec{v} = \frac{1}{2}\left(\|\vec{u}+\vec{v}\|^2 - \|\vec{u}\|^2 - \|\vec{v}\|^2\right)$$



$$\vec{u} \cdot \vec{v} = xx' + yy'$$

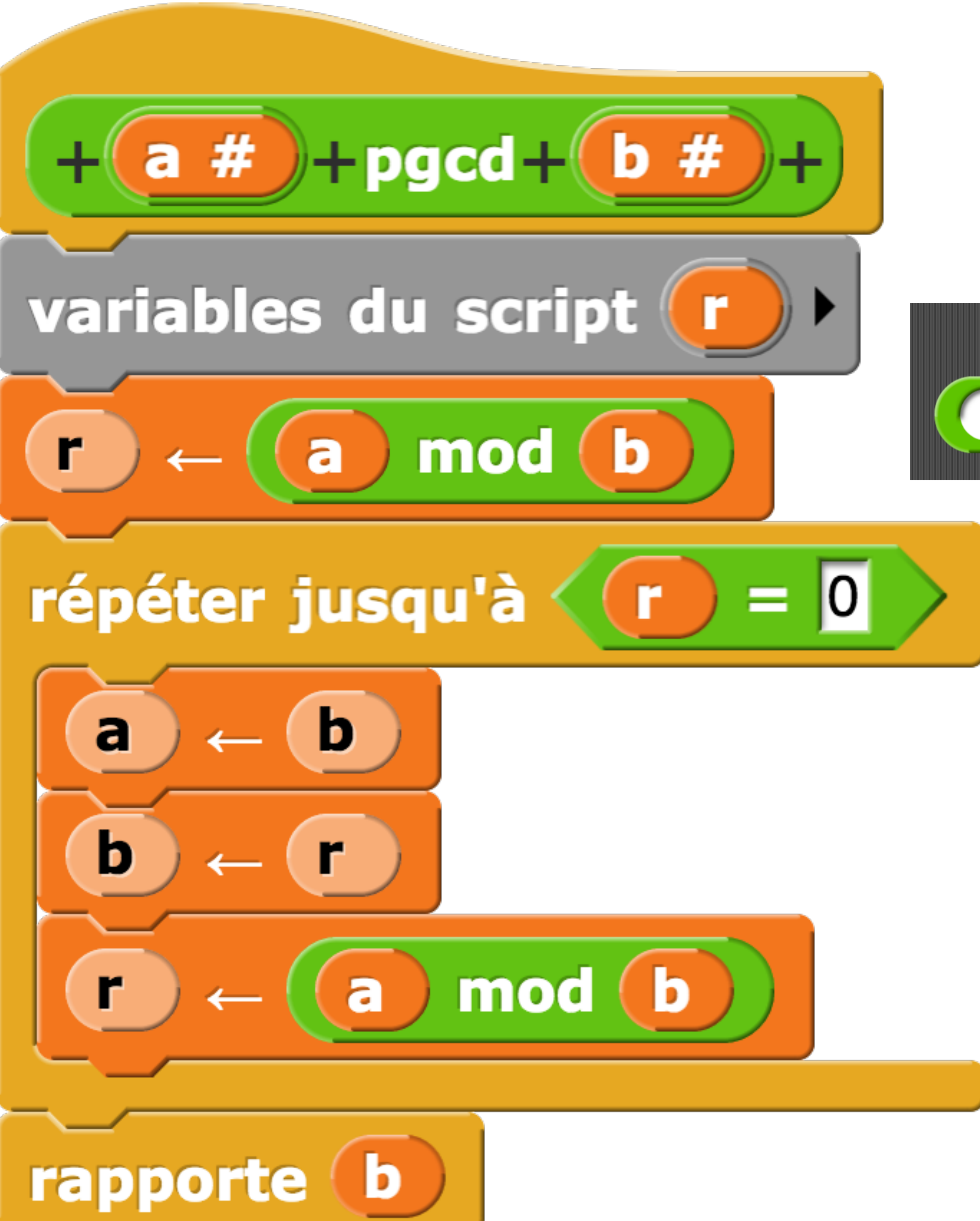


Algorithme d’Euclide,

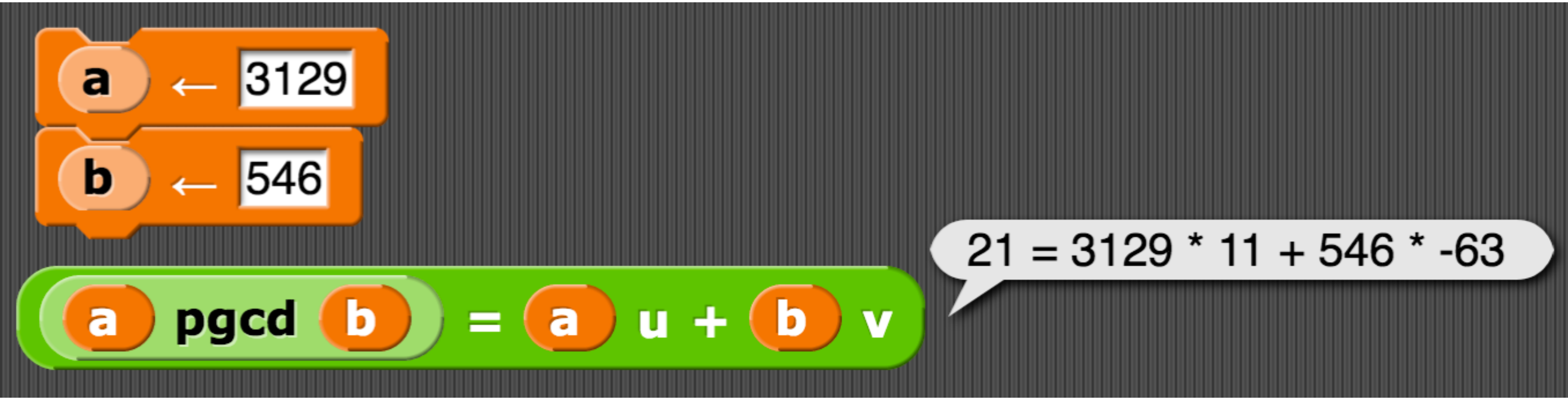
et un peu d’arithmétique…

L’arithmétique est un des domaines des mathématiques les plus appropriés pour commencer la programmation.

Algorithme d’Euclide



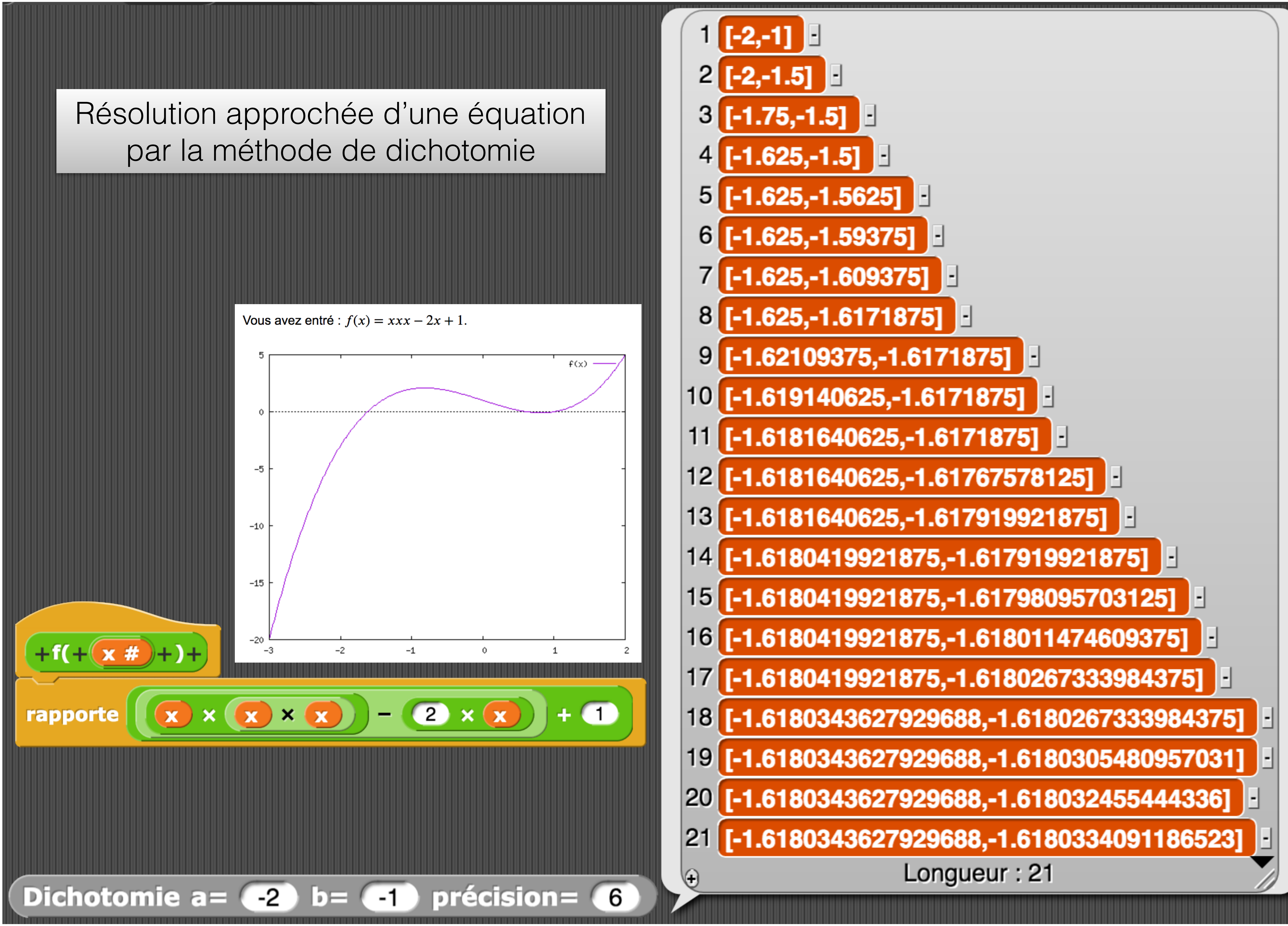
Égalité de Bézout

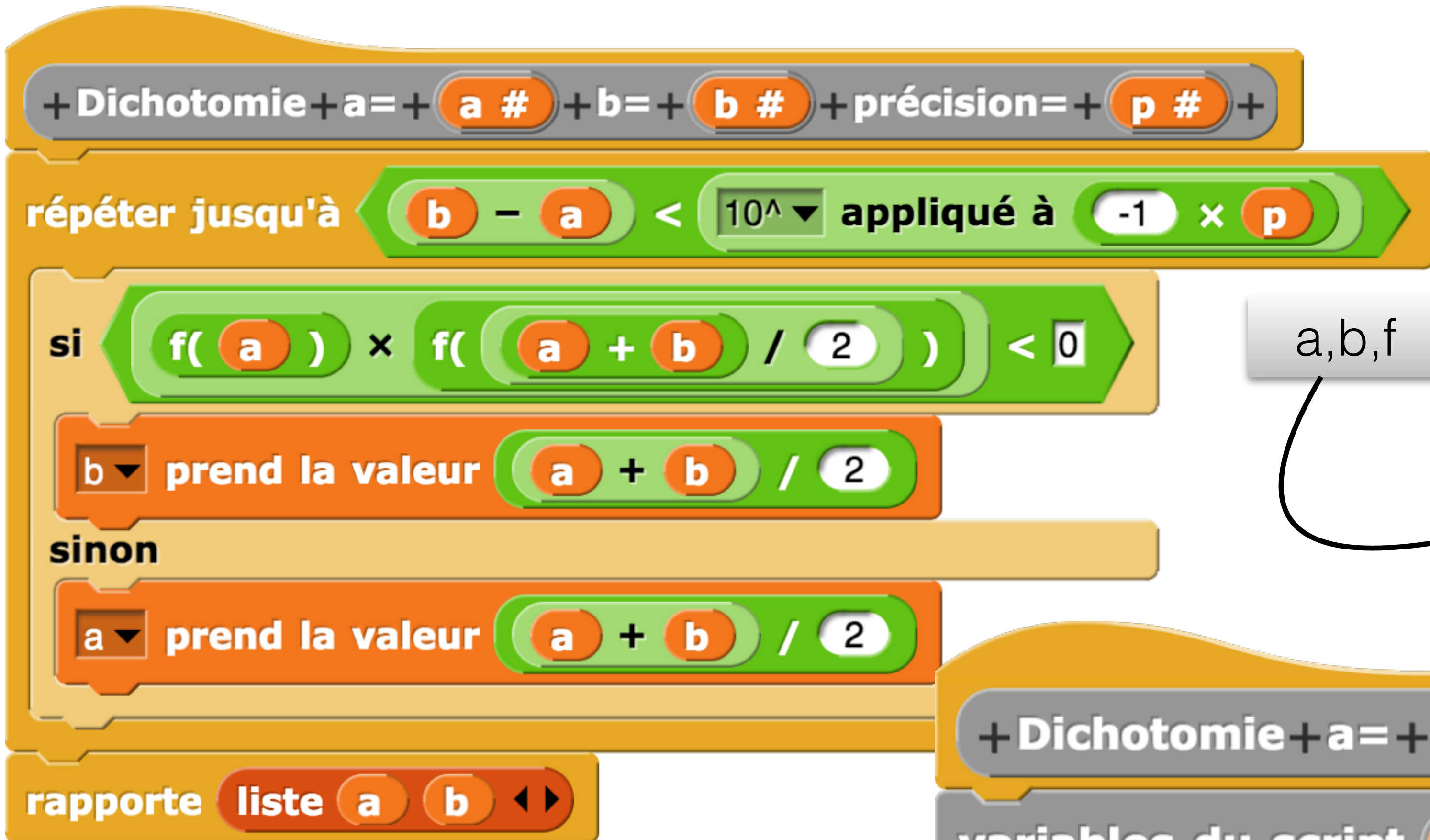


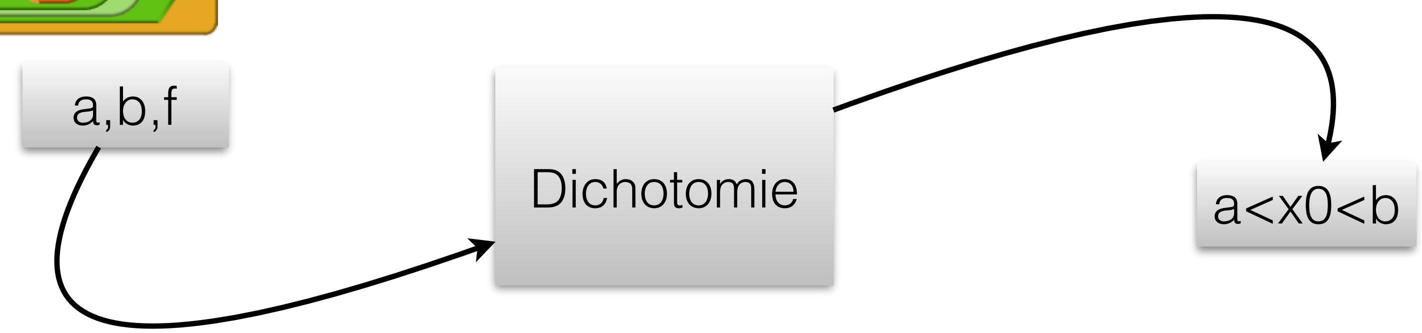
Algorithmes divers

illustrant grâce à *Snap!* la notion de fonction

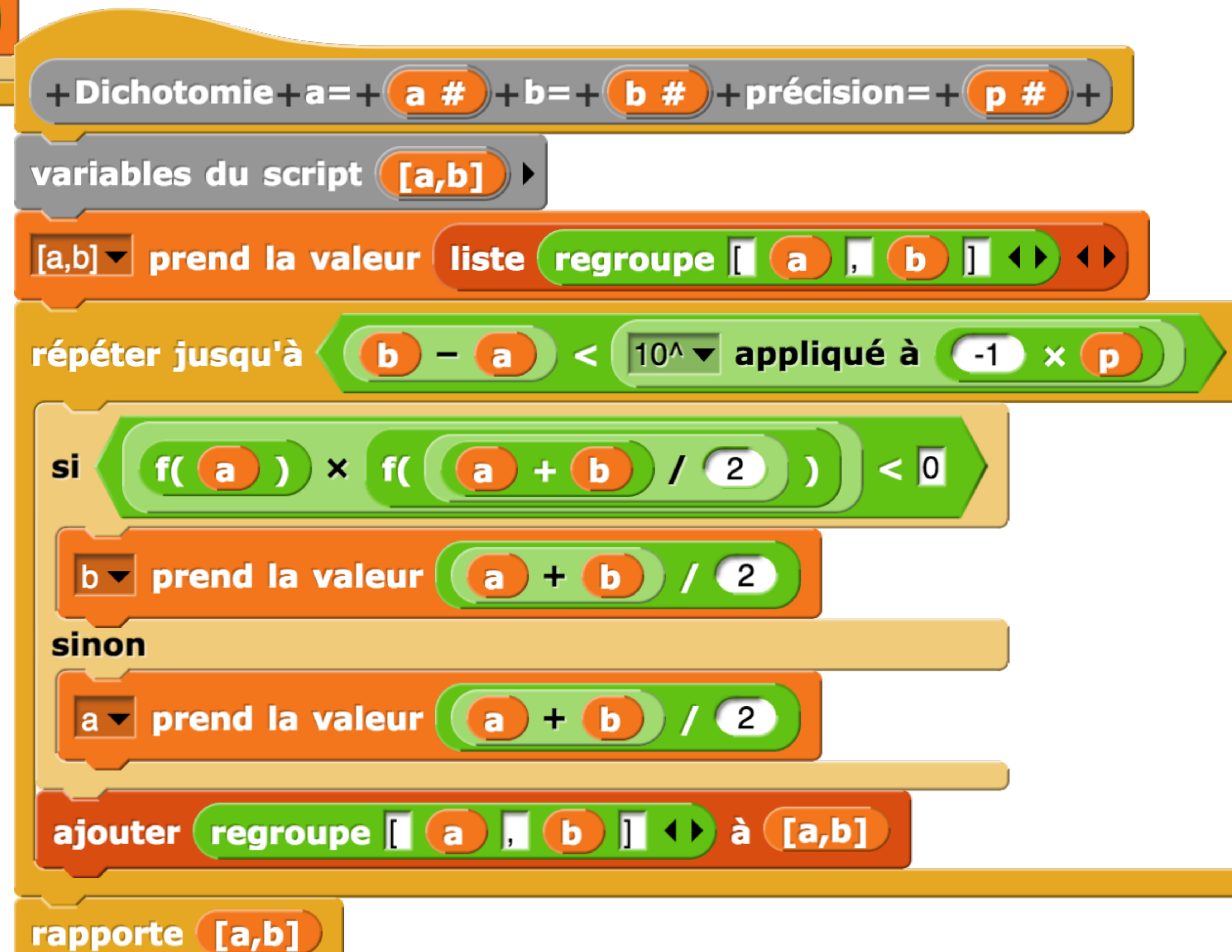
Résolution approchée d'une équation par la méthode de dichotomie







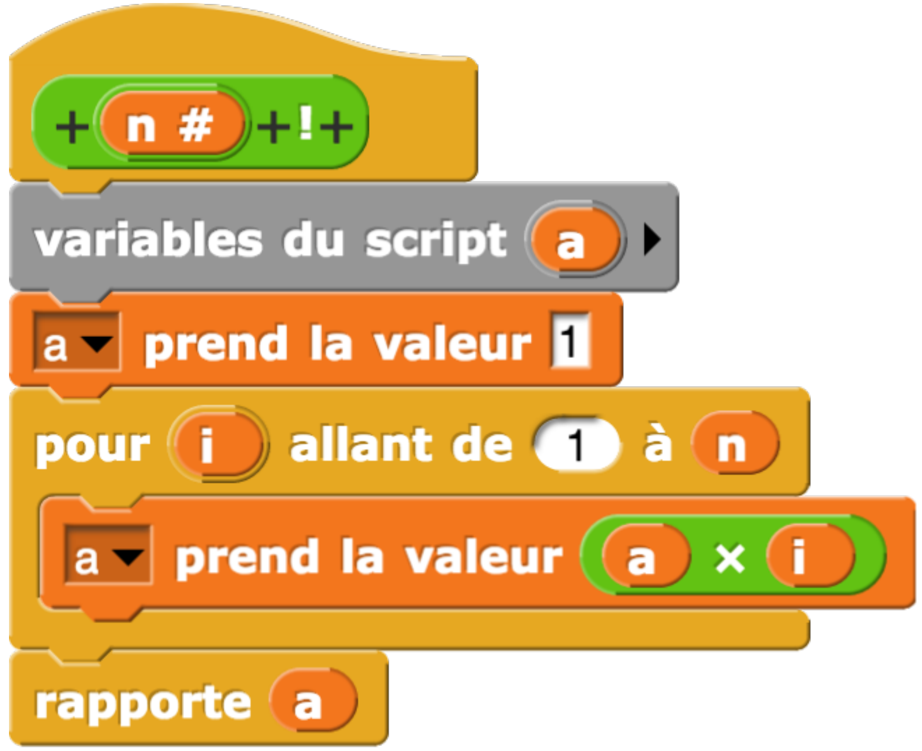
Retour où seule la solution encadrée avec la précision demandée est fournie.

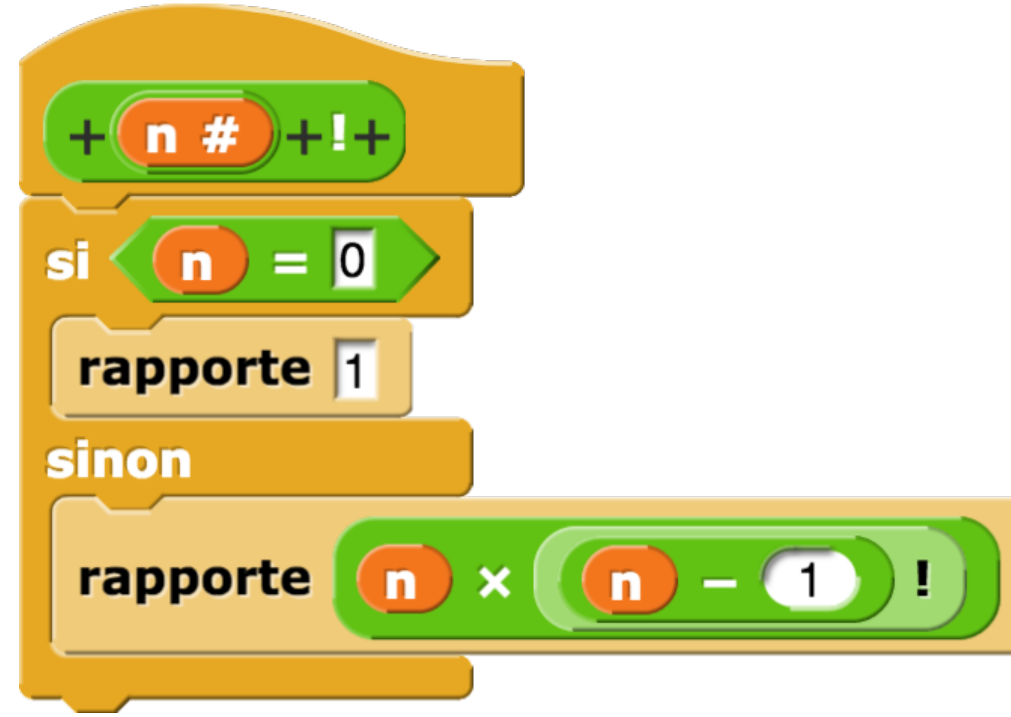


Évidemment, cet algorithme est perfectible et on peut avec Snap! entrer la fonction f comme argument au lieu de la définir séparément.

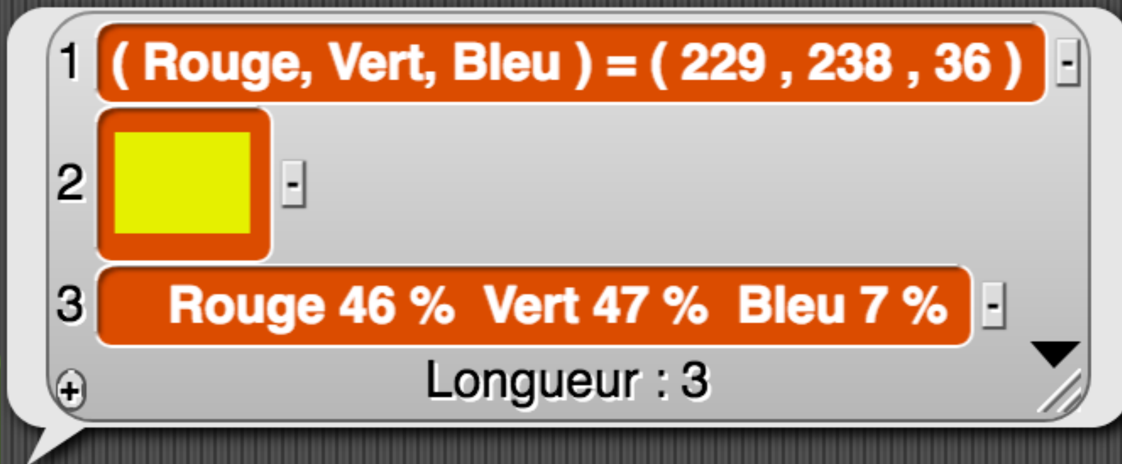
Retour où les encadrements intermédiaires sont fournis jusqu'à la précision demandée.

Probabilités -
Loi binomiale

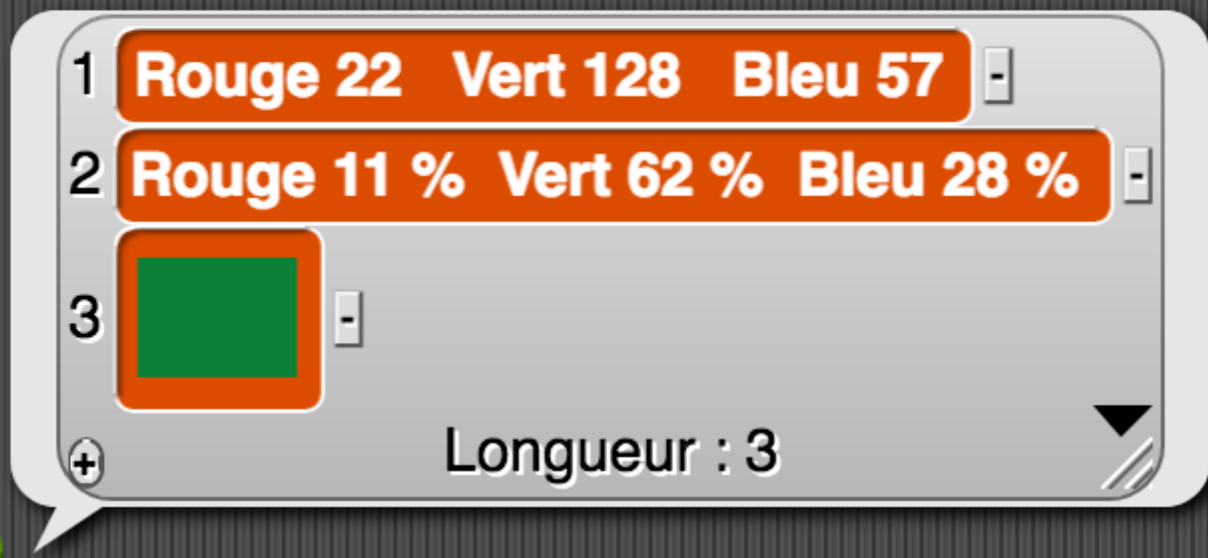


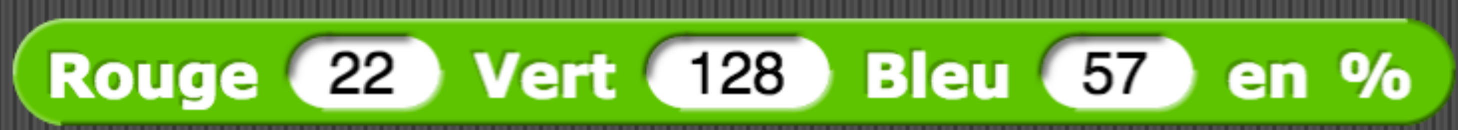


Exercice sur le codage de la couleur



Cet exercice peut être réalisé en AP et engendrer un travail sur les pourcentages.





Retour sur le programme officiel de mathématiques de la classe de première S.

maths_PremiereS.sept2010.AlgorithmeSurligne.pdf

Algorithmique

En seconde, les élèves ont conçu et mis en œuvre quelques algorithmes. Cette formation se poursuit tout au long du cycle terminal.

Dans le cadre de cette activité algorithmique, les élèves sont entraînés à :

- décrire certains algorithmes en langage naturel ou dans un langage symbolique ;
- en réaliser quelques-uns à l'aide d'un tableur ou d'un programme sur calculatrice ou avec un logiciel adapté ;
- interpréter des algorithmes plus complexes.

Aucun langage, aucun logiciel n'est imposé.

L'algorithmique a une place naturelle dans tous les champs des mathématiques et les problèmes posés doivent être en relation avec les autres parties du programme (analyse, géométrie, statistiques et probabilités, logique), mais aussi avec les autres disciplines ou le traitement de problèmes concrets.

À l'occasion de l'écriture d'algorithmes et programmes, il convient de donner aux élèves de bonnes habitudes de rigueur et de les entraîner aux pratiques systématiques de vérification et de contrôle.

Instructions élémentaires (affectation, calcul, entrée, sortie).

Les élèves, dans le cadre d'une résolution de problèmes, doivent être capables :

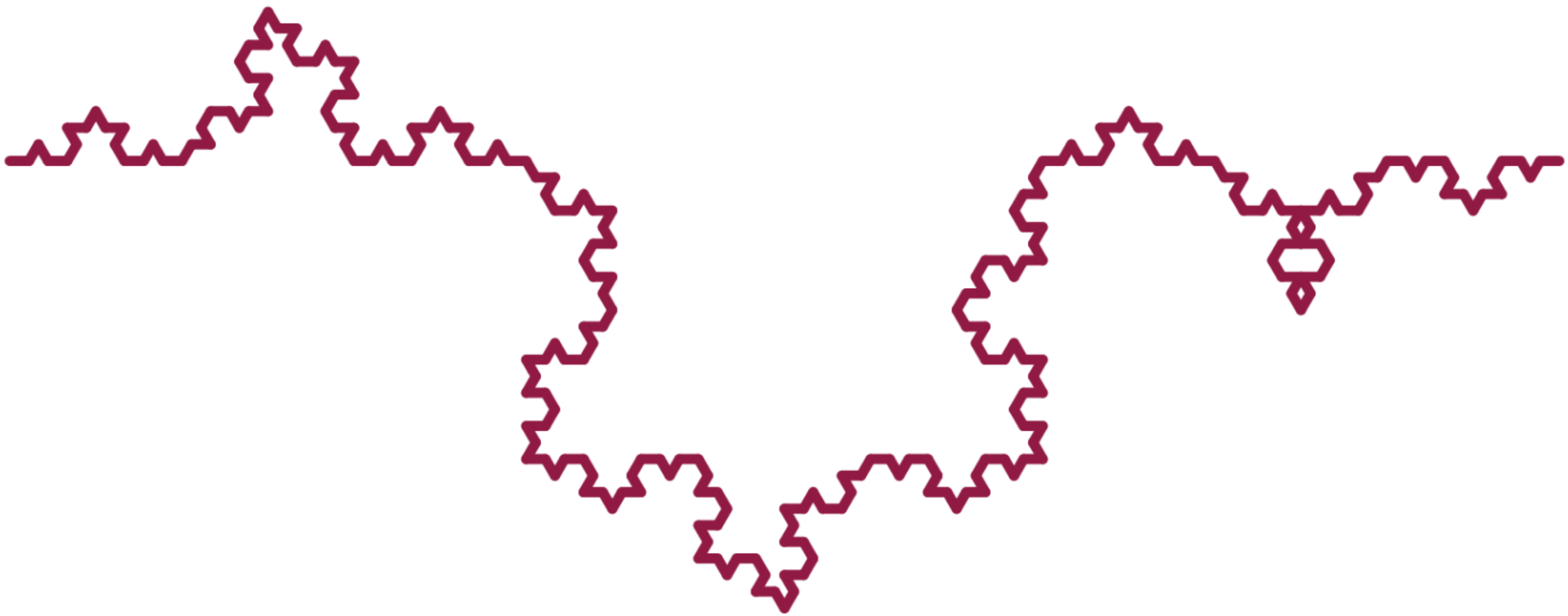
- d'écrire une formule permettant un calcul ;
- d'écrire un programme calculant et donnant la valeur d'une fonction ;
- ainsi que les instructions d'entrées et sorties nécessaires au traitement.

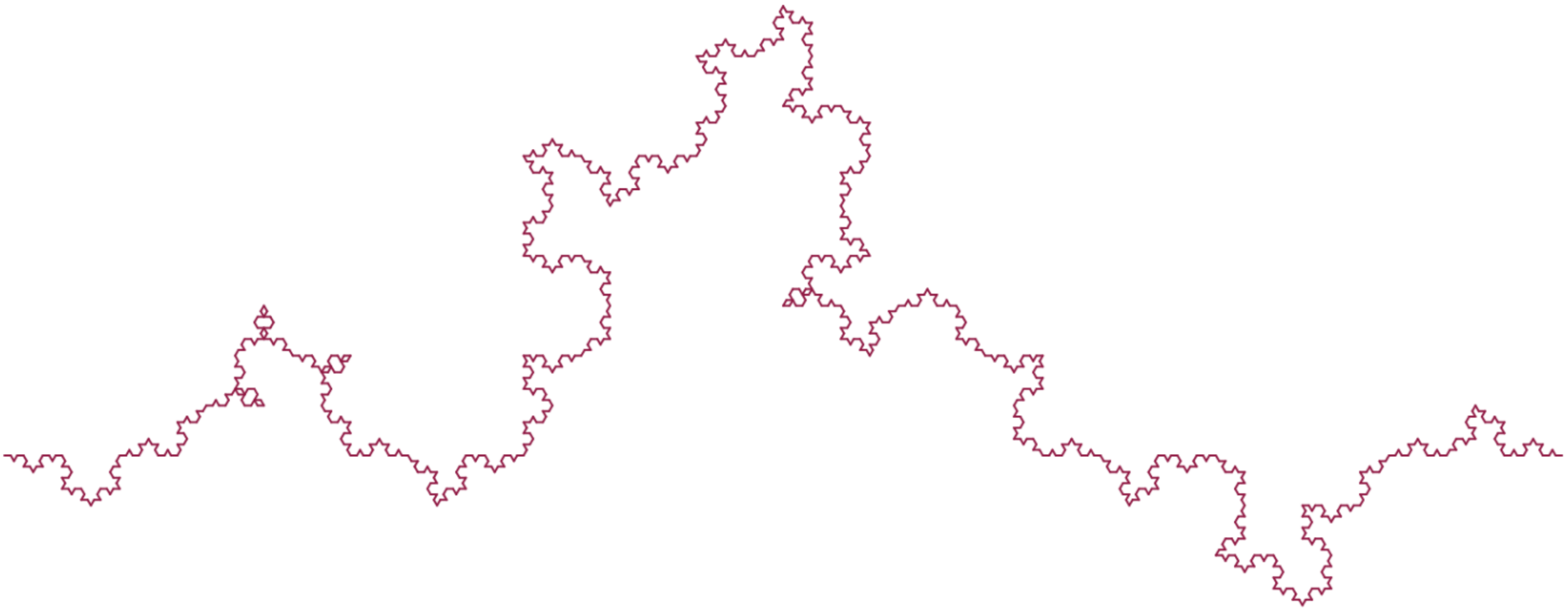
Boucle et itérateur, instruction conditionnelle

Les élèves, dans le cadre d'une résolution de problèmes, doivent être capables de :

- programmer un calcul itératif, le nombre d'itérations étant donné ;
- programmer une instruction conditionnelle, un calcul itératif, avec une fin de boucle conditionnelle.

VonKochAleatoire côté = 760 px ; niveau 4





Logic will get you from A to B. Imagination will take you everywhere.

La logique vous mènera de A à B. L'imagination vous mènera partout.

Albert Einstein

The role of the teacher is to create the conditions for invention rather than provide ready-made knowledge.

Seymour Papert

The reason most kids don't like school is not that the work is too hard, but that it is utterly boring.

Seymour Papert

My basic idea is that programming is the most powerful medium of developing the sophisticated and rigorous thinking needed for mathematics, for grammar, for physics, for statistics, for all the "hard" subjects.... In short, I believe more than ever that programming should be a key part of the intellectual development of people growing up.

Seymour Papert

https://www.azquotes.com/author/4399-Albert_Einstein

https://www.azquotes.com/author/20945-Seymour_Papert