



INTRODUCTION

Les données constituent la matière première, les nombres avec lesquels l'ordinateur travaille. Le traitement des données brutes, comme par exemple le nombre de chômeurs ou la quantité de pluie, permet de produire du savoir ou d'extraire de l'information.

Ces données sont traitées à l'aide d'une série d'instructions qui permettent à l'ordinateur d'exécuter ce que nous commandons, par exemple trier, rechercher ou envoyer des informations. Une série d'instructions est un **algorithme**, un concept central en informatique. Il nous permet de résoudre des problèmes, par exemple trouver des millions de décimales de Pi ou l'intégralité des pages web contenant votre nom.

Apprendre à écrire une liste d'instructions, c'est-à-dire programmer, est essentiel pour comprendre les notions fondamentales en informatique et utiliser le numérique de façon éclairée. En effet, les algorithmes sont omniprésents dans notre quotidien : calcul du meilleur itinéraire routier, atterrissage d'un avion en plein brouillard, définition des prix du blé ou du riz à la bourse, moteur de recherche internet... **Il est essentiel que les élèves ne perçoivent pas tous ces systèmes numériques comme magiques.** Sans cela, elle ou il ne comprendra pas la différence entre l'intelligence mécanique et l'intelligence humaine. Qu'est-ce qu'un algorithme ? A quoi ça sert ? C'est l'objectif des deux activités proposées ci-dessous : le jeu de l'enfant-robot et le jeu de Nim.

UN MÉTIER DU NUMÉRIQUE

Développeur / développeuse

L'enseignant·e introduit l'activité en annonçant que la classe va travailler sur les sciences informatiques. Il ou elle invite les élèves à découvrir la "vidéo métier" du développeur ou de la développeuse (sur le site : www.clesdunumerique.com) et leur demande de réfléchir à la question posée : « À votre avis, comment peut-on donner des ordres à un ordinateur, c'est-à-dire le mettre au travail, ou lui faire réaliser une tâche ? »

Outre la découverte d'un métier du numérique pouvant susciter la curiosité et l'intérêt des élèves, cette introduction leur permet de formuler collectivement des débuts de réponses, qui seront affinées au cours de l'activité.



Dans son travail, un développeur ou une développeuse écrit une liste d'instructions, un programme, afin de donner des ordres à un ordinateur et résoudre un problème. Il ou elle conçoit donc une solution à un problème, la traduit en langage informatique et répond ainsi concrètement aux besoins des utilisateurs. Il existe de nombreux langages de programmation différents. Le/la développeur·euse choisit celui qu'il ou elle utilisera pour écrire ses instructions. C'est un·e spécialiste des techniques de programmation. Les domaines d'application sont très variés : cinéma, imagerie médicale, télécom et réseaux, applications web, informatique, jeux vidéo...



COMPÉTENCES – LIENS AVEC LE PROGRAMME

Comprendre, s'exprimer en utilisant les langages mathématiques, scientifiques et informatiques

L'élève sait que des langages informatiques sont utilisés pour programmer des outils numériques et réaliser des traitements automatiques de données. Il ou elle connaît les principes de base de l'algorithmique et de la conception des programmes informatiques. Il ou elle les met en œuvre pour créer des applications simples.

Mathématiques - Espace et géométrie

- Activités de repérage ou de déplacement, ou activités géométriques pour initier à la programmation,
- Donner et suivre des instructions.

Mathématiques – Nombres et calculs

- Table de multiplication par 4, division par 4,
- Consolider les connaissances et capacités en calcul mental sur les nombres entiers,
- Résoudre des problèmes engageant une démarche à une ou plusieurs étapes et favorisant des démarches de tâtonnements et d'essais-erreurs.

MATÉRIEL

- Craies pour l'activité enfant-robot,
- Boîte de 500 jetons pour l'activité jeu de Nim.

DÉROULEMENT DE L'ACTIVITÉ

Introduction

Un algorithme est une méthode qui permet d'envisager toutes les manières de faire quelque chose. C'est une façon de décrire dans ses moindres détails comment procéder pour faire quelque chose, comme une recette de cuisine où chaque opération est décrite pas à pas dans ses moindres détails. Par exemple, regarder à gauche, puis à droite et à gauche est un algorithme pour traverser la rue. Concevoir un algorithme, c'est concevoir une méthode efficace pour réaliser une tâche ou une action sans plus avoir besoin de réfléchir. Ainsi on peut la faire exécuter par un ordinateur, une machine fabuleusement rapide et exacte mais complètement stupide.

Dès le VIII^{ème} siècle, Al Khuwarizmi définit la notion d'algorithme, le mot est une déformation de son nom. Il avait été chargé par le calife de trouver une méthode infaillible pour résoudre des problèmes comme calculer les impôts ou mesurer les réserves d'eau par exemple. Et c'est lui qui a compris comment envisager toutes les possibilités, tout le temps. Plus tard, on a repris cette méthode et on l'a appliquée aux ordinateurs.



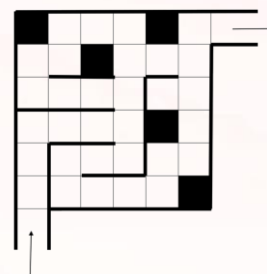
Activité 1

Enfant-robot

Un des aspects les plus frustrants concernant la programmation est que les ordinateurs obéissent toujours à la lettre aux instructions, même si les résultats sont aberrants. Cette activité offre aux élèves une expérience de cet aspect de la programmation. Elle consiste à jouer au jeu de l'enfant-robot qui doit sortir d'un petit labyrinthe que l'on aura construit dans la classe ou dessiné à la craie sur le sol de la cour.

Préparation

Tout d'abord, dessiner un quadrillage ou une sorte de labyrinthe sur le sol, avec éventuellement des zones colorées que l'on pourra interpréter comme une rivière, une forêt interdite... Ce labyrinthe doit être simple pour en sortir en quelques instructions, le niveau est à adapter au niveau de compréhension des élèves. Voici un exemple ci-contre.



Activité

1. Les élèves s'assoient autour du labyrinthe sur le sol. On choisit un.e volontaire qui jouera le rôle du robot et qui sera dirigé.e par les autres élèves pour sortir du labyrinthe.

On explique que notre volontaire qui joue le rôle du robot ne sait faire que trois choses. L'élève qui imitera le robot ne pourra que :




- avancer d'un pas,
- tourner à gauche d'un quart de tour,
- tourner à droite d'un quart de tour.

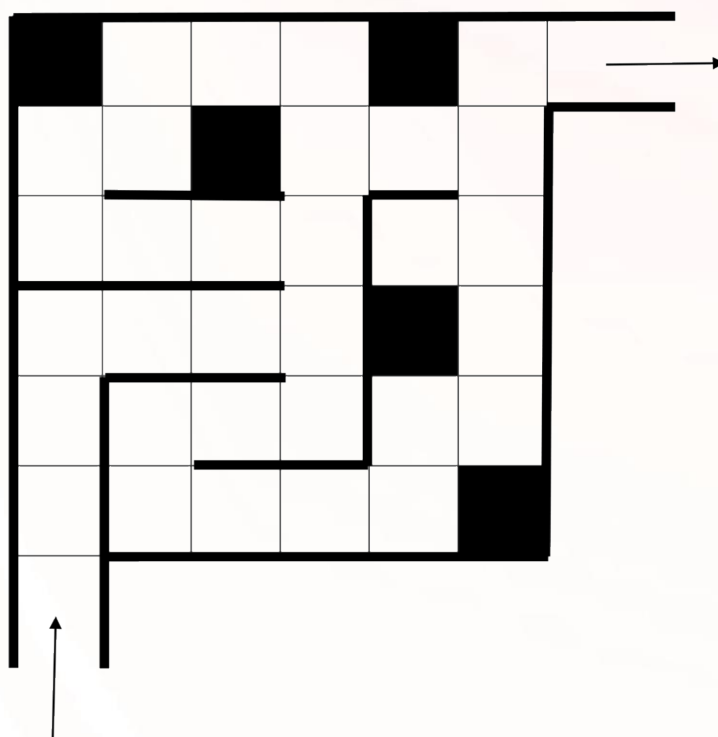
Ce n'est finalement qu'un pantin ou une marionnette qui ne peut exécuter que les trois instructions décrites ci-dessus. Une fois les consignes bien comprises, à tour de rôle, chaque élève assis autour du jeu va proposer à l'enfant-robot une des trois instructions pour lui permettre au final de sortir du labyrinthe.

On peut le refaire plusieurs fois en changeant d'élève robot et en modifiant le labyrinthe. On peut aussi bander les yeux d'un.e élève et demander à ses camarades de le/la guider dans le labyrinthe.



2. Chaque élève écrit son programme pour sortir du labyrinthe grâce aux trois instructions. Il s'agit donc d'une suite d'instructions ou de petites flèches. Exemple :

-  = « avancer d'un pas »,
-  = « tourner à gauche d'un quart de tour »,
-  = « tourner à droite d'un quart de tour ».



Programme : 

Les élèves sont des robots et testent alors leur programme. Ils doivent l'exécuter sans « réfléchir » pour vérifier si cela leur permet de sortir du labyrinthe. On peut ensuite donner le programme à un.e autre élève qui dicte les instructions à l'élève-robot.

3. Puis on peut donner un programme erroné (en informatique, on dira un programme « buggé ») à l'élève qui dicte les instructions. Il faut alors que les élèves trouvent d'où vient l'erreur pour déboguer le programme.



Prolongements

Parfois, on ne voudra pas répéter « avance d'un pas, avance d'un pas, avance d'un pas » mais dire à la place « avance de trois pas ». Donc une instruction peut avoir une valeur *variable* qui permettra d'avancer plus vite.

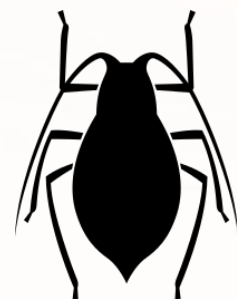
Le jeu peut aussi se compliquer s'il y a une porte (concrétisée par un objet quelconque) qui peut être fermée ou ouverte, sans qu'on le sache à l'avance. Il faudra alors introduire une *condition* dans notre algorithme : « si la porte est fermée alors fais le tour ». Mais bien entendu il faudra expliquer en détail au robot ce que veut dire faire le tour !

Tous les algorithmes du monde fonctionnent avec quatre ingrédients : une série d'instructions, des variables, des conditions et des boucles (pour recommencer plusieurs fois une série d'instructions). Donc tout ce qui se passe dans un smartphone, une tablette, un robot ou un ordinateur se réduit à ces quatre ingrédients (en quantité géante). Sauf bouleversement de la science, aucun robot ne saura faire autre chose qu'exécuter ces algorithmes.

Conclusion

Cette forme de communication est la plus proche de celle qu'utilisent les informaticien.nes lors de l'écriture de programmes. Ils ou elles donnent un ensemble d'instructions à l'ordinateur et ne constatent les résultats de ces instructions que plus tard, au moment de tester leur programme.

Il est important que les programmes soient bien écrits. Une petite erreur peut engendrer de nombreux problèmes. Ces erreurs sont appelées « bugs » (insectes en anglais) en l'honneur, si l'on peut dire, d'un papillon de nuit. En 1947, l'un des premiers gros calculateurs aux Etats-Unis tomba en panne. L'équipe de Grace Hopper, cette scientifique, informaticienne, mathématicienne et officier supérieure de la marine américaine y trouva un papillon de nuit coincé dans le circuit relais. L'insecte fut enlevé avec soin et scotché dans le journal de bord de l'ordinateur. Il y est toujours (le document est conservé au musée Smithsonian de Washington) avec la mention manuscrite « first actual case of bug being found » (le tout premier vrai bug). Il faut bien avouer qu'on parlait déjà de bug pour décrire les problèmes dans les appareils électriques bien avant la panne de ce gros ordinateur. Cependant la plaisanterie a popularisé la notion de bug informatique.





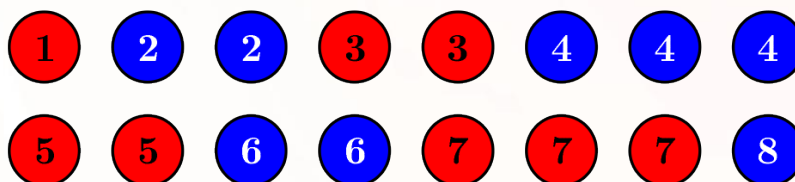
Activité 2

Le jeu de Nim

L'objectif de cette activité est de trouver l'algorithme gagnant, celui qui permet de gagner à tous les coups au jeu de Nim. Ce jeu se joue à 2 joueurs (*J1* et *J2*). Le défi pour les élèves est de trouver la stratégie gagnante à tous les coups.

Préparation et règle du jeu

On dispose sur une table 16 jetons. Chacun leur tour, les deux joueurs ramassent au choix un, deux ou trois jetons sur la table. Le joueur qui ramasse le dernier jeton remporte la partie. Exemple :



Le joueur bleu gagne

Stratégie gagnante

Le jeu de Nim est sans surprise : le premier à jouer (*J1*) perd, car il existe une astuce pour que le deuxième joueur (*J2*) gagne à tous les coups. **La stratégie gagnante est : après avoir joué, laisser un multiple de 4 jetons à l'adversaire sur la table.**

Pour se convaincre de l'efficacité de la stratégie gagnante, prenons le dernier tour comme exemple. Il reste 4 jetons, et *J1* joue :

- si *J1* prend 1 jeton, *J2* en prend 3 (dont le dernier) ;
- si *J1* prend 2 jetons, *J2* en prend 2 (dont le dernier) ;
- si *J1* prend 3 jetons, *J2* en prend 1 (le dernier).

Dans ce cas, si *J2* sait jouer, *J1* perd à tous les coups. En appliquant cette méthode, *J2* peut guider le jeu de manière à passer de 16 jetons à 12, puis 8 et enfin 4. Donc, si *J2* sait jouer, *J1* a perdu la partie avant même de commencer.

Le but de cette activité pour les élèves est d'énoncer de façon explicite la stratégie gagnante.

Activité

1. Commencer à jouer avec les élèves, sans dire qu'il y a un truc. Si vous jouez bien, vous gagnez à tous les coups. Bien sûr, pour gagner vous devez laisser votre adversaire commencer. S'il insiste pour ne pas commencer, vous pouvez toujours essayer de gagner en rattrapant la stratégie gagnante à la première erreur (laisser un multiple de 4 jetons sur la table pour l'autre joueur).



2. Si un élève connaît ou comprend la stratégie gagnante du jeu, il ou elle pourra vous remplacer pour jouer avec d'autres élèves.
3. Pour aider les élèves à découvrir la stratégie gagnante, vous pouvez grouper les jetons par 4, rendant ainsi l'astuce plus visible ou ajouter / retirer 4 jetons du jeu pour observer que la stratégie gagnante reste gagnante.

Conclusion

Comme dans le jeu de Nim, un algorithme est une stratégie gagnante qui permet de trouver la solution à un problème donné. Les élèves découvriront que beaucoup de choses que nous faisons avec notre cerveau (compter, trier des objets, rechercher un mot dans un texte...) peuvent être faites par une machine, car cela se réduit à un algorithme. Mais ils ou elles comprendront aussi que faire un joli dessin, choisir sa ou son meilleur-e ami-e, inventer une danse, appartiennent à une autre forme d'intelligence, très différente !

Sources :

© Sciences manuelles du numérique, Martin Quinson,
<http://people.irisa.fr/Martin.Quinson/Mediation/SMN/>

© Découvrir l'informatique par le jeu, Nathalie Revol et Cathy Louvier,
<https://pixees.fr/?p=4481>

© Images des Mathématiques, CNRS, 2014
<http://images.math.cnrs.fr/Dis-maman-ou-papa-c-est-quoi-un-algorithme-dans-ce-monde-numerique>

POUR ALLER PLUS LOIN

L'aventure des Sépas, web-série de 20 dessins animés en 3D où les terriens apprennent la science aux extra-terrestres.

Vidéo « Les Sépas et les algorithmes » :

<http://www.universcience.tv/video-les-sepas-et-les-algorithmes-5829.html> (3'24'')



Lightbot, un jeu qui permet de se familiariser aux premiers principes de la programmation informatique. Les élèves doivent planifier les déplacements et les actions d'un robot ou d'une robotte :

<https://lightbot.com/hour-of-code-2015.html>