

Pourquoi apprendre à coder à l'école ?

Pour apprendre...



THIERRY KARSENTI, Ph. D
Directeur, CRIFPE

Directeur du CRIFPE, détenteur de la Chaire de recherche du Canada sur les technologies en éducation, et professeur titulaire à l'Université de Montréal. Au cours de sa carrière, il a publié de nombreux livres et articles scientifiques portant sur le numérique et la formation. Membre de l'Académie des sciences du Canada (Société Royale), ses réalisations ont été reconnues par de nombreux prix nationaux et internationaux.



JULIEN BUGMANN, Ph.D.
Stagiaire postdoctoral

Docteur en Sciences de l'Éducation. Il est spécialisé dans les nouvelles technologies et les apprentissages. Actuellement stagiaire postdoctoral à l'Université de Montréal, ses projets portent sur l'apprentissage du code par les élèves à l'aide de jeux vidéo tels que Minecraft mais aussi via la programmation de robots.

par **Thierry Karsenti et Julien Bugmann**

Qui aurait cru, qu'un jour, des robots et des jeux vidéo entreraient dans les salles de classe avec l'accord des enseignants, des décideurs politiques, et même des parents d'élèves ? Qui aurait cru, qu'un jour, des enseignants, du primaire au secondaire, proposeraient une initiation à la robotique, à Minecraft, ou encore utiliseraient des LEGO pendant leurs cours de mathématiques ou de sciences ? Personne. Personne, si ce n'est quelques précurseurs en la matière tels que Seymour Papert (1981), qui proposa à ses élèves, dès les années 70, d'apprendre à programmer la célèbre tortue LOGO. Et même si ce projet s'est quelque peu perdu en chemin, l'apprentissage de la programmation revient aujourd'hui, en 2017, en force dans nos écoles. Ainsi, les programmes scolaires s'adaptent, les outils évoluent, les enseignants se forment... et les élèves apprennent. Cet article propose un bref tour d'horizon de ce qu'est, et comment fonctionne, l'apprentissage de la programmation aujourd'hui, présente quelques outils permettant d'apprendre à programmer, et revient sur différents projets de recherche qui ont été menés, ici au Québec, sur l'apprentissage de la programmation à l'école primaire et secondaire.

Coder, pour pouvoir décoder...

Le code est partout...tout autour de nous ! Et, alors que les technologies sont de plus en plus présentes, il faut s'attendre à y être encore plus exposé dans les années à venir...Aussi, la question de la formation de tous, et en particulier des plus jeunes, à sa compréhension, est d'une absolue nécessité. Il s'agit même d'une compétence majeure à acquérir pour les jeunes (OECD, 2015).

En effet, dès lors que l'on utilise un outil technologique, que cela soit un téléphone portable, un ordinateur, un réseau social, et même, de plus en plus, une voiture, on génère un code qui nous est invisible, mais qui va dicter à l'outil numérique ce qu'il doit faire, comment le faire et en réponse à quelle action de l'utilisateur. Mais il va aussi permettre l'enregistrement de certains éléments, l'adoption de certains « comportements », et in fine, peut-être aussi la future promotion de produits ou de services en lien avec la demande initiale de l'utilisateur.

Aussi, pour se prémunir de tout abus et rester, autant que possible, maître de nos actes et de notre environnement, il est indispensable pour tous de comprendre ce que fait ce code et pourquoi il le fait. C'est encore plus important pour les élèves qui vont avoir à affronter un univers dans lequel les technologies auront la part encore bien plus belle qu'aujourd'hui. Et pour cela, il est tout à fait possible de leur apprendre à programmer, et ce, dès la maternelle. Certains pays l'ont bien compris et ont intégré, à leurs curriculums, des enseignements obligatoires de la programmation, d'autant plus que certains chercheurs ont montré qu'apprendre à programmer était important (Duncan & Bell, 2015 ; Mubin, Stevens, Shahid, Mahmud, & Dong, 2013) et comportait des impacts éducatifs (Smith, Sutcliffe, & Sandvik, 2014). C'est le cas pour la Colombie-Britannique¹, la Nouvelle-Ecosse², mais aussi, à l'international, la France³, le Royaume-Uni, les Etats-Unis⁴ ou encore la Suède⁵.

Ainsi, la programmation est-elle en train de bousculer les habitudes éducatives de millions d'élèves ? Et surtout, comment est-ce possible d'apprendre à programmer à des élèves qui, de prime abord, n'y connaissent rien en programmation ?

Eh bien, simplement, en les initiant d'une manière progressive et en ayant recours à des outils adaptés. Par exemple, il est possible de profiter d'un encadrement et de soutien de la part d'organismes tels que *code.org*⁶, qui propose de multiples ressources et encourage la pratique et l'apprentissage du code à travers le monde, notamment via *l'Hour of code* (<https://hourofcode.com/fr>). Cet évènement permet, le temps d'une semaine, à des dizaines de millions d'élèves d'apprendre à programmer, et ce sur différents

supports. En effet, et nous ne l'avons pas encore précisé jusqu'ici, on peut apprendre à programmer tant sur une tablette, qu'un téléphone, qu'un ordinateur...etc.

Exemples d'outils pour apprendre/enseigner la programmation

On peut donc apprendre à programmer à l'aide de différents outils. Que ce soit une application tablette ou ordinateur (par exemple *Swift Playgrounds* ou *Scratch*), un jeu vidéo (par exemple *Minecraft*), un robot éducatif (par exemple, *Bee-Bot*, *Mbot*, *Blue-Bot* ou encore *Dash and Dot*) et même, depuis peu, en programmant un robot humanoïde (par exemple, le robot humanoïde *NAO*). Cette partie présente quelques exemples d'outils d'initiation à la programmation utilisés et utilisables en contexte éducatif :

Les applications

Scratch et Scratch Jr : Le logiciel Scratch a été développé par le *Massachusetts Institute of Technology* (MIT) et permet d'initier les enfants au code tout en s'amusant. Les élèves créent et modifient un code informatique en déplaçant des briques de couleurs et en les organisant selon leurs besoins et souhaits. Scratch est gratuit et fonctionne sur tablette pour la version *ScratchJr* destinée aux plus jeunes, et sur ordinateur pour la version *Scratch* traditionnelle. Les élèves peuvent y créer des jeux vidéo, des histoires interactives et donner vie à un scénario que l'enseignant aura, par exemple, déterminé à l'avance. En proposant un fonctionnement simplifié, cette application permet aux élèves de devenir de vrais auteurs et mêmes acteurs de leurs histoires, et donc, de leurs apprentissages. ►



Figure 1 – Une élève apprend à programmer sur Scratch

Les robots

Swift Playgrounds: Swift Playgrounds est une application accessible sur iPad qui a été développée par Apple. Elle s'avère être l'une des meilleures applications d'apprentissage du code, car elle mêle qualité graphique et intérêt éducatif. En effet, tout en donnant vie à l'histoire affichée à l'écran, les élèves peuvent voir le code qu'ils y appliquent sur la fenêtre attenante. Cette application affiche donc des qualités indéniables d'outil pédagogique et ludique au service des élèves.

Tickle: Avec Tickle, là aussi une application d'apprentissage du code accessible sur tablette, les élèves vont pouvoir programmer différents types d'outils, y compris des robots. L'intérêt de cette application pour les élèves est qu'ils peuvent programmer sur un environnement proche de celui de Scratch, c'est-à-dire relativement intuitif et simple, et appliquer le programme créé à différents robots (Lego WeDo 2.0, BB-8, le drone Airborne cargo, Sphero ou encore Dash & Dot). L'intérêt est donc ici clairement le croisement entre une application de création de programmes telle que Scratch et une concrétisation de la production des élèves sur des robots physiquement actifs.

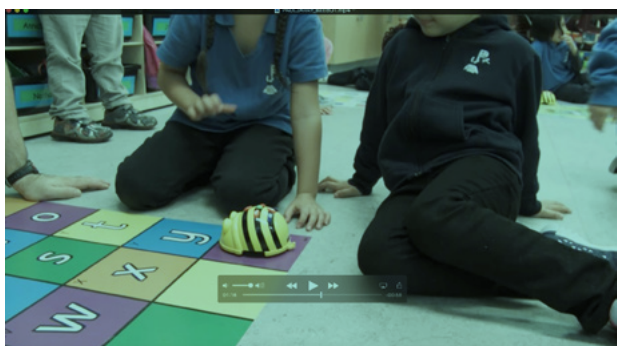


Figure 2 – Le robot Bee-Bot utilisé à l'école primaire

Bee-Bot : pouvant être utilisé dès la maternelle, ce robot, au physique proche de celui d'une abeille, permet une douce introduction à la programmation grâce à ses fonctions simplifiées de guidage. Ainsi, en appuyant sur les touches directionnelles sur son dos et en lançant le programme, les enfants donnent vie à Bee-Bot et peuvent lui faire faire un parcours complet, tout en apprenant les bases de la logique computationnelle.

Dash : ce robot, qui roule et émet des sons, peut, grâce à une application extérieure sur tablette tactile, être programmé, et donc guidé par les élèves. Plusieurs applications, à la logique proche de celle du logiciel Scratch, c'est-à-dire utilisant le « glisser-déposer » de boîtes graphiques et textuelles, peuvent permettre aux élèves de faire bouger le robot. Il s'agit probablement d'un des robots les plus complets en termes d'outils d'apprentissage de la programmation.

Le robot humanoïde NAO : NAO est un nouvel outil qui permet, associé au logiciel Chorégraphe, d'enseigner la programmation aux élèves. Son usage inédit dans un tel contexte a fait l'objet de recherches à l'Université de Montréal. Sa proximité physique avec l'être humain en fait un outil d'excellence, bien que complexe, d'enseignement/apprentissage de la programmation.

Retour sur des projets qui fonctionnent

Afin d'illustrer nos propos, voici un bref retour sur trois projets d'enseignement/apprentissage de la programmation que nous avons mené au Québec dans des établissements au primaire et au secondaire.

SCRATCH ET BEE-BOT : Le premier concernait l'usage de robots et d'applications à l'école primaire. Nous avons ainsi étudié les effets de l'application Scratch, et de Scratch Jr, sur des élèves du primaire. Ces derniers ont programmé des histoires (par exemple sur Halloween), avec certaines contraintes, telles qu'un nombre de scènes à respecter, un scénario à suivre, un certain nombre de personnages à adapter, etc. Les potentialités créatives qu'offre cette application, qui plus est simple d'utilisation, en fait un outil complet de développement de la logique informatique, de l'éducation à la programmation, de la stimulation créative et même de l'apprentissage de l'écriture et de la lecture. Un autre outil que nous avons étudié, au potentiel étonnant, est le robot Bee-Bot. Ce robot, que nous avons présenté précédemment, est un robot de sol qui répond aux commandes que l'utilisateur formule en actionnant des boutons directionnels sur son dos. Au-delà du fait de pouvoir effectuer des parcours proposés par l'enseignant, il est possible pour les élèves de créer des figures en plaçant un stylo sur le dos du robot, par exemple, ou en utilisant Pro-Bot, un robot-voiture ressemblant à Bee-Bot, avec un support à crayon prévu à des fins de création graphique ou artistique.

NAO : Dans un autre projet de recherche, cette fois au primaire et au secondaire, nous avons demandé aux élèves de programmer le robot humanoïde NAO. Pour ce faire, nous avons conçu un programme scolaire d'apprentissage du code appelé « Devenez le Maître NAO » dans lequel nous proposons 10 niveaux, composés chacun de 3 tâches à réaliser, soit 30 tâches au total, que devaient effectuer les élèves. Ils apprenaient ainsi à programmer le robot, à le faire parler, à le faire avancer ou reculer, et même à le faire danser. Ce projet, innovant et inédit, a permis, tant aux élèves du primaire que du secondaire, tous d'ailleurs issus de l'adaptation scolaire, de développer de multiples compétences. En effet, ces élèves ont développé des compétences en logique computationnelle, certes, mais aussi en mathématiques, en résolution de problème, en écriture, en lecture, tout en développant leur motivation, leurs aptitudes au travail collaboratif et coopératif, leur créativité, etc. Aussi, ce projet proposé comme nous l'avons dit à des élèves en adaptation scolaire, leur a permis de développer leur estime de soi et de se sentir plus performants et plus en confiance à l'école. Nul doute que de telles initiatives pourraient rapprocher les élèves de l'informatique, mais aussi, simplement, des environnements scolaires.

MINECRAFT : Un autre projet a été mené avec le jeu vidéo Minecraft, deuxième jeu vidéo le plus vendu au monde après Tétris. Dans Minecraft, les élèves peuvent déplacer des cubes pixellisés, les détruire, les assembler, pour former un objet/outil ou autre, etc. En fait, avec ce jeu, les potentialités créatives sont multiples et presque sans limites, si ce n'est celles de l'imagination des joueurs. Aussi, il est possible de demander à des élèves de créer n'importe quel environnement, n'importe quel objet, n'importe quel monument. Dans le cadre de notre projet, intitulé « Devenez le maître Minecraft », nous avons pu voir que les élèves développaient avec un tel outil de nombreuses compétences en matière de collaboration, de persévérance, mais aussi qu'ils s'entraidaient très souvent, et cela, avec un grand plaisir et une forte motivation. En adaptant les niveaux et les activités demandées aux élèves (30 activités, sur 10 niveaux de complexité), nous avons pu faire travailler les élèves sur certaines compétences en histoire, en mathématiques, en géographie, mais aussi en sciences et en programmation informatique avec la rédaction de lignes de code. Les élèves participant au projet faisaient même tout ce qui était en leur pouvoir pour prolonger les séances et rester après l'école, juste pour passer encore un peu de temps devant le jeu vidéo Minecraft. Un grand succès donc pour ce jeu commercial qui a, depuis, développé une version éducative appelée « Minecraft Education Edition ». ►



Figure 3 – Deux enfants programment NAO



Figure 4 – NAO dans un établissement d'enseignement adapté



Figure 5 – Deux élèves construisent une réplique fidèle du célèbre Titanic

Conclusion

Alors que de plus en plus de nouvelles technologies investissent les salles de classe, et que le numérique tend à être plus présent encore dans les années à venir, il est d'une importance capitale de préparer les élèves à maîtriser ces outils qu'ils vont retrouver dans le monde de demain. Pour ce faire, il est nécessaire de les aider à mieux comprendre la logique informatique derrière les téléphones intelligents : les applications ou simplement les téléphones en eux-mêmes. Pour cela, et comme nous en avons fait état dans cet article, il est possible de les éduquer à la programmation avec certains outils, simples d'usage, et terriblement efficaces. En effet, en dehors du développement de leur motivation à se rendre à l'école, mais aussi à apprendre, nous avons constaté que les élèves avaient développé de nouvelles compétences en matière de programmation, de logique, de résolution de problème et même de travail collaboratif, dans l'ensemble des projets que nous avons menés ces derniers mois. Et même si tout le monde n'est pas familier avec les nouvelles technologies, et prêt dès demain à en faire un usage aguerri, nos expériences montrent qu'il n'est pas nécessaire d'être un informaticien, un programmeur ou autre, mais simplement d'être intéressé à cette problématique et d'essayer certains outils. Ainsi, par une simple prise en main de certains outils, il serait très facile pour tout enseignant, formateur, éducateur, de mettre en place un programme d'éducation à la logique informatique à destination des élèves. Et cela concerne les enseignants de tous les niveaux et qui accueillent tout type de public. Aussi, nous ne formulerons qu'une seule et unique recommandation : lancez-vous et essayez... ! Vous serez surpris, tant de vos capacités, que de celles de vos élèves, et ce pour le plus grand bien de ces derniers... ■

¹ <http://ecolebranchee.com/2016/02/22/les-eleves-de-la-colombie-britannique-apprendront-a-coder/>

² <http://ici.radio-canada.ca/regions/atlantique/2015/10/21/016-codage-informatique-programmation-education-ne-ecoles.shtml>

³ <http://www.cnetfrance.fr/news/programmation-a-l-ecole-les-eleves-francais-apprendront-a-coder-des-la-rentree-2016-39837908.htm>

⁴ https://www.washingtonpost.com/lifestyle/kidspost/minecraft-spawns-classroom-lessons/2013/03/14/717aed66-87b8-11e2-98a3-b3db6b9ac586_story.html

⁵ <http://www.pcgamer.com/minecraft-becomes-a-compulsory-class-for-swedish-school/>

⁶ <https://code.org/>

Références

Duncan, C., & Bell, T. (2015). A Pilot Computer Science and Programming Course for Primary School Students. In *Proceedings of the Workshop in Primary and Secondary Computing Education* (p. 39–48). New York, NY, USA : ACM. <https://doi.org/10.1145/2818314.2818328>

Mubin, O., Stevens, C. J., Shahid, S., Mahmud, A. A., & Dong, J.-J. (2013). A REVIEW OF THE APPLICABILITY OF ROBOTS IN EDUCATION. *Technology for Education and Learning* 2013, 1(9). <https://doi.org/10.2316/Journal.209.2013.1.209-0015>

OECD. (2015). *Schooling Redesigned* | OECD READ edition. Consulté à l'adresse http://www.keepeek.com/Digital-Asset-Management/oecd/education/schooling-redesigned_9789264245914-en

Papert, S. (1981). *Jaillissement de l'esprit scientifique : ordinateurs et apprentissage*. Flammarion.

Smith, N., Sutcliffe, C., & Sandvik, L. (2014). Code Club : Bringing Programming to UK Primary Schools Through Scratch. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (p. 517–522). New York, NY, USA : ACM. <https://doi.org/10.1145/2538862.2538919>
