



Algorithme et programmation en Mathématiques

Document rédigé dans le cadre des TraAM

Travaux Académiques Mutualisés

2015-2016

par les membres du groupe :

- Loic Asius professeur de collège
- Angèle Britel professeure de collège
- Robert Corne professeur de collège
- Jérôme Dieudonné professeur de collège
- Aurelie Huillery-Perrin professeure de lycée
- Emilie Jadoul professeure de collège
- Nicolas Lemoine professeur de collège
- Mohamed Mesmoudi professeur de collège
- Cyril Michau professeur de collège
- Olivier Vogt professeur de collège

sous la direction de Richard Breheret IA-IPR de mathématiques de l'académie de CRÉTEIL

Introduction

Ce document a pour objectif de rendre compte de la réflexion conduite au cours de l'année scolaire 2015-2016 dans le cadre des travaux mutualisés sur l'algorithmique et la programmation au cycle 4 en mathématiques.

Les programmes ont été conçus en utilisant le logiciel Scratch. Les productions ont évolué en cours d'année en laissant davantage de place à des programmes dont la problématique n'était pas a priori mathématique, en insistant sur une approche plus ludique que permet le logiciel.

Les activités proposées par le groupe ne sont pas toutes publiées ; elles ont fait l'objet d'une sélection cohérente avec l'évolution de notre réflexion. Notre réflexion continue, en particulier, dans l'exploitation du thème E en relation avec les EPI (robotique, poèmes, cryptographie et cryptanalyse ...). Les ressources en cours d'élaboration seront publiées au fur et à mesure sur le site académique.

Nous avons cherché à illustrer quelques exemples proposés par les programmes et leur mise en œuvre pédagogique en classe. Certains ont été considérés comme difficiles pour une première approche à la fois pour les enseignants et pour les élèves.

Les documents produits ont une charte graphique commune détaillant en particulier, les compétences mathématiques développées, les notions algorithmiques mises en jeu ainsi que les domaines du socle investis.

Les 8 fiches analysées dans la suite de cette synthèse essaient de balayer les attendus du programme.

Différentes entrées pour travailler le thème E du programme

Plusieurs entrées sont possibles afin d'introduire la notion d'algorithmique et de programmation à partir du cycle 4. Nous nous sommes intéressés à trois d'entre elles :

- l'algorithmique débranchée ;
- la géométrie ;
- les jeux.

Algorithmique débranchée

Il nous a paru nécessaire et évident de mener notre première réflexion autour de l'algorithmique débranchée qui est une introduction au mode de pensée algorithmique et un réinvestissement possible tout au long du cycle. Il est tout à fait possible de faire sentir un algorithme sans formalisme mais par manipulation d'objets.

Un premier exemple

Un exemple d'algorithmique débranchée contenant de la manipulation d'objets que nous souhaitons mettre en avant est celui du tri à bulles.

Le tri à bulles consiste à comparer deux valeurs côte à côte d'une suite de nombres, et de changer leurs places s'ils sont mal placés.

Par exemple : pour la suite : 9, 4, 7, 3

1	9	4	7	3
2	4	9	7	3
3	4	7	9	3
4	4	7	3	9
5	4	3	7	9
6	3	4	7	9

L'activité permet d'aborder cet algorithme de tri par la manipulation de cartes avec une approche progressive afin d'essayer de faire comprendre comment l'ordinateur « voit » les nombres.

Cette activité peut être tout à fait prolongée en danse.

Un second exemple

L'activité du magicien (deviner un nombre) permet de travailler, sur papier, la notion de boucle ainsi que de test conditionnel. Un travail sur les puissances de deux est aussi engagé.

Le but de cette activité est d'aiguiser la curiosité de l'élève et l'amener à comprendre le mystère qui se cache derrière ce jeu de magicien, ainsi que les étapes nécessaires pour le maîtriser et le montrer à son tour aux autres. L'élève entre naturellement dans une logique algorithmique en respectant l'ordre des étapes. De plus, le professeur peut exploiter les règles du jeu pour travailler les puissances d'un nombre et le système binaire.

Un troisième exemple

Il nous est apparu également nécessaire de proposer des activités débranchées orientées mathématiques que l'on puisse mettre en oeuvre en cours de cycle 3 ou dès le début du cycle 4.

L'activité de Kaprekar entre pleinement dans ce cadre : c'est une activité mathématique qui nécessite uniquement la connaissance des nombres entiers et de la soustraction de ceux-ci.

Elle est donc envisageable en début d'apprentissage pour introduire la notion d'algorithme puisque qu'on retrouve une série de questions avec parfois des répétitions mettant déjà en oeuvre les bases de la construction algorithmique.

L'activité consiste à faire conjecturer aux élèves le résultat associé à l'algorithme de Kaprekar : peu importe le nombre de trois chiffres choisi, lorsque l'on soustrait au plus grand nombre composé de ces trois chiffres, le plus petit nombre composé de ces trois chiffres, on obtient en un minimum d'étapes (cinq) toujours le même résultat de 495.

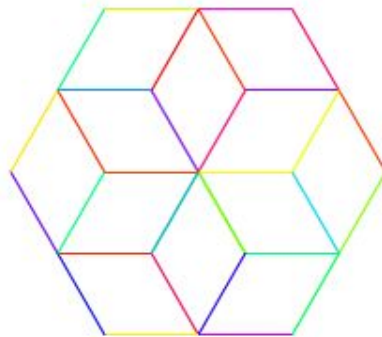
L'activité est même évolutive puisqu'on peut demander aux élèves de tester avec quatre chiffres plutôt que trois et pour les élèves les plus en avance, on peut aussi leur demander de conjecturer ce qu'il se passe pour un nombre à deux chiffres.



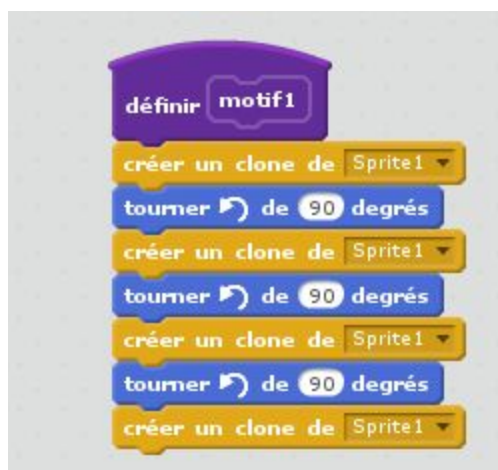
C'est une activité qui lie donc mathématiques, jeu, méthode algorithmique et culture scientifique. Elle est idéale pour bien débiter en algorithmique et programmation.

La géométrie

Les constructions géométriques, notamment les frises, les pavages et les rosaces permettent de travailler une grande partie du thème E. La construction de frises permet de travailler les boucles mais aussi le repérage dans le plan ainsi que la notion de variables. Du point de vue de la mise en place de ces travaux au sein de la classe, il est possible, dans un premier temps, de donner le motif de départ puis de programmer la construction. Dans un second temps, de montrer la construction complète et de laisser trouver aux élèves un motif initial permettant de construire la figure demandée.



L'utilisation du bloc "Ajouter blocs" permet de définir le motif initial et de l'utiliser lors de la construction. Cette utilisation peut aider à construire la notion de fonction dans le cadre de la programmation.



Les jeux

Les jeux ont tenu une place importante dans notre réflexion. Les activités du labyrinthe et celle du serpent s'inscrivent dans cette démarche.

Un premier exemple

Le jeu du labyrinthe est une activité qui permet de manipuler très rapidement plusieurs éléments dans le logiciel Scratch :

- un arrière plan qui sert de décor ou le labyrinthe lui-même importé sous forme d'image ;
- l'interaction entre deux objets (ici entre deux lutins) ;
- les déplacements d'un lutin.

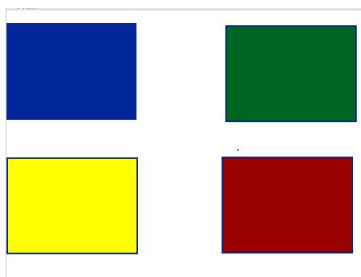
L'activité peut être agrémentée de plusieurs autres défis comme ajouter des passages secrets, des objets se déplaçant en permanence (fantômes), des objets à saisir par le lutin au cours de son labyrinthe, une variable qui compte le nombre de points, un chronomètre...

Un second jeu

Le jeu du serpent est bien connu des utilisateurs des premiers téléphones portables. Il permet de mettre en place facilement la création et la manipulation des clones dans Scratch. Il est possible de démarrer en laissant libre court à l'imagination des élèves pour dessiner la tête du serpent ou bien de proposer un fichier Scratch contenant juste la tête dessinée.

Comme pour le labyrinthe, il est tout à fait possible d'ajouter des objets dans le jeu ou des contraintes supplémentaires (comme la vitesse par exemple). Cela dépendra du niveau d'avancement dans la programmation des élèves, c'est un moyen de différencier autour d'une activité commune à toute la classe.

Un projet sur quelques séances : le jeu des couleurs



Ce jeu peut être abordé dans le cadre de la démarche de projet. La règle du jeu est très connue : l'ordinateur propose une suite de couleurs que le joueur doit reproduire. Il commence par une seule couleur, puis ajoute à chaque réussite du joueur un élément à suite et ce jusqu'à ce que le joueur échoue.

Les élèves découvrent le jeu grâce à l'enseignant et réfléchissent à le décomposer en sous problèmes :

- Comment diminuer la luminosité des rectangles et ajouter du son lorsqu'on clique sur chaque rectangle ?
- Comment faire jouer à l'ordinateur une séquence de couleurs ?
- Comment récupérer la suite des clics sur les rectangles du joueur ?

- Comment comparer la séquence de l'ordinateur à celle du joueur ?

Il peut être intéressant de comparer les projets sans en attendre une expertise et de partager avec les élèves les différentes propositions.

Conclusion

Des élèves, trop nombreux encore, ne proposent aucune ébauche de solution aux exercices ou aux problèmes que nous leurs soumettons. Non pas qu'ils n'ont rien à en dire, mais après discussion avec eux, et malgré notre insistance à dédramatiser "la faute", nous constatons qu'ils sont, malgré tout, souvent bloqués par la peur de se tromper ou d'avoir un "mauvais" retour : ils ont parfois juste perdu le goût de "s'y mettre".

Approcher l'algorithmique et la programmation avec Scratch, en classe, en proposant des activités éloignées des mathématiques pourrait, a priori, amener ces élèves à changer leur regard sur la matière.

Ils peuvent être en mesure de produire une réponse, correspondant ou non à la demande, mais qui aura une existence intrinsèque sur laquelle l'élève et l'enseignant pourront construire.

La "non réponse" étant une situation que nous combattons tous.

Notre réflexion a aussi abouti à une autre remarque qui pourrait être abordée :

Le logiciel Scratch possède des fonctionnalités qui paraissent inutiles car elles ne semblent pas apporter de valeur ajoutée à la résolution de problèmes mathématiques. Ces fonctions qui peuvent apparaître comme gadgets aux yeux de l'enseignant, sont ces mêmes fonctions qui attirent l'élève en difficulté. Jamais Scratch n'affiche « syntaxe erreur » : il y a un résultat exploitable même si ce n'est pas le résultat attendu.

L'élève écrit un programme qui donne un résultat qu'il voit à l'écran.

L'utilisation d'un logiciel pour programmer par blocs permet un meilleur rapport à l'erreur de l'élève. En effet, ce dernier voit immédiatement si son programme n'effectue pas ce qu'il souhaitait faire, ainsi il n'y a pas besoin de l'intervention de l'enseignant pour lui signifier son erreur. L'élève peut alors essayer par lui-même de trouver une solution. Son rapport à l'erreur s'en trouve radicalement changé, elle n'est plus source d'échec mais au contraire sert de base pour trouver une solution nouvelle. Nos pratiques de ces activités ont montré que les élèves acceptaient bien plus facilement de se tromper. De plus, le fait que le "programme ne fonctionne pas" n'est pas source d'arrêt du travail ; il permet au contraire à l'élève de chercher à trouver la cause de ce mauvais fonctionnement.

Mais il est aussi important de ne pas résumer l'enseignement de l'algorithmique et de la programmation uniquement à Scratch, et donc à la partie programmation. Lors des activités que nous avons pu mener en classe dans le cadre de ces travaux mutualisés, nous avons tous pu constater que les séances débranchées avaient toute leur place dans l'apprentissage des élèves. Ces dernières permettent à l'élève de structurer sa pensée en mettant par écrit des algorithmes (sous forme de langage naturel) ou bien en faisant tourner d'autres (l'élève prend

alors le rôle de l'ordinateur, ce qui lui est très utile pour mieux comprendre le fonctionnement d'un logiciel de programmation). Ces séances sont complémentaires de celles sur ordinateur, il nous semble important de proposer ces deux approches à toutes les années du cycle 4.