

## **Proposition d'orientations générales pour un programme d'informatique à l'école primaire**

Ce texte propose des orientations générales permettant de structurer un futur programme d'informatique à l'école primaire.

Comme dans les autres disciplines fondamentales, la sensibilisation précoce aux grands concepts de la science et technique informatique est essentielle. Elle donne des clés aux élèves pour comprendre le monde qui les entoure, elle évite que se forment des idées fausses et représentations inadéquates, elle fabrique un socle sur lequel les connaissances futures pourront se construire au Collège et au Lycée. À l'École, il est important de montrer les liens qui unissent les concepts de l'informatique et ceux enseignés dans les autres disciplines, ainsi que ceux qui les unissent aux objets familiers que les élèves utilisent tous les jours. Sur ces deux points, nous pouvons nous appuyer sur des expériences longues et riches d'enseignements menées en France et hors de France.

Ces orientations s'inscrivent aussi dans une vision plus globale : après cette première sensibilisation à l'École primaire, vient le temps, au Collège, de l'acquisition de l'autonomie puis, au Lycée, celui de la maîtrise des concepts. Il est important de veiller à la progressivité et à la cohérence des programmes pour l'École, le Collège et le Lycée.

La question décisive de la formation des Professeurs n'est pas traitée dans ce document. Disons simplement qu'il nous semble essentiel que les *Écoles supérieures du professorat et de l'éducation* intègrent une formation, mais surtout une certification, en informatique.

---

L'école primaire doit être le temps de la découverte des concepts fondamentaux de l'informatique, celui où l'on parle aux élèves, avec leurs mots, à partir de leur quotidien et de leurs connaissances acquises dans les autres disciplines, d'informations, de langages de programmation, d'algorithmes et de machines. L'enseignement de l'informatique à l'École nous semble être trop souvent limité à l'utilisation d'ordinateurs et de logiciels créés par d'autres. Cette vision dénature une discipline scientifique et technique qui donne un rôle essentiel à l'abstraction et à l'expérimentation personnelle. Faire de l'informatique ne consiste pas à passer des heures devant un écran, mais à acquérir des notions fondamentales et universelles. L'initiation à l'informatique doit donc n'être liée ni à un ordinateur particulier, ni à un logiciel particulier, ni à un langage particulier. Elle doit par ailleurs chercher un équilibre entre des activités fondées sur l'utilisation d'un ordinateur et des activités « débranchées », c'est-à-dire ne recourant pas à une telle utilisation.

### **Des logiciels aux concepts**

Les activités s'effectuant avec un ordinateur débutent avec l'apprentissage des logiciels les plus courants : logiciel de courrier électronique, navigateur, moteur de recherche, logiciel de traitement de texte, tableur, etc. Cet apprentissage ne doit pas rester une fin en soi, mais aussi conduire à s'interroger sur le fonctionnement de ces objets, menant ainsi à découvrir certains concepts de l'informatique.

Beaucoup d'élèves, par exemple, savent envoyer un courrier électronique, mais ils ne cherchent pas toujours à savoir comment un tel message arrive dans la boîte aux lettres de son destinataire. Pourtant l'apprentissage de l'utilisation d'un logiciel de courrier électronique est une occasion de les amener à se poser cette question et à y chercher des réponses. Cette interrogation, cette énigme, peut être l'occasion d'une recherche collective, chaque élève proposant une hypothèse et critiquant celles des autres. Elle peut donner lieu à une contextualisation historique : l'acheminement d'un courrier électronique n'est peut-être pas si différente de l'acheminement d'un courrier postal, qui fonctionnait déjà dans l'Antiquité quand les ordres militaires irriguaient de vastes empires. Cette question permet donc d'introduire la notion de *réseau* – les ordinateurs sont reliés entre eux par des câbles ou par voie aérienne – et de *routage* – un message doit trouver son chemin dans le labyrinthe que constituent ces milliards d'ordinateurs reliés entre eux.

Le Web et ses logiciels – navigateurs, moteurs de recherche, etc. – permet de poursuivre la réflexion sur ces questions, mais aussi d'en poser de nouvelles. L'acheminement du contenu d'une page d'un serveur Web jusqu'à l'ordinateur de l'école suit les mêmes principes que l'acheminement d'un courrier électronique. À un certain niveau d'abstraction rien ne distingue un courrier du contenu d'une page web, et les méthodes permettant d'acheminer l'un permettent également d'acheminer l'autre. Mais de nouvelles questions apparaissent : quelle est l'origine des informations auxquelles on accède ? Qui écrit ? Où les informations sont-elles enregistrées ? Comment sont-elles identifiées ? Les élèves peuvent même concevoir leur propre page web et devenir ainsi des participants actifs du Web ; c'est le meilleur moyen de comprendre que n'importe qui peut dire n'importe quoi dans une page web, et de s'interroger sur la pertinence de l'information que l'on y trouve. Le Web est aussi l'occasion d'aborder la question de la recherche des pages contenant certains mots-clés, fondée sur la notion d'indexation : les moteurs de recherche recherchent d'abord les documents qui contiennent des occurrences des mots signifiants de la question qu'on leur pose. L'indexation conduit à une réflexion sur le sens d'un texte, puisqu'elle vise à identifier ce dont le texte parle, même si cette réflexion est souvent réduite à l'identification de mots clés. Or, la compréhension et l'extraction du sens d'un texte sont parmi les buts fondamentaux de l'enseignement de la langue ; il faut désormais y adjoindre une initiation à l'indexation et une réflexion critique sur la pertinence d'un texte vis-à-vis d'une question posée.

L'initiation à l'informatique doit aussi passer par la découverte des concepts fondamentaux de langage, d'information, d'algorithme et de machine, sans toujours utiliser un ordinateur pour cela.

### **Des langages simples**

Un langage formel se distingue d'une langue naturelle par sa spécialisation, son caractère

artificiel, le caractère limité de son lexique et la simplicité des règles qui régissent sa grammaire. Un exemple simple est le langage formé de quatre mots : « nord », « sud », « est » et « ouest » et d'une construction, la séquence, qui permet de former des suites de tels mots. Ce langage permet d'indiquer un chemin à suivre sur une grille carrée, par exemple sur le carrelage du préau d'une école. L'expression « nord, nord, nord, est, est, est, sud, sud, sud, ouest, ouest, ouest » indique ainsi à un élève de se déplacer de trois carreaux vers le nord, puis de trois carreaux vers l'est, puis de trois carreaux vers le sud et enfin de trois carreaux vers l'ouest, parcourant ainsi un carré sur le sol.

Ce même mouvement peut être exprimé dans un autre langage qui ne comprend que trois mots : « avancer », « tourner à droite » et « tourner à gauche », composés par l'opération de séquence : « avancer, avancer, avancer, tourner à droite, avancer, avancer, avancer, tourner à droite, avancer, avancer, avancer, tourner à droite, avancer, avancer, avancer, tourner à droite ».

L'apprentissage de ces deux langages permet de mettre en place de nombreuses activités et de poser de nombreux problèmes. Une première activité consiste à interpréter les instructions données par un autre élève ou à trouver la phrase qui commande d'aller d'un point du préau à un autre. C'est le jeu du « robot idiot ». Ce problème est le même que celui de la conception d'un programme dans un langage tel que Logo (il y a de nombreux enseignements à tirer de l'utilisation de Logo à l'École primaire), mais où un élève joue le rôle de l'avatar informatique afin de mieux s'approprier le lien qui s'établit entre une expression du langage et une action. On peut ensuite passer à des exercices plus élaborés, comme la traduction d'une expression d'un langage dans un autre – par exemple une expression formée dans le premier des langages présentés ci-avant dans le second –, la mise en évidence de la redondance d'un langage – par exemple, un « tourner à gauche » pourrait être remplacé par une séquence de trois « tourner à droite ». Il est aussi possible d'évoquer dans une telle activité la notion de bug : une petite erreur dans une instruction exprimée dans le second langage, par exemple un « tourner à droite » de trop, change complètement la trajectoire et envoie l'élève n'importe où.

Ce type d'activité permet aussi d'aider les élèves à comprendre, dans un cadre très simplifié, quelques-uns des traits essentiels de la langue écrite : son caractère conventionnel, la nécessité de règles et la correspondance entre les mots et les actions. Elle leur permet aussi de comprendre qu'il est possible de calculer, non avec des nombres, mais avec des mots.

### **Des langages moins simples**

Ces activités débranchées peuvent mener à des activités sur machine, par exemple à des activités de programmation dans un langage tel que *Scratch*, développé au *Massachusetts Institute of Technology* spécialement pour enseigner la programmation à l'École. Il permet d'assembler visuellement des instructions et de créer des tests et des boucles, afin d'animer de petits personnages. Ce langage est emblématique de cette démarche ludique où les élèves créent des objets informatiques, d'abord par un simple dessin, puis les animent et augmentent leurs savoir-faire au fur et à mesure de leur découverte personnelle des possibilités offertes. De telles activités de programmation – ou de codage – sont aujourd'hui fréquemment proposées aux élèves hors de l'École. Les mettre en œuvre pour tous les

élèves permettra à toutes et tous de profiter de ce type d'apprentissages essentiel dans le monde dans lequel nous vivons.

D'autres activités autour de la notion de langage sont liées à la programmation de robots physiques animés par des algorithmes. Les clubs de robotique développent ce type d'activités et leur efficacité auprès des enfants est un fait avéré.

Des activités plus difficiles peuvent être proposées à la fin de l'École primaire : des exercices visant au rangement, à la classification de données, à l'analyse de multiples situations combinatoires simples, à la recherche d'objets ayant certaines propriétés dans un ensemble fini d'objets. Les expériences menées hors de France et en France, à l'École et hors de l'École, montrent qu'il est possible d'aller assez loin dans cette direction, même avec de jeunes enfants.

### **La notion d'information**

La notion d'information est aussi une formidable clé pour entrer dans l'informatique. La première notion à transmettre est celle de représentation : toute information peut être représentée par une suite de lettres dans un alphabet fini, par exemple par une suite de 0 et de 1. Les images, les sons, les textes, les nombres ont tous une représentation en machine, qui permet de les mémoriser, de les transmettre, de les transformer et de les reproduire à l'infini. Il est possible dès l'école primaire d'introduire l'atome d'information, le bit, et de se demander combien de bits sont nécessaires pour représenter une information. Pour exprimer si la lumière est allumée ou éteinte, un bit suffit, alors que pour décrire la couleur des cheveux d'une personne – bruns, châains, blonds ou roux – deux bits sont nécessaires. Pour décrire une couleur parmi les 16 777 216 du système RVB, vingt-quatre bits suffisent. Ici apparaît la notion de quantité d'information contenue dans un message, qui est, en première approximation, sa taille.

Ces notions peuvent être introduites par des jeux. On peut, par exemple, proposer un langage pour coder un petit dessin en noir et blanc : il faut pour cela décomposer le dessin en pixels, puis coder chaque pixel, qui est ou bien noir ou bien blanc, par un bit. Par exemple, en supposant que 1 code pour un pixel noir et 0 pour un pixel blanc, la suite de vingt bits 11111000111100011111 représente un dessin bi-dimensionnel :

```
1 1 1 1
1 0 0 0
1 1 1 1
0 0 0 1
1 1 1 1
Ou encore
■ ■ ■ ■
■
■ ■ ■ ■
      ■
■ ■ ■ ■
```

Deux élèves ou groupes d'élèves, de part et d'autre d'un paravent, peuvent ainsi s'échanger,

par oral, des dessins sur une grille de vingt-cinq ou cent pixels. Il est cependant probable que des erreurs apparaissent lors de la transmission, ce qui sera l'occasion de s'interroger sur la manière de corriger ces erreurs, par exemple en introduisant une forme de redondance dans le message. Il est aussi possible d'envoyer les bits à l'envers, du dernier au premier, et de s'interroger sur l'effet de cette transformation sur le dessin.

### **La notion d'algorithme**

Un algorithme est une manière de résoudre un problème en effectuant des opérations élémentaires mécaniquement et donc sans réfléchir. Tout algorithme doit s'exprimer d'une part dans une langue naturelle, ce qui est nécessaire pour sa compréhension, et d'autre part dans un langage technique précis, ce qui est indispensable pour faire en sorte que la machine puisse l'effectuer automatiquement. Les algorithmes qui transforment des symboles écrits – addition, soustraction, déclinaisons, conjugaisons, etc. – sont aussi anciens que l'écriture. Mais l'humanité a bien entendu utilisé des algorithmes avant même la naissance de l'écriture, pour tisser des étoffes, tailler des silex, etc.

Comme les notions de langage et d'information, la notion d'algorithme peut s'aborder par des activités « débranchées ». L'initiation peut commencer par l'identification d'algorithmes simples que les élèves utilisent tous les jours : pour s'habiller il faut mettre son tee-shirt avant son pull ; pour faire une tarte aux pommes, il faut mettre les pommes avant la cuisson de la pâte, mais pour une tarte aux fraises, il faut mettre les fraises après la cuisson.

Une deuxième étape est une interrogation sur les constructions qui permettent d'exprimer un algorithme :

- la séquence : faire ceci puis cela ;
- le test : si telle condition est vérifiée, alors faire ceci, sinon faire cela ;
- la boucle : faire ceci trois fois, ou alors répétitivement jusqu'à ce que telle condition soit vérifiée.

La notion d'algorithme est une formidable opportunité de relier l'informatique aux autres disciplines enseignées à l'École telles le français, les mathématiques ou les travaux manuels, car beaucoup des connaissances enseignées dans ces disciplines se formulent sous la forme d'algorithmes. Par exemple, l'algorithme de l'addition de deux nombres de trois chiffres décimaux peut être décomposé en une boucle à l'intérieur de laquelle se trouvent des instructions élémentaires en séquence : la lecture d'un chiffre de chacun des nombres et de la retenue, la consultation d'une table qui permet d'ajouter trois chiffres, l'écriture d'un chiffre du résultat et celle de la retenue. De nombreux exercices de mathématiques proposés en cycle 2 nécessitent la mise en œuvre d'un algorithme, souvent formé d'une unique boucle, mais où les notions de donnée, d'instruction, de test, de terminaison apparaissent clairement.

Avant même l'apprentissage de l'algorithme de l'addition, l'apprentissage de l'art de compter des objets recèle une possibilité de poser plusieurs questions d'informatique. Quoique l'on compte, il faut commencer par choisir arbitrairement un élément auquel on attribue le nombre 1, puis choisir un deuxième élément, distinct du premier, lui attribuer le nombre 2, etc. Des questions essentielles apparaissent : comment se saisir d'un élément, lui attribuer un nombre et le marquer afin de ne pas le compter deux fois, tout en n'oubliant aucun élément dans

l'énumération. Cette question du marquage est résolue de façons différentes selon les objets comptés : s'il s'agit des billes contenues dans un sac, on se munit d'un deuxième sac dans lequel on transfère une à une les billes déjà comptées. Si on compte des croix dessinées sur un cahier, on les entoure d'un cercle. Mais quel algorithme utilise-t-on pour compter les tuiles sur un toit ? Ou les arbres dans une forêt ?

De même l'accord d'un participe passé peut se décomposer en deux tests imbriqués, l'un sur le verbe auxiliaire utilisé, l'autre sur la présence d'un complément d'objet situé avant le verbe. Ces algorithmes s'apprennent en mathématiques, en français, etc., par imitation d'exemples de difficulté graduelle. En informatique, il est possible de porter un nouveau regard plus systématique sur certains de ces algorithmes et de montrer comment ils se construisent à partir des constructions de séquence, test et boucle. Il est aussi possible de commencer à sensibiliser les élèves au fait que ce sont les mêmes constructions qui permettent de construire les algorithmes appris en français, en mathématiques ou en travaux manuels, anticipant ainsi la notion d'universalité, qui sera développée dans la suite du cursus des élèves.

### **La notion de machine**

La notion de machine peut paradoxalement, elle aussi, être abordée par des activités débranchées. Il est par exemple possible de montrer aux élèves combien il est difficile pour eux de se comporter comme des robots, et de les amener à s'interroger sur l'origine de cette difficulté.

Le jeu du « robot idiot » peut être réutilisé ici, non pour réfléchir à la notion de langage, mais à celle d'architecture des machines. On peut fabriquer de petites cartes avec les mots « avancer », « tourner à gauche », « tourner à droite » et donner au robot humain un paquet de cartes qui est son « programme ». L'élève n'a désormais plus le droit d'écouter ce que lui disent ses camarades, mais doit uniquement lire une carte, exécuter une action et passer à la carte suivante. Pour d'autres programmes, tels le programme de l'addition, on pourra lui adjoindre des boîtes en carton – des variables – dans lesquelles il pourra stocker une valeur. Il illustrera alors le fonctionnement d'un processeur qui lit une instruction, exécute une action et passe à l'instruction suivante.

Il est aussi utile d'attirer l'attention des élèves sur quelques éléments clés de l'histoire des machines. On pourra leur raconter comment les Péruviens calculaient avec des ficelles couvertes de nœuds – les *kipu* –, et comment un boulier sert de béquille à notre mémoire : on y « pose » les chiffres des nombres à additionner en déplaçant les boules, tandis que l'accumulation des boules permet d'additionner. On peut montrer la conception au XIX<sup>e</sup> siècle, de machines munies d'un moteur qui ne nécessitaient pas d'intervention humaine pour effectuer un calcul. Enfin au XX<sup>e</sup> siècle, illustrer la révolution conceptuelle des premiers ordinateurs, machines capables d'exécuter non plus un calcul particulier mais n'importe quel calcul, ce qui demande un langage pour les programmer.

Évoquer les machines du passé, du boulier au minitel, permet de s'interroger sur ce qui est invariant dans l'histoire des machines à traiter de l'information comme les concepts de représentation, d'algorithme, de langage, la généralité de l'ordinateur, etc. et, au contraire, de

ce qui évolue constamment : la vitesse de calcul et la capacité des machines, leur taille, leurs domaines d'application, leur liaison avec le monde physique etc. Cela permet aussi de développer un sens de l'histoire : comment vivions-nous, comment communiquions-nous, comme cherchions-nous de l'information avant l'informatique ?

### **Un enseignement adapté à l'élève et à son rapport au monde**

Ce qui est ébauché dans ce document répond à des objectifs spécifiques aux élèves de l'École primaire. Nous avons cherché à proposer des activités ludiques mais instructives au sens profond du terme, et adaptées aux connaissances et aux capacités des élèves. Mais l'essentiel n'est pas là. Il est d'abord dans le fait que l'informatique est une science et une technique faite d'abstractions, et que ces abstractions ne sont pas accessibles aux élèves directement. Elles doivent être appréhendées à travers des situations concrètes et de l'expérimentation personnelle. C'est le levier des activités débranchées, qui propose aux enfants de manipuler des concepts à travers des gestes de leur propre corps et la manipulation d'objets familiers. Ensuite, à partir de sept ans, les enfants entrent dans l'âge des « pourquoi », mais à un niveau métaphorique. On doit donc leur proposer des explications, certes provisoires, des concepts informatiques, mais en mettant en place un vocabulaire précis qui leur permettra en grandissant d'affiner et d'enrichir les concepts associés aux mots.

Les activités débranchées sont complémentaires des activités avec un ordinateur, notamment l'apprentissage des logiciels les plus courants et l'utilisation des logiciels pédagogiques disponibles sur les ordinateurs et les tablettes, par exemple ceux d'apprentissage de la lecture, qui conduisent aussi à se poser des questions d'algorithmique humaine et mécanique.

On dira « langage », « information », « algorithme », « machine » et les enfants grandiront en se souvenant que l'information est aussi une quantité qui se mesure, qu'un langage peut être une forme de codage très rudimentaire, qu'une méthode devient un algorithme quand on a éliminé tous les implicites de la langue ordinaire et qu'une machine n'est qu'un outil qui permet d'exécuter des algorithmes.

- Serge Abiteboul, Professeur au Collège de France (2012), membre de l'Académie des sciences, membre du Conseil National du Numérique.
- Jean-Pierre Archambault, Président de l'association Enseignement Public et Informatique (EPI).
- Gérard Berry, Professeur au Collège de France, membre de l'Académie des sciences et de l'Académie des technologies.
- Colin de la Higuera, Président de la Société Informatique de France (SIF).
- Gilles Dowek, Directeur de recherche à l'INRIA, Grand Prix de philosophie de l'Académie Française.
- Maurice Nivat, membre de l'Académie des sciences.

Le 6 décembre 2013