

# Programmer avec *p5Visuel*<sup>1</sup> ... ... pour résoudre des équations

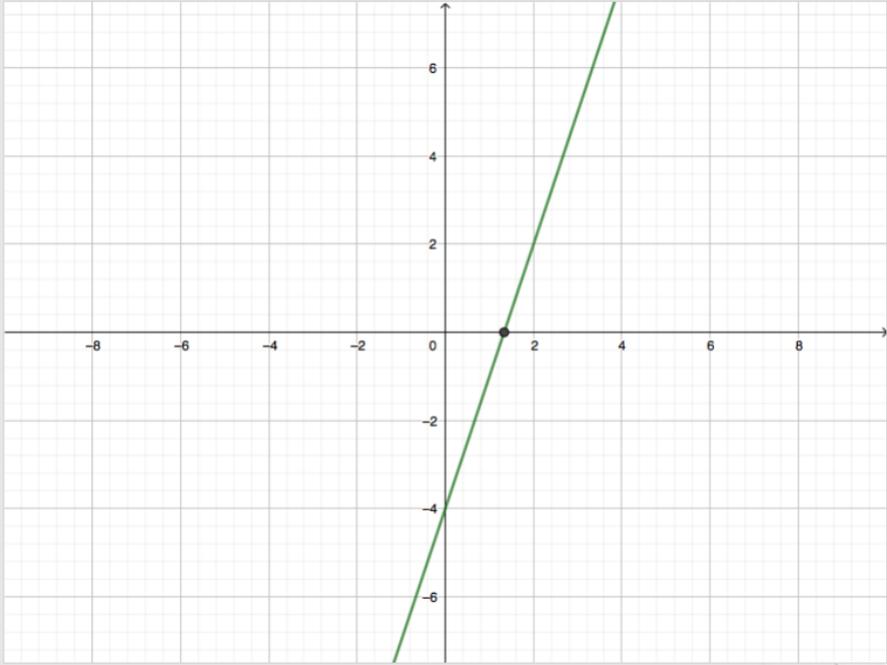
Quand on réfléchit à l'utilisation de la programmation dans l'enseignement des mathématiques, on pense souvent à des applications traditionnelles, directement liées au curriculum, comme la résolution d'équations. On va voir que *p5Visuel* nous permet de réaliser facilement de tels programmes, comme celui illustré ci-dessous, comportant des notations mathématiques professionnelles et des graphes avancés.

**Résolution d'une équation linéaire  $mx + b = 0$**

Valeur de  $m$  :     Valeur de  $b$  :    

Une solution =  $\frac{-b}{m} = \frac{-(-4)}{3} = 1.3333333333333333$

Graphe de  $y = mx + b$  et son point d'intersection avec l'axe des  $x$



Pour exécuter le programme, aller à <http://profmath.uqam.ca/~boileau/Nouvelles/Fichiers/Envol2021d/degre1.html>

Pour donner une idée des possibilités offertes par *p5Visuel*, nous allons aussi insister plus que nécessaire sur quelques options de présentation utiles pour obtenir un affichage soigné.

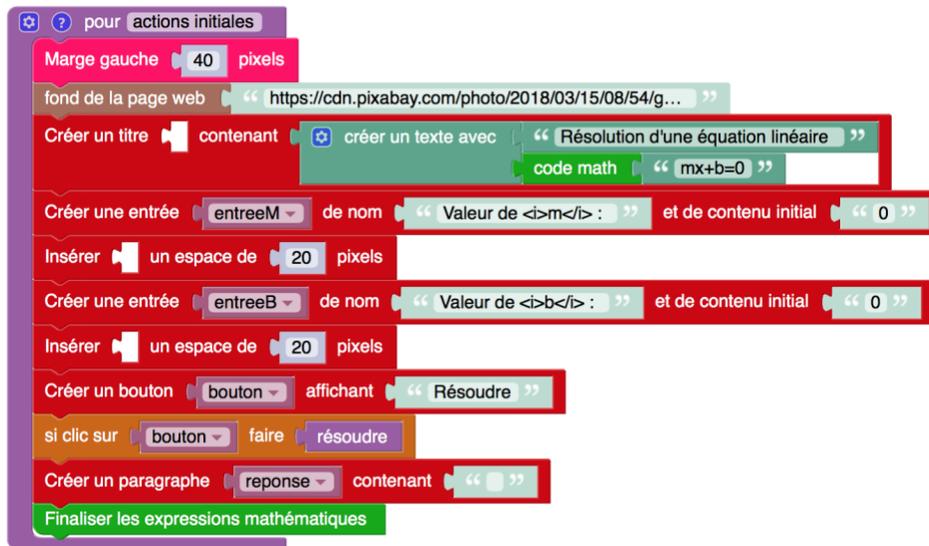
---

<sup>1</sup> *p5Visuel* est un logiciel libre et gratuit disponible à l'adresse suivante

<http://profmath.uqam.ca/~boileau/p5VisuelWEB/index.html>

Dans la version web de l'article, tous les liens web de l'article seront actifs. Faire contrôle+clic (sur Windows ou Linux) ou commande+clic (sur Macintosh) pour ouvrir le lien dans une nouvelle fenêtre.

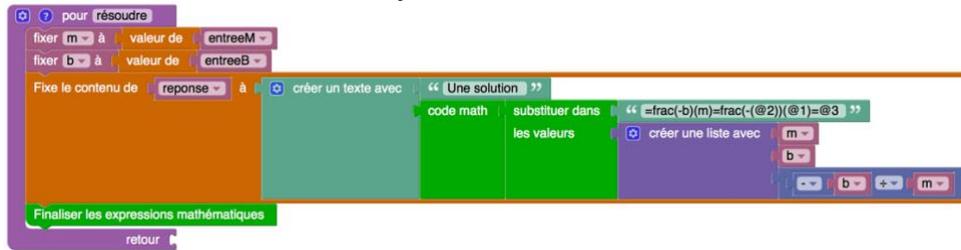
Au départ, nous allons placer dans la fonction *actions initiales* les éléments de base de l'interface.



Décrivons brièvement les blocs qui composent la fonction ci-dessus :

- La première instruction nous assure que tous les éléments de la page web seront à 40 pixels du bord gauche de la fenêtre.
- La seconde instruction spécifie qu'une image (tirée de Pixabay, un site d'images libres de droits) occupera le fond de la page.
- La troisième instruction met bout à bout deux textes, dont le second sera rendu comme un code mathématique, et les place dans un titre. Ce titre n'a pas besoin d'être nommé car on n'y appliquera aucun changement (de taille, de couleur, etc.).
- L'instruction #4 crée une entrée (nommée « entreeM »), qui servira à obtenir la valeur du paramètre  $m$ . Cette entrée est nommée car on va vouloir récupérer la valeur qui y sera placée par l'utilisateur.
- Par défaut, les entrées et les boutons sont placées consécutivement sur une même ligne. C'est pourquoi on voudra ajouter des espaces entre ces éléments. C'est précisément le rôle des instructions #5 et #7.
- L'instruction #6 crée une entrée (nommée « entreeB »), qui servira à obtenir la valeur du paramètre  $b$ .
- L'instruction #8 crée un bouton nommé « bouton » qui affiche « Résoudre ».
- L'instruction #9 spécifie que, en cas de clic sur « bouton » on devra exécuter la fonction « résoudre ».
- L'instruction #10 crée un paragraphe (nommé « reponse ») qui ne contient rien pour le moment. Ce paragraphe contiendra la réponse obtenue lors d'un clic sur le bouton « Résoudre ».
- L'instruction #11 déclenche le rendu de tous les codes mathématiques précédents. En général, comme ce rendu prend un temps visible par l'utilisateur, il est préférable de le faire une seule fois, pour tous les codes mathématiques. (Mais ici, comme il n'y a qu'un seul code mathématique, cette précaution est superflue.)

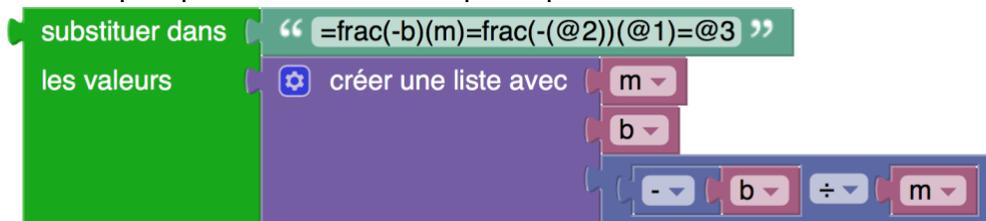
Quand le bouton « Résoudre » est cliqué, la fonction suivante est exécutée :



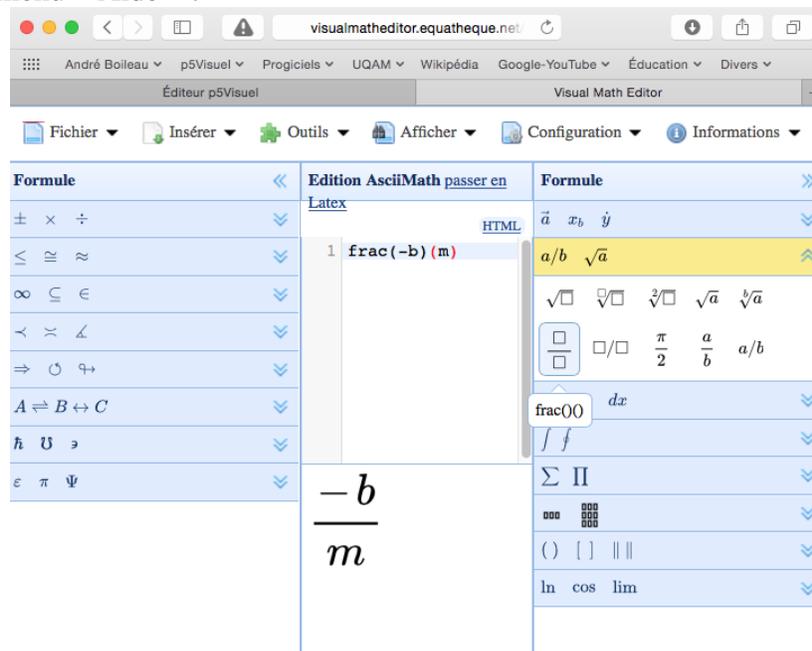
Examinons brièvement ce qu'elle fait :

- Les deux premières instructions vont chercher les valeurs de  $m$  et  $b$  dans les entrées correspondantes.  
Notez que la valeur d'une entrée est une de ses propriétés, mais qu'elle en a plusieurs autres : taille, style et couleur des caractères, couleur du fond, etc.
- La troisième instruction va placer dans le paragraphe « reponse » le texte indiqué (qui comporte une partie « code math »).
- La dernière instruction déclenche le rendu du code mathématiques précédent.

Examinons de plus près un extrait de la réponse précédente :



L'expression «  $\text{frac}(-b)(m)$  » est un code mathématique pour  $\frac{-b}{m}$ . On peut facilement obtenir ces codes dans l'éditeur p5Visuel en choisissant l'item « Accès au Visual Math Editor » du menu « Aide » :



Le mécanisme du bloc « substituer dans ... les valeurs ... » a été conçu pour résoudre le problème suivant : comment, dans une expression, remplacer certaines occurrences d'une lettre (mais pas d'autres) par une valeur numérique ? Il suffit de coder les lettres à remplacer par une expression « @ » suivie d'un nombre qui indiquera la position des nombres dans une liste de valeurs numériques. Ainsi, dans le cas ci-dessus, « @1 » sera remplacé par la valeur de  $m$ , « @2 » par la valeur de  $b$ , et « @3 » par la valeur de  $(-b)/m$ .

Tel qu'il est, le programme répond de façon adéquate, sauf quand les valeurs de  $m$  et de  $b$  sont problématiques : par exemple, si les valeurs entrées par l'utilisateur ne sont pas des nombres, on obtiendra « NaN » (« Not a Number » c'est-à-dire « Pas un Nombre »).

### Résolution d'une équation linéaire $mx + b = 0$

Valeur de  $m$  :     Valeur de  $b$  :    

Une solution =  $\frac{-b}{m} = \frac{-(\textit{bonsoir})}{\textit{bonjour}} = \text{NaN}$

Nous pouvons rendre le tout plus convivial en ajoutant (à la fonction *réponse*) diverses vérifications couplées aux réponses appropriées :

```

si pas est un nombre ? m
faire
  Fixe le contenu de reponse à " Le paramètre <i>m</i> doit être un nombre "
  quitter la fonction (en retournant )
si pas est un nombre ? b
faire
  Fixe le contenu de reponse à " Le paramètre <i>b</i> doit être un nombre "
  quitter la fonction (en retournant )
si m = 0 et b = 0
faire
  Fixe le contenu de reponse à " Tout nombre <i>x</i> est solution de l'équation "
  quitter la fonction (en retournant )
si m = 0
faire
  Fixe le contenu de reponse à " Il n'y a pas de solution "
  quitter la fonction (en retournant )
  
```

Notez au passage les expressions « <b>texte</b> » (qui indique que **texte** doit être en gras) et « <i>texte</i> » (qui indique que *texte* doit être en italique). On aurait pu obtenir des résultats semblables en utilisant des blocs tels

```

texte en gras " abc "
  
```

On peut aussi ajouter une touche de visualisation en affichant le graphe de la fonction

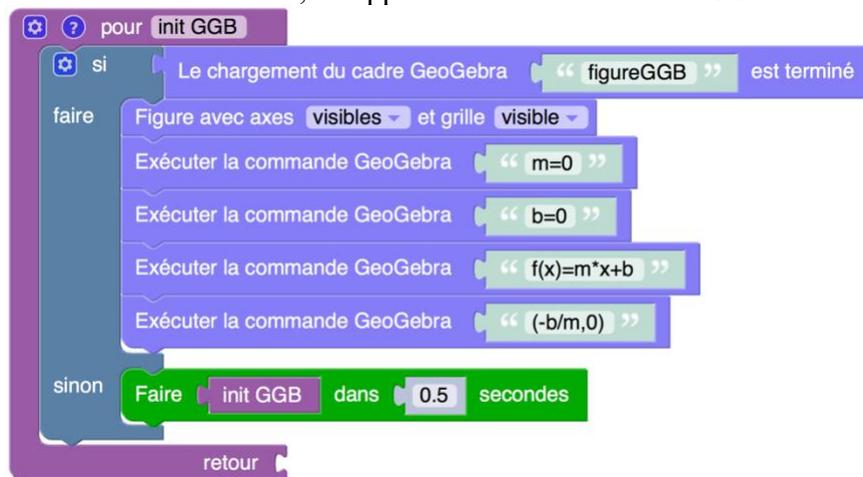
$$y = m x + b$$

et son point d'intersection avec l'axe des abscisses (s'il existe). Mais plutôt que de recourir à une esquisse boiteuse, on fera appel à une représentation sophistiquée utilisant *GeoGebra*.

La première étape est d'ajouter les instructions suivantes à la fin de la fonction *actions initiales* :



On demande d'abord d'installer une figure *GeoGebra* dans notre page web avec les caractéristiques spécifiées (figure 2D de dimensions 800x600, sans interface). Puis on demande d'en faire l'initialisation, en appelant notre fonction *init GGB* :



Au départ, c'est *GeoGebra* qui va se charger d'installer une figure vierge, et ça peut lui prendre un certain temps. Mais pendant ce temps là, *p5Visuel* continue de fonctionner : il pourrait donc arriver que *p5Visuel* tente de modifier la figure avant que celle-ci soit prête. Pour éviter que ça se produise, *p5Visuel* doit vérifier que la figure est complètement chargée avant de commencer à la modifier : si elle est chargée, on la modifie, sinon on réessaie dans une demi-seconde.

Et quand la figure est chargée, on doit décrire ce qu'on veut lui ajouter. Pour ce faire, après avoir spécifié que les axes et la grille doivent être visibles, on va utiliser des commandes qu'on aurait pu taper dans le *champ de saisie* (en utilisant le bloc *Exécuter la commande GeoGebra*) pour définir les variables *m* et *b*, la fonction  $f(x)=m*x+b$ , et le point d'intersection  $(-b/m,0)$ .

Par la suite, à chaque fois que la fonction *résoudre* est appelée, il faudra appeler la fonction *mise à jour GGB* suivante, qui communiquera à *GeoGebra* les nouvelles valeurs de *m* et de *b*.



Au bout de toute cette démarche, nous obtenons le programme suivant :

The image shows a Scratch script for a GeoGebra linear equation solver. It is organized into several loops:

- pour actions initiales:** Sets the page URL, creates a title, input fields for coefficients  $m$  and  $b$ , a 'Résoudre' button, and a paragraph for the final result.
- pour init GeoGebra:** Loads the GeoGebra environment, sets axes, and defines the functions  $y = mx + b$  and  $y = -b/m$ .
- pour mise à jour GGB:** Updates the variables  $m$  and  $b$  in GeoGebra.
- pour résoudre:** Checks if  $m$  is zero. If not, it calculates the intersection point and displays the solution. If  $m$  is zero, it checks if  $b$  is zero (infinite solutions) or not (no solution).

Pour accéder au code source, aller à <http://profmth.uqam.ca/~boileau/Nouvelles/Fichiers/Envol2021d/degre1>

Notez que le graphique *GeoGebra* conserve toute sa polyvalence habituelle : déplacer le système d'axes en appuyant sur le bouton de la souris, on peut faire des zooms en utilisant la roulette de la souris, on peut obtenir des informations sur les diverses composantes de la figure par des clics droits, etc.

Et pourquoi nous arrêter en si bon chemin : on peut suivre une démarche semblable pour les équations du second degré.

The image shows a web-based interface for solving a quadratic equation. The background features a person in a red coat standing on a beach looking at the ocean.

**Trouver les solutions de l'équation**  
 $ax^2 + bx + c = 0$  (où  $a \neq 0$ )

La formule utilisée

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$a = 1$   
 $b = 3$   
 $c = 2$

Trouver les solutions

discriminant  $= b^2 - 4ac = (3)^2 - 4(1)(2) \approx 1$

Deux solutions

Solution 1  $= \frac{-b + \sqrt{b^2 - 4ac}}{2a} = \frac{-(3) + \sqrt{(3)^2 - 4(1)(2)}}{2(1)} \approx -1$

Solution 2  $= \frac{-b - \sqrt{b^2 - 4ac}}{2a} = \frac{-(3) - \sqrt{(3)^2 - 4(1)(2)}}{2(1)} \approx -2$

A coordinate plane graph shows a parabola opening upwards with its vertex at  $(-1.5, -1.25)$  and x-intercepts at  $x = -1$  and  $x = -2$ .

Pour accéder au code source, aller à <http://profmth.uqam.ca/~boileau/Nouvelles/Fichiers/Envol2021d/degre2>