

Université de Franche-Comté
ESPE de Franche Comté

Mémoire

présenté le 26 mai 2016

pour l'obtention du Grade de

MASTER

“Métiers de l'Enseignement, de l'Education et de la Formation”

Mention 1^{er} Degré Professeur des Ecoles

sur le thème

Introduction du code informatique à l'école primaire

Projet présenté par
Aude Alavoine

Directeur
Frédéric Dadeau, maître de conférences (UFR ST, Université de Franche-Comté)
Examinatrice
Anne Héam, professeur agrégé (Lycée Pasteur)

Remerciements

Je tiens à remercier M. Frédéric Dadeau, mon directeur de mémoire, pour avoir accepté de diriger ce travail. Sa disponibilité, ses conseils et compétences m'ont permis de mener de projet à termes, et de ne pas abandonner dans les moments de doutes.

Je remercie également Mme Anne Héam d'accepter d'être membre du jury.

Je souhaite remercier les professeurs qui ont accepté que je fasse une intervention dans leur classe : M. Joseph Vernot, des CM2 de l'école Fourier, M. Christophe Chabot des CM1-CM2 l'école Fanart, Mme Audrey Dadeau de l'ULIS et Mme Delphine Giancatarino des CM1-CM2 de l'école des Terreaux. Mes remerciements vont également aux élèves de ces classes pour leur travail, leurs questions pertinentes et leur enthousiasme.

Je remercie le département informatique de l'Université de Franche-Comté pour m'avoir fourni les tablettes nécessaires aux interventions.

Enfin, je remercie mes collègues de l'école maternelle Fourier qui m'ont soutenue durant cette année et qui m'ont aidée dans mes recherches de classes tests.

Remerciements

Table des matières

REMERCIEMENTS	1
TABLE DES MATIERES	3
TABLE DES FIGURES	5
TABLE DES TABLEAUX	7
INTRODUCTION	9
CHAPITRE 1 : CONTEXTE ET PROBLEMATIQUE	13
I. HISTOIRE DE L'ENSEIGNEMENT INFORMATIQUE EN FRANCE DEPUIS 1966	13
I.1. LE PLAN CALCUL [1966-1974]	13
I.2. L'EXPERIENCE DES 58 LYCEES [1970-1980]	14
I.2.a. Formation des enseignants	15
I.2.b. Equipement	16
I.2.c Enseignement / vie associative	16
I.3. LE PLAN INFORMATIQUE POUR TOUS [1985-1992]	17
I.4. PLAN POUR LES NOUVELLES TECHNOLOGIES DANS L'ENSEIGNEMENT [1999-2015]	20
I.5. PLAN NUMERIQUE [2015-...]	20
II. PROBLEMATIQUE	21
CHAPITRE 2 : ANALYSE DES PRATIQUES	23
I. PRATIQUES A L'ETRANGER	23
I.1. AUX ETATS-UNIS	23
I.1.a. Dans le système éducatif	23
I.1.b. Initiatives extérieurs au milieu scolaire	24
I.2. EN GRANDE-BRETAGNE	29
I.3. PAYS BALTES	35
II. PRATIQUES EN FRANCE	38
II.1. LES CODING GOUTER	38
II.2. ISSY-LES-MOULINEAUX	38
II.2.a. Périscolaire	38
II.2.b. LeCube	39
III. BILAN DES PRATIQUES	39
CHAPITRE 3 : PROPOSITIONS D'INTERVENTION	40
I. LES ACTIVITES	40
I.1. ACTIVITES DECONNECTEES	41
I.1.a. Activité de représentation de nombres en binaire	41

Table des matières

I.1.b. Activité de représentations de données	43
I.1.c. Activité de tri	44
I.2. ACTIVITES CONNECTEES	45
II. DEROULEMENT ET PREVISIONS	48
III. LES CLASSES TESTS	49
CHAPITRE 4 : EXPERIMENTATIONS ET RESULTATS EN CM1-CM2	50
I. DESCRIPTIONS DES INTERVENTIONS	50
I.1. CLASSE DE CM2 A FOURIER	50
I.2. CLASSE DE CM1-CM2 A FANART	52
II. ANALYSE DES INTERVENTIONS	52
II.1. ACTIVITE DECONNECTEE : ECRITURE BINAIRE	53
II.2. ACTIVITE DECONNECTEE : REPRESENTATION DES DONNEES	55
II.3. ACTIVITE CONNECTEE SUR TABLETTE	57
III. BILAN	64
III.1. REUSSITES	64
III.2. DIFFICULTES	64
III.2.a. Le matériel	64
III.2.b. Gestion des élèves et du temps	65
CHAPITRE 5 : INTERVENTION EN ULIS	66
I. ADAPTATIONS SUR LES ACTIVITES	66
I.1. ACTIVITE DECONNECTEE : REPRESENTATION DE DONNEES	66
I.2. ACTIVITE CONNECTEE SUR TABLETTE	66
II. DESCRIPTION DE L'INTERVENTION	68
III. ANALYSE DE L'INTERVENTION	69
III.2. ACTIVITE DECONNECTEE : REPRESENTATION DE DONNEES	70
III.2. ACTIVITE CONNECTEE SUR TABLETTE	71
IV. BILAN	72
CONCLUSION	74
ANNEXE	77
BIBLIOGRAPHIE – NETOGRAPHIE	89

Table des figures

Figure 1 : Mitra 15 à l'Ecole Centrale de Lyon 1979	16
Figure 2 : Robot pédagogique Tortue	18
Figure 3 : Part des cours choisis en langues aux Etats-Unis en 2014.....	24
Figure 4 : Première étape du cours Code.org avec Angry Birds	26
Figure 5 : interface du puzzle 1.....	27
Figure 6 : code du premier puzzle.....	27
Figure 7 : Certificat d'Une Heure de Code	28
Figure 8 : Etape 6 du cours destiné aux enfants non lecteurs.....	28
Figure 9: Programme scolaire britannique - Partie Informatique, Justification	29
Figure 10 : Programme scolaire britannique - Partie Informatique, Objectifs	30
Figure 11 : Programme scolaire britannique - Partie Informatique, Key stage 1	31
Figure 12 : Programme scolaire britannique - Partie Informatique, Key stage 2	32
Figure 13 : Programme scolaire britannique - Partie Informatique, Key stage 3	33
Figure 14 : Programme scolaire britannique - Partie Informatique, Key stage 4	34
Figure 15 : Statistique du concours Castor Informatique en 2014.....	35
Figure 16 : Question 1 du niveau Première/Terminale 2014	36
Figure 17 : Correction de la question 1.....	37
Figure 18 : Notation binaire.....	43
Figure 19 : Représentation du A script	43
Figure 20 : Lightbot niveau 1-1 - But du niveau.....	46
Figure 21 : Lightbot niveau 1 - Explication des instructions	46
Figure 22 : Interface de Lightbot.....	47
Figure 23 : Planning de l'intervention.....	48
Figure 24 : extrait de la fiche pour la représentation de nombres en binaire.....	51
Figure 25 : Stratégie de résolution lors de l'utilisation des cartes 0 et 1.....	54
Figure 26 : extrait de la fiche pour l'activité binaire	54
Figure 27 : exemple du A et du N en script.....	55
Figure 28 : A et N en script faux.....	55
Figure 29 : Code Solution 1 Code Solution 2.....	56
Figure 30 : Lightbot niveau 1-1	57
Figure 31 : Lightbot 2-1 solution non optimale	59
Figure 32 : Lightbot 2-1 solution optimale.....	59
Figure 33 : Lightbot 2-4 robot bloqué.....	61
Figure 34 : Lightbot 2-4 solution avec motif non optimale	61
Figure 35 : Lightbot 2-4 solution optimale.....	62
Figure 36 : Lightbot 3-1 consigne.....	63
Figure 37 : quadrillage à taille humaine	67
Figure 38 : Lightbot et quadrillage sur feuille	68
Figure 39 : Résolution d'une grille pour l'activité de représentation de données	70

Table des tableaux

Tableau 1 : Répartition des enseignants pour la formation lourde par année et par discipline.....	15
Tableau 2 : Objectifs et Chiffres de Code.org	25
Tableau 3 : Résolution de la carte cible 10 avec les cartes des puissances de 2	42
Tableau 4 : Résolution de la carte cible 10 avec uniquement des 0 et des 1	42
Tableau 5 : pourcentage d'utilisation de P1 comme une procédure (et non comme un espace supplémentaire)	60

Introduction

Titulaire d'un Master Informatique, spécialité Conception et Développement de Logiciels Sûrs, j'ai choisi de me réorienter vers le métier de professeur des écoles. Cette formation initiale et mon expérience professionnelle du métier m'ont conduite à proposer un travail de recherche qui liait ces deux champs : l'enseignement de l'informatique, et plus précisément au cycle 3. Les étudiants découvrent l'algorithmique en première année de licence. On peut donner la définition suivante d'un algorithme : suite finie et non ambiguë d'opérations (d'instructions) permettant la résolution d'un problème. De fait, on peut affirmer que les élèves produisent déjà des algorithmes à l'école lorsque, par exemple, ils résolvent un problème mathématique. On peut donc s'imaginer que l'enseignement de l'algorithmique pourrait être enseigné dès l'école primaire, ce qui aurait des bénéfices dans les champs disciplinaires déjà présents à l'école, tels que la rigueur et la mise en place d'une démarche pour la résolution de problème peu importe sa nature. Par ailleurs, dans une société telle que la nôtre, l'informatique est présente tout autour de nous. Son enseignement répond de ce fait à un fort enjeu sociétal. Il est facile de comprendre à quel point nous sommes dépendants des « connaisseurs de l'informatique » lorsqu'un de nos appareils informatiques tombe en panne. Ainsi, on donnant des outils aux élèves dès leur plus jeune âge, on leur permet d'accéder à leur maîtrise et d'accéder à l'autonomie, qui est, rappelons-le, l'objectif de l'école. Il me semble important et nécessaire d'introduire l'enseignement de l'informatique dès l'école primaire. Cela permettra de travailler les points suivants :

- anticiper
- améliorer
- se tromper et rectifier
- tester
- raisonner
- parler et justifier ses choix
- travailler en équipe

Cependant, il faut me faut d'abord définir ce que j'entends par l'introduction de l'informatique. On peut distinguer trois champs d'enseignement possible :

- L'apprentissage de l'outil informatique : cela correspondrait aux objectifs du B2i, apprendre aux élèves à se servir de l'outil dans un but pratique (traitement de texte, recherche sur internet, ...)
- L'utilisation de l'outil informatique comme support d'apprentissage : utilisation de Tableau Blanc Interactif (TBI), tablettes, etc en classe, utilisation de logiciels à destination d'autres disciplines (GéoGébra en mathématiques, Stellarium en Sciences, etc)
- L'apprentissage du code informatique : programmer des algorithmes

Ces trois définitions ne s'opposent pas, elles se complètent. L'informatique est à la fois un outil, au service d'autres disciplines, et une discipline. Il est important de faire ces

apprentissages en parallèle, et ne pas se contenter seulement d'un apprentissage purement pratique jusqu'à l'entrée au lycée.

On voit depuis les années 1960 se succéder différents plans visant l'introduction de l'informatique à l'école. On peut donc s'interroger sur la succession de ces plans. Pourquoi semble-t-elle être une préoccupation importante des différents gouvernements ? Pourquoi l'introduction de l'informatique semble-t-elle compliquée à mettre à œuvre ?

Selon The World Economic Forum, la France est classée 26^{ème} sur 143ⁱ pour les opportunités prises à utiliser les technologies de l'information et la communication (TIC). Ce classement est obtenu par l'étude de :

- l'environnement TIC offert par le gouvernement
- la volonté du gouvernement à améliorer l'usage des TIC
- l'usage effectif des TIC par le gouvernement

Bien que cette place soit honorable, il est important de rappeler que la France a perdu 8 places depuis 2013. De plus, rappelons que nous sommes la 6^{ème} puissance mondiale. Nous avons les moyens techniques et économiques pour améliorer cette 26^{ème} place. Est-ce une des raisons de l'introduction de l'informatique à l'école ? Améliorer notre place au rang mondial ? On remarque que les 5 puissances nous précédant sont également positionné avant nous dans le rapport de The World Economic Forum. On peut supposer qu'il existe une corrélation entre ces deux places.

On remarque que, depuis les années 60, les gouvernements successifs ont mis en place des plans (et en conséquence des ressources économiques et humaines) pour introduire l'informatique à l'école. On relève cinq grandes périodes :

- le Plan Calcul de 1966 à 1975
- l'expérience des 58 lycées de 1970 à 1980
- le plan Informatique Pour Tous (IPT) de 1985 à 1992 (avec expérimentation de 1981 à 1985)
- le Plan pour les nouvelles technologies dans l'enseignement en 1999
- le Plan Numérique en 2015

La prise de connaissance et l'analyse de ces différents plans dans un premier temps permettra de faire émerger la problématique.

Nous nous intéresserons ensuite aux pratiques actuelles à l'étranger. Plusieurs pays sont en avance par rapport à la France, et il serait intéressant de s'y inspirer, tout du moins d'en avoir connaissance. On pense naturellement aux USA, avec notamment l'initiative Code.org, mais également à la Grande Bretagne. Les pays Baltes, plus particulièrement l'Estonie et la Lituanie, sont également pionniers dans ce domaine.

Pour chaque pays, une présentation des projets mis en place sera faite, puis une analyse des mises en œuvre de ceux-ci sera effectuée.

De plus, certaines initiatives existent déjà en France, comme par exemple à Issy-les-Moulineaux, qui proposent aux élèves dès 6 ans des cours de programmation sur le temps périscolaire, ou encore les coding goûter. Cela fera l'objet d'une troisième partie.

Introduction

Ensuite, à partir de ces éléments, une séance à destination d'élèves de CM1-CM2 sera proposée et testée dans trois classes de milieu social différent. La même séance, adaptée sera testée en ULIS (Unité Locale pour l'Inclusion Scolaire). L'analyse de ces interventions permettra de tirer un bilan sur l'introduction du code informatique en classe de cycle 3 et de répondre à la problématique.

Chapitre 1 : Contexte et problématique

Ce chapitre, qui fait l'objet du travail de recherche de Master 1, présente l'historique de l'enseignement de l'informatique depuis 1966 en France. Cette étude nous permet de poser la problématique du mémoire.

I. Histoire de l'enseignement informatique en France depuis 1966

Nous présenterons les cinq plans mis en place par les gouvernements successifs depuis 1966 pour permettre l'introduction de l'informatique dans le système scolaire français. Chaque plan sera expliqué, analysé et nous essaierons de comprendre ce qui a fonctionné et ce qui ne l'a pas. Nous verrons dans un premier temps le Plan calcul (1966-1974). Puis l'expérience des 58 lycées (1970-1980). Nous verrons ensuite le Plan Informatique Pour Tous (1985-1992). Dans un quatrième point, nous verrons le Plan pour les nouvelles technologies dans l'enseignement (1999-2015). Nous verrons enfin le Plan Numérique (2015-...).

I.1. Le Plan Calcul [1966-1974]

Le Plan Calcul, initié en 1966 par le Président De Gaulle est le premier plan dans l'histoire de l'informatique. Bien qu'il ne soit pas destiné à l'enseignement, il est important de l'analyser pour comprendre les plans qui ont suivi.

Il faut tout d'abord rappeler qu'en mars 1966, le Président De Gaulle décide de retirer la France des commandements intégrés de l'OTAN. Cette volonté de s'affranchir de la suprématie des Etats- Unis, présents sur son territoire depuis le début de l'OTAN en 1949 (avec notamment des charges nucléaires), est la prémisse au Plan Calcul. Celui-ci avait comme fonction d'assurer l'indépendance de la France en termes de gros ordinateursⁱⁱ. Il faut rappeler qu'à cet époque, l'ordinateur n'était pas un objet démocratisé comme actuellement. Il était réservé aux centres de recherches type Commission à l'énergie atomique et aux énergies alternatives (CEA) et Centre National de la Recherche Scientifique (CNRS). Il servait à effectuer des calculs, d'où le nom du plan. Le gouvernement prit conscience de l'importance de l'informatique pour le pays, surtout par la perte de Bull, plus grosse industrie informatique de l'époque, racheté par General Electrics en 1964, ce qui affaiblit l'industrie électronique française.

Le plan prévoyait deux axes de travailⁱⁱⁱ :

- Un axe politique :
 - o création d'une Délégation à l'informatique en septembre 1966, dirigée par Maurice Allègre, dont la mission est de gérer les budgets du plan, de définir ses principales orientations et d'en assurer l'application

- création de l'Institut National de Recherche en Informatique et en Automatique (INRIA) en janvier 1967, encore existant aujourd'hui.
- Un axe industriel : création d'une « super » entreprise française de l'informatique, nommée Compagnie Internationale de l'Informatique (CII) qui est en fait la fusion, imposée par l'Etat, des entreprises Compagnie Européenne d'Automatisme Electronique (CAE) et Société d'Electronique et Automatisme (SEA). La mission de cette entreprise était donc de produire et concevoir des ordinateurs 100% français à destination des entreprises et des institutions officielles.

Le Plan Calcul avait uniquement un objectif industriel. La Compagnie Internationale de l'Informatique est aujourd'hui la société Bull, rachetée par Atos en 2014. Atos fait partie des dix plus grands acteurs des Entreprises de Services Numériques.

Bien que le Plan Calcul n'ait pas abouti au souhait de Maurice Allègre de créer une entreprise de type Airbus dans le domaine informatique, à taille européenne, on peut tout de même constater une certaine réussite dans l'évolution de la CII devenue Bull.

En conséquence, on peut noter la fin du Plan Calcul en 1974 par la dissolution de la Délégation à l'Informatique, remplacée par Direction des industries électroniques et de l'informatique.

I.2. L'expérience des 58 lycées [1970-1980]

En parallèle du Plan Calcul, une expérience fut lancée dans 58 lycées en 1970^{iv}. Il faut rappeler avant tout que la réforme Haby sur le collège unique n'a pas encore eu lieu. Le lycée commençait dès l'actuel 4^{ème}. Le plan Calcul ayant des objectifs purement industriels, il était nécessaire d'éduquer les futurs utilisateurs de ces ordinateurs produits. Pour cela, il fallait permettre aux élèves de lycée de se familiariser avec l'outil. Cette expérience avait deux objectifs :

- L'introduction d'une formation de culture générale à l'outil informatique dans toutes les disciplines (i.e. : sans créer une nouvelle discipline)
- L'amélioration de la pédagogie par le recours à l'Enseignement Assisté par Ordinateur (EAO) et amener les enseignants à s'interroger sur le contenu de leur enseignement. L'EAO est une méthode qui prévoit l'usage d'un ordinateur pour dispenser un cours

Cette expérience était donc destinée entièrement à l'enseignement, mais pas à l'apprentissage de l'informatique comme défini précédemment. Il apprenait à se servir de l'outil informatique dans une perspective de métier futur. Bien entendu, on ne peut négliger l'impact financier positif pour les industries françaises telles que la CII.

I.2.a. Formation des enseignants

La place de la formation était importante durant ce plan. Le gouvernement en proposa deux types^v:

- Formation dite lourde d'un an de 1970 à 1976 de 528 enseignants au total. Il s'agissait d'une formation d'une année dans un centre universitaire pendant laquelle l'enseignant était totalement déchargé de ses heures d'enseignement.
- Formation dite légère de 1970 à 1980 d'entre 5000 et 6000 enseignants. Celle-ci était une formation à distance, proposée par le Centre National de Télé-Enseignement, complétée par quatre jours de stage en présentiel.

Les contenus des formations lourdes et légères étaient relativement semblables, mais en proportion moins importante pour la formation légère. Elles comprenaient l'apport de notions générales de l'informatique, une initiation à la programmation et une discussion sur les possibilités de l'informatique dans les différents champs disciplinaires.

Tableau 1 : Répartition des enseignants pour la formation lourde par année et par discipline

	1970 1971	1971 1972	1972 1973	1973 1974	1974 1975	1975 1976	Totaux par discipline
Disciplines Artistiques	0	1	3	1	0	0	5
Lettres	9	8	15	13	21	10	76
Langues	10	10	11	10	7	7	55
Philosophie	3	2	3	2	2	0	12
Histoire et Géographie	8	6	6	4	8	10	42
Sciences Économiques	8	5	8	12	6	4	43
Sciences Naturelles	6	6	3	6	6	5	32
Mathématiques	18	23	26	25	22	26	140
Sciences Physiques	13	19	12	16	10	5	75
Techniques Industrielles	1	7	3	3	4	4	22
PEGC - PETT (*)	4	9	0	0	1	10	24
Divers	0	2	0	0	0	0	2
Totaux par année	80	98	90	92	87	81	528

(*) N'ont été comptabilisés ici que les enseignants dont il n'a pas été possible de déterminer la discipline dominante.

Par l'analyse du tableau présenté en tableau 1, on remarque toutes les disciplines ont été concernées par les formations lourdes. Bien que l'on note une forte majorité pour les enseignants de mathématiques, si l'on regroupe les matières dites littéraires (5 premières lignes auxquelles on ajoute les sciences économiques) on obtient 233 enseignants. On regroupant les matières dites scientifiques (de la ligne 7 à la ligne 10), on obtient 269 enseignants. Il y a ainsi équilibre entre scientifique et littéraire.

Les deux types de formation étaient sur la base du volontariat. L'équilibre entre scientifique et littéraire de la formation lourde montre un réel entrain pour l'informatique pour les sciences de l'informatique, peu importe la formation initiale.

I.2.b. Equipement

Le plan comportait un volet économique, dans l'esprit du Plan Calcul, toujours en vigueur. Des accords furent passés avec des entreprises françaises pour l'équipement des établissements scolaires entre 1973 et 1976. Ce sont le Mitra 15 (créées par la CCI) et les T1600 (créés par Télémécanique) qui furent retenus.



Figure 1 : Mitra 15 à l'Ecole Centrale de Lyon 1979

On constate grâce à l'image du Mitra 15 donnée en figure 1 les contraintes dues à la possession d'une telle machine : on peut facilement imaginer les dimensions (environ 2m de hauteur sur 1m50 de largeur), on remarque la présence d'un ruban (la disquette n'apparue qu'en 1984 sur le Mitra 15, le cd, bien que créée en 1982 par Sony et Philips, ne fut démocratisé que tardivement car cher^{vi}), etc. Le nom Mitra est l'acronyme de Mini-machine pour l'Informatique Temps Réel et Automatique. Le terme « mini » fait sourire aujourd'hui.

I.2.c Enseignement / vie associative

Pour l'enseignement, il fut décidé de créer un langage spécial : le Langage Symbolique d'Enseignement (LES). Il avait, entre autre, l'avantage d'être un langage intuitif, naturel pour les élèves, et ne nécessitaient pas un apprentissage long et fastidieux. Par ailleurs, étant un langage français avec une syntaxe en français, cela donnait l'impression que l'informatique n'était pas qu'une science anglaise. Cela rendait l'informatique accessible aux yeux des français.

On relève par ailleurs trois créations associatives en relation directe avec le plan :

- La création de l'association Enseignement Public et Informatique (EPI) par les enseignants ayant suivis la formation lourde pour la plupart. L'EPI « *veut faire de l'informatique et des technologies de l'information et de la communication un facteur de progrès et un instrument de démocratisation* »^{vii}. L'EPI est encore aujourd'hui très active quant aux questions de l'enseignement de l'informatique.
- La création des clubs informatiques permet aux élèves et enseignants qui le souhaitent de mettre à profit leurs apprentissages dans des projets, ce qui avait comme finalité le perfectionnement de leurs connaissances. Ces clubs permirent également l'auto-formation d'autres enseignants n'ayant pu bénéficier des formations
- Création en 1971 de la Section Informatique et Enseignement (SIE) à l'IRNDP. La SIE regroupait des enseignants qui avaient bénéficiés des formations et qui, sur un temps déchargé d'enseignement, concevaient et réalisaient des logiciels à destination de l'enseignement, des fiches séquences, etc. La SIE les regroupait dans une bibliothèque pour permettre leur utilisation à d'autres établissements.

Cependant, d'un point de vue économique, le premier choc pétrolier de 1973 a marqué le début du déclin de la croissance française. Ceci peut être une explication à l'arrêt de l'expérience par le gouvernement français, dont les bénéfices financiers ne se feraient pas à courts termes.

Il est à déplorer que le rapport de l'évaluation effectuée n'ait été publié. Cependant, on peut sembler-il affirmer que l'expérience des 58 lycées fut une réussite pour les enseignants et les élèves qui en ont bénéficié.

I.3. Le Plan Informatique Pour Tous [1985-1992]

Le plan Informatique Pour Tous^{viii, ix, x}, lancé en 1985 et qui prendra fin en 1992, est dans la continuité du Plan Calcul lancé en 1966 par le Président De Gaulle et de l'expérience des 58 lycées. On peut noter trois enjeux :

- **Un enjeu industriel et économique** : encore une fois, l'enjeu majeur reste le soutien de l'industrie française. Pour la mise en place du plan, il faut se fournir en matériel, former des enseignants, et faire la maintenance du parc informatique. Ces trois composantes font, bien entendu, marcher l'économie française. De plus, l'achat de matériel en nombre conséquent à une industrie française permet à celle-ci de s'agrandir dans le but de devenir internationale. On retiendra principalement les entreprises, évidemment françaises, Bull, Thomson, SMT Groupil, Léanord^{xi}. On peut supposer qu'en formant dès le secondaire les élèves à programmer, on permet ainsi à ces mêmes entreprises d'embaucher du personnel déjà formé, elles n'auront pas à les former elles-mêmes à l'outil.

- **Un enjeu politique** : les nouvelles technologies, nouvelles depuis près de 40 ans, sont perçues comme essentielles pour l'avenir. Il est tout de ce fait indispensable pour les politiques d'œuvrer pour l'avenir, donc dans les enfants.
- **Un enjeu pédagogique** : l'idée étant qu'en utilisant la technologie, les pratiques pédagogiques s'améliorent car dans l'ère du temps, en phase avec la société.

Le gouvernement, en amont de ce plan, lança en 1981 un démarrage expérimental en Seconde, puis en 1982 en Première, en 1983 en Terminale, pour arriver à un ensemble de 78 lycées proposant cette expérimentation en 1984. Septembre 1985 fut le lancement officiel dans 50 000 établissements français. Durant ces 4 années, 100 000 enseignants furent formés^{xiii}, en plus des enseignants formés pour l'expérience des 58 lycées. Le gouvernement a ainsi pu profiter de ce temps pour améliorer et clarifier les objectifs de ce plan. A la rentrée septembre 1985, le plan se déclina de plusieurs façons :

- Mise en place progressive d'une option Informatique de la Seconde à la Terminale (une classe par année)^{xiii, xiv}: l'étude des programmes nous montre que l'enseignement au lycée portait sur trois composantes : l'informatique générale (connaître et comprendre le fonctionnement d'un ordinateur), l'analyse et la programmation (algorithmique, structure de données, etc), la société et l'informatique (place de l'informatique, enjeux, utilisation, etc)
- Mise en place d'atelier de programmation de l'école primaire à l'université avec différents objectifs suivant le niveau. En particulier dans les écoles primaires et collèges, des activités d'éveil humain et social, technologique et logistique sont les objectifs principaux. On retrouve les trois composantes enseignées au lycée, mais adapté à ces jeunes enfants. La partie programmation est d'ailleurs présente, notamment par le langage LOGO et une tortue, robot pédagogique type promobile Jeulin (cf figure 2).



Figure 2 : Robot pédagogique Tortue

- Mise à disposition de ces ateliers au public en dehors des heures scolaires : le plan IPT avait également un objectif sociétal, faire de l'ordinateur un objet du quotidien dans les familles.

Les ambitions du plan étaient élevées : enseigner l'informatique à la nation entière. Bien sûr, le gouvernement comptait sur le bénévolat du public non scolaire pour se rendre d'eux-mêmes aux ateliers disponibles.

Le gouvernement a mis en place des moyens conséquents pour le lancement du plan :

- Formation des enseignants de l'école primaire à l'université et de toute discipline (non formés pendant l'expérience des 58 lycées) par un stage approfondi de 50h sur 6 jours pendant les vacances scolaires. Sur la base du volontariat, cette formation suscita un engouement massif de la part des enseignants. Le gouvernement reçut 300 000 demandes pour 110 000 places. Cette formation permettait de donner une culture générale de l'informatique, et d'amener l'enseignant à poursuivre sa formation de manière individuelle et volontaire^{xv}.
- Equipement matériel : 12 000 réseaux d'un micro-ordinateur dans les écoles primaires et des nanoréseaux aux collèges, lycées et universités dans 12 000 établissements^{xvi, xvii}. Le choix de nanoréseaux plutôt que de micro-ordinateurs dans le secondaire et supérieur était économique. Le nanoréseau était organisé en une tête de réseau et jusqu'à 30 nanomachines. Pour le prix d'un micro-ordinateur, on pouvait s'équiper d'environ 15 nanomachines. Pour le secondaire et supérieur, il aurait fallu 30 micro-ordinateurs par établissements, ce qui représentait un coût colossal.

Le gouvernement avait donc une réelle envie de réussite de ce plan, les moyens étaient là, la motivation des enseignants également et la mise en œuvre semblait pertinente. Cependant, en 1986, la droite gagne les élections législatives et on peut noter une modification d'ambition du plan. Pour commencer, la circulaire du 17 octobre 1987 interrompt tout financement d'équipement informatique pour les écoles primaires, qui relèvent du budget communal. Cette interruption marque le début de disparité d'équipement et d'objectifs pédagogiques de l'informatique dans les écoles primaires. Par ailleurs, l'équipement du secondaire et supérieur reste à la charge du gouvernement. Cela apporte néanmoins beaucoup de contraintes financières : entretien et renouvellement du matériel notamment. De plus, la formation des enseignants a un coût considérable, les enseignants, prenant sur leur temps de vacances pour se former, percevaient une compensation d'ordre financière, le personnel formateur et la location des locaux de formation était à la charge du gouvernement.

Cependant, ce plan permit l'introduction de l'informatique au grand public. Rappelons l'engouement des enseignants à suivre la formation (300 000 postulants), ce qui marque ainsi le début de prise de conscience de l'importance de l'informatique dans la société. De plus, les élèves ayant reçu cet enseignant ont bénéficié de nouvelles pratiques pédagogiques, et ont acquis la réflexion nécessaire à la résolution d'algorithmes, des problèmes. Cette pratique amène donc un esprit réfléchi.

Ce plan aurait ainsi pu se poursuivre et atteindre ses objectifs s'il n'avait pas été interrompu aussi tôt. La formation massive des enseignants montre qu'il était conçu pour

durer sur le long terme. C'est pourquoi on entend que ce plan fut un échec. Or, c'est le premier plan d'une telle envergure, avec ces objectifs ambitieux, et une réussite en terme de formation, d'enseignement, et d'équipements.

I.4. Plan pour les nouvelles technologies dans l'enseignement [1999-2015]

Le Plan pour les nouvelles technologies de la communication et de l'information dans l'enseignement a été présenté à la presse en novembre 1997^{xviii}. Il avait pour objectif l'introduction des nouvelles technologies à l'école dans un but pédagogique : renouveler les pratiques pédagogiques et permettre aux élèves d'acquérir les bons réflexes quant à l'utilisation des outils numériques. Ce plan concernait tous les niveaux scolaires, de l'école primaire à l'université.

Il n'y eut pour ce plan aucune formation des enseignants, mais la création des MATICE (Maitre Accompagnateur TICE) permit aux enseignants non formés de recevoir une aide pour la mise en place de séquences utilisant l'outil numérique.

Le B2i fut créé en 2000 pour attester des apprentissages des élèves sur ces outils, avec des items à valider tels que l'utilisation d'un logiciel de traitement de texte^{xix}.

La volonté du gouvernement était donc la maîtrise de l'outil informatique dans un sens pratique. On peut ainsi affirmer que ce plan a fonctionné : les élèves maîtrisent l'usage des outils numériques, ceux-ci ayant pris, au fil des années, une place de plus en plus importante dans le quotidiens des Français.

I.5. Plan Numérique [2015-...]

Le plan numérique est dans la continuité du plan précédent. Il s'inscrit dans le plan France numérique 2012-2020^{xx}. Pour préparer au mieux à cela, les élèves doivent donc être formés le plus tôt possible :

- Aux utilisations possibles du matériel avec, par exemple, les collèges connectés et une tablette par élève de 5^{ème} d'ici 3 ans
- A l'algorithmique par une initiation à cet apprentissage dès le CE1 de manière facultative, et obligatoire dès le secondaire

Pour cela, le gouvernement prévoit le financement d'un milliard d'euros sur 3 ans en soutien aux collectivités pour permettre l'équipement des écoles, collèges et lycées, et à la formation des enseignants.

Le gouvernement a lancé en janvier 2015 une concertation nationale où 60 000 participants ont répondu aux questionnaires en ligne et aux rencontres académiques. De ces réponses^{xxi} sort un fort intérêt pour le numérique en classe, que ce soit de la part des élèves et parents ou de la part des enseignants. On peut aussi rappeler que Benoit Hamon précisait que "97% des enseignants jugent que le numérique est un atout pour les élèves mais seulement 5% l'exploite en classe^{xxii}. Il semble donc manquer la formation des enseignants à l'utilisation du

numérique, et un équipement conséquent. A Besançon, notons qu'il n'y a, en moyenne, que trois ordinateurs par classe du 1^{er} degré. Bien que ce soit un budget conséquent pour la commune de Besançon, ce n'est pas assez pour en faire un réel usage dans les classes.

Le principal argument au plan numérique est le renouvellement et la diversification des pratiques pédagogiques, dans le but de réduire les inégalités et décrochages scolaires. En généralisant, par exemple, les livres scolaires numériques, on pourrait plus facilement différencier les exercices et les modalités d'apprentissages et d'évaluation des élèves. Par ailleurs, les élèves connaissent et utilisent ces outils à la maison. On peut ainsi espérer une certaine motivation de la part des élèves, ce qui semble manquer actuellement à l'école.

Il semble que l'utilisation du numérique et l'apprentissage de l'outil prenne une place importante dans ce plan, l'apprentissage du code informatique de manière obligatoire faisant débat actuellement. Le principal argument à ces débats étant la question de la formation. Peu d'information sur ces formations sont disponibles, et il est naturel que les enseignants s'en inquiètent. Ce plan a débuté en septembre 2015 dans 500 écoles et collèges expérimentaux pour se généraliser d'ici 3 ans sur l'ensemble du territoire. On note néanmoins un réel intérêt quant à l'introduction et l'utilisation de l'outil numérique au sein de l'école.

On peut mentionner également une certaine inquiétude de la part des parents concernant l'équipement (son entretien, son financement, mais également d'un point de vue sécurité sur internet)^{xxiii}.

Enfin, ce plan étant la continuité du plan précédent, l'utilisation des outils est donc naturellement reconduite, avec l'ajout des nouvelles technologies telles que les tablettes. L'introduction du code est plus problématique, car les enseignants ne sont pas encore formés. On peut cependant penser que ce plan peut réussir car il intervient dans un contexte favorable aux nouvelles technologies, qui sont présentes dans les foyers depuis quelques années déjà. On voudrait maintenant amener les élèves à réfléchir sur le fonctionnement du matériel, des logiciels, pour leur permettre d'avoir un esprit critique et leur permettre d'accéder à l'autonomie.

II. Problématique

On remarque que l'informatique prend de plus en plus de place dans la société depuis une cinquantaine d'années. C'est donc naturel que l'informatique entre dans l'école, passage obligatoire pour la vie en société. Cependant, l'étude des différents plans montre qu'il y a plusieurs façons d'accéder à l'introduction de l'informatique dans l'enseignement. On peut soit introduire l'outil au service de la pédagogie et des autres disciplines, soit en tant que discipline à part entière. Il est important d'amener toutes les possibilités de l'informatique, pour permettre aux élèves d'avoir une formation complète.

Par l'historique des différents plans à l'école, on remarque une forte prédominance de l'informatique comme support d'enseignement. L'apprentissage du code informatique était présent jusqu'à la fin du plan IPT, mais il a disparu complètement par la suite. Le dernier

plan tente une réintroduction de cet apprentissage mais se heurte aux appréhensions des différents partenaires de l'école, concernant sa mise en œuvre.

Or, l'enseignement de la programmation a de nombreux bénéfices :

- Développement de l'esprit logique : cela développe l'esprit de raisonnement, nécessaire pour comprendre et s'insérer dans la société.
- Développement de la capacité d'analyse : c'est la capacité à faire un examen minutieux d'une situation. L'école doit donc développer cela, et l'apprentissage de l'algorithmique y contribuera.
- Capacité d'abstraction : c'est la capacité à raisonner en l'absence de tout support concret, à travailler sur les représentations et les images mentales. C'est donc une capacité à développer tôt chez les élèves pour leur permettre un meilleur raisonnement.
- Résolution de problèmes : l'algorithmique est une suite d'actions dans le but de résoudre un problème, cette capacité sera donc réutilisable tout le long du cursus scolaire, mais également pour la vie professionnelle future.

Ceci m'amène à formuler la problématique suivante :

Comment introduire la programmation à l'école primaire ?

Cette question amène plusieurs pistes de réflexion :

- Il faut tout d'abord clarifier les enjeux de l'introduction du code, de la programmation dès l'école primaire.
- On peut ensuite s'interroger à quel public s'adresser : on restreindra le travail du mémoire sur le cycle 3 pour plusieurs raisons, la principale étant que les sciences aux cycles 1 et 2 sont des initiations et découvertes. On pourra bien entendu initier les élèves dans ces premiers cycles, mais un travail plus approfondi sera fait sur le cycle 3.
- Il faut ensuite se demander quel environnement de travail sera utilisé : la psychologie infantile nous aiguille sur ce point, il faut garder un aspect ludique pour intéresser les élèves. L'usage de tablettes est bien sur une idée, mais concernant les ordinateurs, il faudra probablement garder une interface graphique avec par exemple le logiciel Scratch, développé par le MIT^{xxiv}, adapté aux jeunes enfants.
- Il ne faudra pas négliger l'aspect formation des enseignants : faudra-t-il former massivement les enseignants comme pour le plan IPT ? ou alors proposer des formations continues ? Il faudrait également intégrer la formation informatique dans les formations des professeurs dans les Ecoles Supérieures du Professorat et de l'Education (ESPE)

La suite du travail portera donc sur l'analyse des différentes initiatives, internationales et nationales^{xxv}. La réussite de l'opération Code.org lancé par le gouvernement américain en partenariat avec les industriels, l'introduction de la programmation dès le primaire au Royaume Uni et les initiatives dans les Pays Baltes devront être analysées, pour en extraire les moyens mis en place, la mise en œuvre effective, les réussites, les échecs, et l'accueil reçu. Différentes expérimentations seront effectuées dans diverses classes. Pour la réussite de ces

expérimentations, il y aura toute une analyse préalable autour des objectifs, du public visé et de la mise en œuvre. L'aspect ludique pourrait être apporté par le jeu vidéo. Enfin, ce projet arrivé à terme sera analysé pour répondre à la problématique.

Chapitre 2 : Analyse des pratiques

Nous étudierons dans ce chapitre les pratiques existantes. Plusieurs initiatives ont été lancées à travers le monde et en France. Il est donc important d'en prendre connaissance dans un premier temps, et de les analyser dans un second temps. Cette analyse permettra de mettre en avant ce qui fonctionne. Cela nous aidera à construire notre proposition.

I. Pratiques à l'étranger

Trois pays sont précurseurs dans le domaine de l'enseignement de l'informatique : les Etats-Unis d'Amérique, la Grande-Bretagne et les Pays Baltes. Nous analyserons ainsi les initiatives de ces pays, que ce soit dans le cadre scolaire ou non.

I.1. Aux Etats-Unis

I.1.a. Dans le système éducatif

Présentation

Plusieurs états d'Amérique du Nord ont décidé début 2015 par amendements d'intégrer l'apprentissage des langages informatiques au même titre que l'apprentissage des langues^{xxvi} étrangères. Ceci montre une forte volonté des gouvernements d'améliorer les compétences informatiques de leurs citoyens, mais également de permettre à plus de lycéens de sortir du secondaire avec les connaissances nécessaires pour suivre un cursus en Information Technology, qui jusqu'à présent, commençait seulement au supérieur.



Figure 3 : Part des cours choisis en langues aux Etats-Unis en 2014

Sur le graphique présenté en figure 3, on remarque que les lycéens américains choisissent principalement l'apprentissage de l'Espagnol puis des Sciences de l'informatique, avant l'apprentissage du Français, du Chinois et de l'Allemand. Ceci montre bien l'importance et l'engouement des sciences de l'informatique aux Etats-Unis.

Cependant, il n'y a que quelques états qui ont pris cette décision : Le Kentucky, le Texas et la Géorgie. L'état de Washington, un temps concerné, l'a mise en suspens pour prendre la meilleure décision pour l'intérêt des élèves. Cette mesure ne fait pas l'unanimité, car elle substitue l'apprentissage d'une langue étrangère au profit d'un langage informatique. Ces détracteurs prônent pour l'ajout d'une matière informatique à part entière, leur argument étant : « *Should geometry be substituted for history?* », Richard Barton, co-fondateur et président exécutif de Zillow Group, entreprise d'annonce immobilière.

L'argument des initiateurs de ces amendements est que d'ici 5 ans, les entreprises basées aux Etats-Unis offriront 1,4 millions d'emplois nécessitant des connaissances en sciences de l'informatique, pour seulement 400 000 diplômés. Il y aurait donc un déficit d'1 millions de diplômés.

Démarche

Les programmes scolaires américains diffèrent des programmes scolaires français. Ceux-ci ne sont pas établis par le gouvernement pour l'ensemble du pays, mais par les assemblées législatives de chaque Etat fédéré, qui décide d'un socle minimum commun de connaissances. Ensuite, chaque école détermine le programme des matières et leur mise en œuvre. Nous ne pouvons donc pas analyser le programme correspondant aux sciences informatiques, car il diffère probablement pour chaque école.

I.1.b. Initiatives extérieurs au milieu scolaire

Présentation

En parallèle des initiatives dans le milieu scolaire décidé par les gouvernements, l'association Code.org à but non-lucratif a décidé de lancer le projet « The hour of code » en 2013. Cette opération annuelle promeut le code informatique auprès des enfants principalement, mais également auprès de toutes personnes intéressées. L'association a pour but également, de toucher les femmes et les personnes de couleur, sous-représentées dans le milieu professionnel des sciences de l'informatique^{xxvii}.

Le tableau 2 regroupe tous les objectifs de l'association Code.org et leur accomplissement. Concernant la diversité de genre et d'appartenance ethnique, on remarque que 42% des utilisateurs des cours online sont des filles. Nous pouvons comparer ces chiffres aux moyennes de 11% d'étudiantes féminines en écoles d'ingénieur informatique et 20% en Licence Professionnelle Métiers de l'Informatique, du Traitement du de l'information et des Réseaux. Il semblerait que l'objectif de féminisation fonctionne.

Objectifs et Statistiques

Tableau 2 : Objectifs et Chiffres de Code.org^{xxviii}

Objectifs	Accomplishment
Diversifier davantage dans les sciences de l'informatique	Dans nos cours en ligne, 42% des élèves sont des filles et 37% sont Afro-Américains ou Hispanics. Dans nos classes de lycée, 34% sont des filles et 60% sont Afro-Américains ou Hispanics.
Inspirer les élèves	Des dizaines de millions d'élèves ont essayé The Hour of Code (142 498 661 ont été réellement effectués, pour 48% de filles)
Créer des cours intéressants	99% des professeurs interrogés recommandent les cours d'introduction aux sciences informatiques de Code.org.
Atteindre les salles de classe	203 192 professeurs se sont engagés pour suivre les cours de Code Studio et 5 709 144 étudiants ont suivi les cours.
Former les nouveaux professeurs	Nous avons formé 16 000 nouveaux professeurs à enseigner les sciences de l'informatique jusqu'au niveau Senior (équivalent Terminale).
Modifier les programmes scolaires	Nous sommes partenaires de 90 des plus larges districts scolaires pour ajouter les sciences de l'informatique aux programmes scolaires. Ces districts enseignent à environ 10% des élèves américains et 15% des élèves Hispanics et Afro-Américains.
Mettre en place des programmes politiques qui supportent les sciences informatiques	Les programmes politiques ont changes dans 16 Etats, dont la Californie, l'Etat de Ney York, la Floride, l'Illinois et l'Ohio.
S'étendre mondialement	Nos cours sont disponibles dans plus de 40 langues et utilisés dans plus de 180 pays.

L'association Code.org pense que l'apprentissage des sciences de l'informatique devrait être au curriculum scolaire au même titre que la biologie, les mathématiques, etc. C'est pour cela qu'elle propose des cours gratuits aux professeurs pour qu'ils puissent être formés, et ainsi avoir les compétences nécessaires pour donner ensuite des cours de programmation à leurs élèves^{xxix}. On peut ainsi s'attendre à ce que de plus en plus de cours d'informatique soit proposés aux élèves américains, de manière non obligatoire, grâce à cette initiative.

De plus, l'association propose ses applications dans plus de 40 langues. Elle a donc une réelle envie de d'accroître le nombre de personnes ayant accès à ses cours dans le monde, et pas seulement aux USA.

Le tableau 2, bilan des objectifs depuis la création, sert également de justification. L'initiative Code.org est recommandée par 99% des 203 000 professeurs l'ayant testé. Par ailleurs, on remarque que 16 Etats ont déjà changé leur politique concernant les sciences de l'informatique. On peut dire que Code.org a atteint ses objectifs, et que ces chiffres justifient l'importance de l'introduction du code informatique dès l'école primaire.

Démarche

Les cours en ligne proposé par Code.org sont très ludiques, et mettent en scène des personnages aimés des enfants. Par exemple, un des premiers cours était basé sur les personnages d'Angry Birds, Plants vs Zombie et Scrat de L'âge de glace^{xxx}. Chaque cours est composé de 20 étapes.

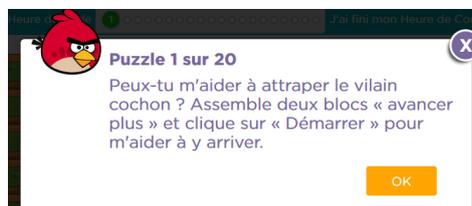


Figure 4 : Première étape du cours Code.org avec Angry Birds

La figure 4 montre la consigne de la première étape du cours avec Angry Birds. En partant du principe que les enfants sont totalement débutants, les premières étapes sont très guidées pour ne pas décourager l'enfant.

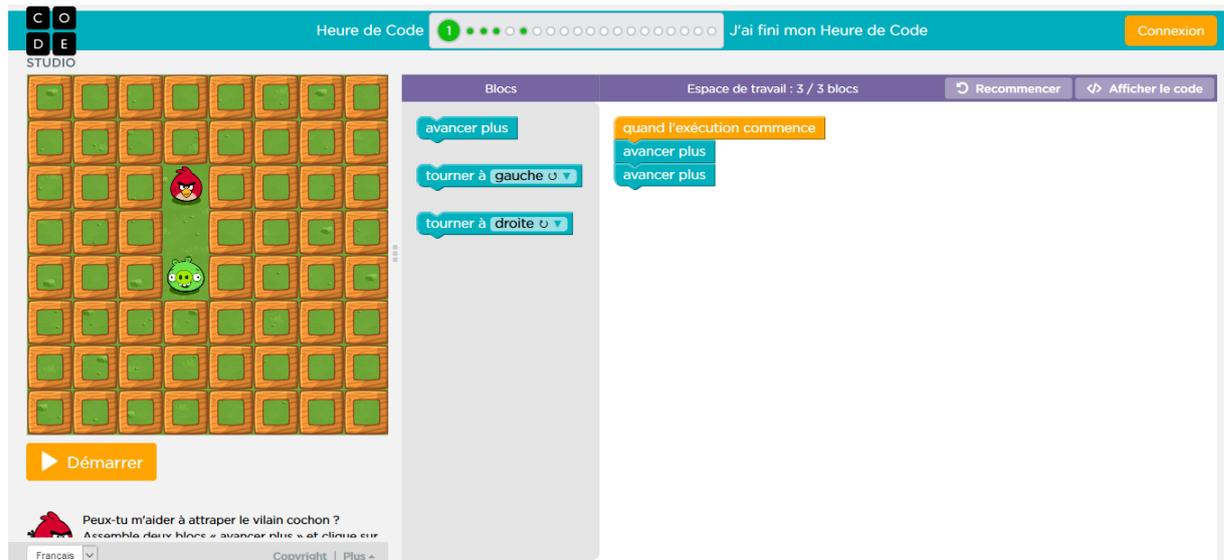


Figure 5 : interface du puzzle 1

Sur la partie gauche de la figure 5 se trouve la fenêtre de visualisation de l'exécution du code. Sur la partie droite, on voit les blocs que l'on peut utiliser pour résoudre le problème (ici « Avancer plus », « Tourner à gauche », « Tourner à droite »), et la partie code où l'on place nos blocs d'instruction. On remarque que la partie droite nous indique la meilleure solution en nombre d'instruction, ce qui nous amène à améliorer notre code par rapport à cela.

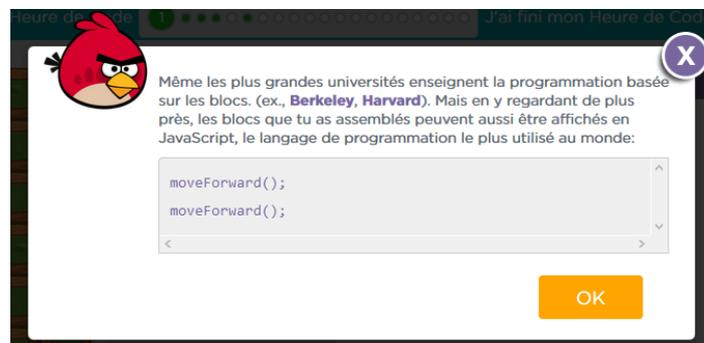


Figure 6 : code du premier puzzle

On remarque également dans la figure 5, le lien « Afficher le code », qui affiche la figure 6. Afficher le code permet aux enfants qui essaient Code.org de se rendre compte de ce qu'il y a derrière les instructions qu'ils ont à leur disposition, et que « même les plus grandes universités enseignent la programmation basée sur les blocs ».

Enfin l'exécution du programme à l'aide du bouton « Démarrer » est également ludique, avec les sons typiques des personnages.

Ensuite, les 19 autres étapes permettent la découverte d'autres types de blocs, avec par exemple le bloc « répéter n fois » à l'étape 6, le bloc « répéter jusqu'à » à l'étape 10, le bloc « si » à l'étape 14, le bloc « si-sinon » à l'étape 18. La dernière étape reprend plusieurs blocs découverts et est la résolution d'un problème plus complexe.



Figure 7 : Certificat d'Une Heure de Code

Enfin, une fois les 20 étapes réussies, un certificat est délivré à l'enfant (cf figure 7).

The hour of code est donc ludique et permet aux enfants de découvrir les bases de la programmation par bloc, en mettant en scène des personnages qu'ils aiment. Ce principe les motive donc.

Code.org propose également des cours en fonction du niveau scolaire des enfants, avec 20 heures de cours par niveau. On trouve quatre niveaux, en commençant par les élèves qui commencent seulement à lire (GS-CP), à des élèves lecteurs de 10 ans et plus. Les cours 2, 3 et 4 se suivent, il est donc conseillé de commencer par le cours 2. Le cours 1, destiné aux élèves non lecteurs n'est pas requis pour le cours 2.

Par exemple, le cours de niveau 1, destiné aux enfants non lecteurs, se présentent en 18 étapes^{xxxii}. Un plan de cours est proposé aux professeurs, et toutes les étapes ne requièrent pas l'utilisation d'un ordinateur.

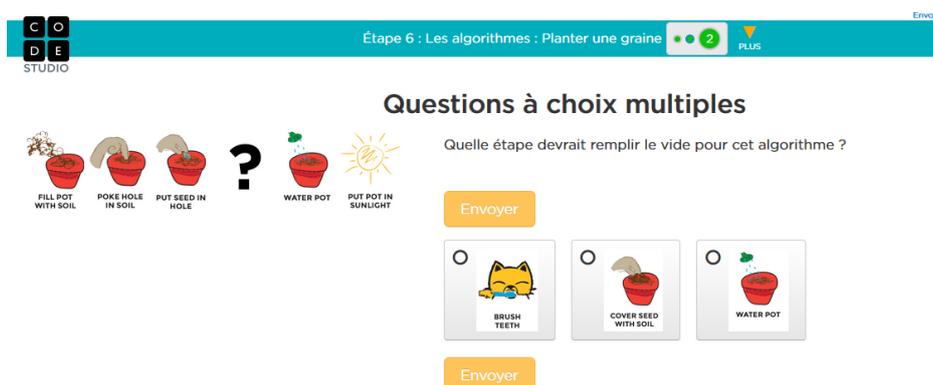


Figure 8 : Etape 6 du cours destiné aux enfants non lecteurs

Par exemple, la figure 8 montre que certaines étapes peuvent se faire sans ordinateur. Le travail proposé consiste à trouver l'élément manquant dans la séquence. Notons que ce travail peut être effectué à l'aide de flashcards sur table.

Que ce soit les cours de The hour of code, ou les 20h de cours par niveau scolaire, ils sont tous les deux ludiques, et adaptés au niveau des enfants. La progression est bien pensée, et donne envie aux enfants qui les testent de continuer.

I.2. En Grande-Bretagne

Présentation

Depuis les années 1990, les cours d'informatique en Grande-Bretagne sont des cours de bureautique. Ces cours, sous le nom d'informatique, ne satisfont pas les entreprises high tech britanniques. En 2012, le ministre de l'Education lance une consultation nationale afin de supprimer cet apprentissage désuet, au profit de l'apprentissage de la programmation^{xxxii}. Comme aux Etats-Unis, les entreprises sont les premières à avoir alerté le gouvernement des lacunes des jeunes diplômés de l'informatique. Elles se sont donc regroupées sous l'association Next Gen Skills, pour l'introduction de cours de sciences de l'informatique dans le cursus national^{xxxiii}.

Le gouvernement, ayant pris la mesure du problème, supprime le caractère obligatoire des cours d'informatiques, tels qu'ils sont proposés. Il laisse le choix aux écoles d'élaborer elles même le programme proposé aux étudiants, de gérer la formation des professeurs et le recours au secteur privé si besoin. Cependant, le gouvernement promet de mieux former les enseignants à l'apprentissage de la programmation en formation continue et initiale. Cette initiative peut paraître extrême, mais cela souligne bien le fait qu'il y a un manque énorme de professionnels dans ce domaine, et qu'il faut y remédier rapidement.

Les programmes scolaires officiels du gouvernement britannique^{xxxiv} incluent désormais l'apprentissage de l'informatique.

Purpose of study

A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with mathematics, science, and design and technology, and provides insights into both natural and artificial systems. The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of content. Computing also ensures that pupils become digitally literate – able to use, and express themselves and develop their ideas through, information and communication technology – at a level suitable for the future workplace and as active participants in a digital world.

Figure 9: Programme scolaire britannique - Partie Informatique, Justification

Dans la figure 9, on retrouve les objectifs principaux et la justification de ces apprentissages : **« Un enseignement de qualité dans les sciences de l'informatique permette aux élèves d'utiliser la pensée computationnelle et la créativité pour comprendre et changer le monde. »**. Le gouvernement britannique place ainsi

l'informatique comme matière obligatoire et nécessaire pour les élèves du key stage 1 au Key stage 4, c'est-à-dire de 5 à 16 ans.

Aims

The national curriculum for computing aims to ensure that all pupils:

- can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- are responsible, competent, confident and creative users of information and communication technology.

Figure 10 : Programme scolaire britannique - Partie Informatique, Objectifs

La figure 10 recense les objectifs d'apprentissage. A la fin du key stage 4, les élèves connaîtront les concepts fondamentaux des sciences de l'informatique, qu'ils auront mis en œuvre à plusieurs reprises lors d'écriture de programme de résolution de problème. Ils seront également des utilisateurs responsables et créatifs des technologies de communication et d'informations. Il y aura donc, au cours des 4 key stage, des cours de théories sur les concepts fondamentaux, tels que les représentations de données et les algorithmes, des cours de résolution de problèmes et de code, et enfin des cours d'éthique sur l'utilisation des technologies.

Démarche

L'analyse des programmes officiels nous indiquent la démarche utilisée par les enseignants. La scolarité obligatoire en Grande-Bretagne commence à 5 ans et termine à 16 ans. Elle est divisée en 4 key stage, l'équivalent des cycles en France.

Key stage 1

Pupils should be taught to:

- understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions
- create and debug simple programs
- use logical reasoning to predict the behaviour of simple programs
- use technology purposefully to create, organise, store, manipulate and retrieve digital content
- recognise common uses of information technology beyond school
- use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies.

Figure 11 : Programme scolaire britannique - Partie Informatique, Key stage 1

La figure 11 montre le programme du Key stage 1. Il est composé de 2 années scolaires pour les enfants de 5 à 7 ans. Les trois premiers points portent sur les sciences de l'informatique. Les élèves devront en fin de key stage comprendre ce qu'est un algorithme et comment il est implanté en tant que programme. Ils devront de plus être capables de prédire les réponses d'un programme, et d'en créer. Les trois derniers points portent sur l'utilisation d'un ordinateur de manière efficace et responsable, comme par exemple, utiliser la mémoire d'un PC pour organiser ses données, et apprendre à protéger sa vie privée sur internet.

Key stage 2

Pupils should be taught to:

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration
- use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Figure 12 : Programme scolaire britannique - Partie Informatique, Key stage 2

La figure 12 montre le programme du Key stage 2. Il est composé de 4 années et concerne les enfants de 7 à 11 ans. Le programme du key stage 2 est dans la continuité du key stage 1. Les élèves doivent concevoir, écrire et débogger des programmes résolvant des problèmes, partitionner leur code en fonctions, et utiliser les structures répétitives et conditionnelles. Ils doivent de plus être capables de justifier pourquoi un algorithme fonctionne. Ils développent donc des compétences de raisonnement logique. Par ailleurs, ils doivent également apprendre à utiliser un ordinateur de manière sûre, respectueuse et responsable, en particulier sur internet.

Key stage 3

Pupils should be taught to:

- design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems
- understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem
- use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
- understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]
- understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems
- understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits
- undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
- understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct and know how to report concerns.

Figure 13 : Programme scolaire britannique - Partie Informatique, Key stage 3

La figure 13 montre le programme du Key stage 3. Celui-ci est composé de 3 années concernant les élèves de 11 à 14 ans. Les élèves ne doivent plus seulement coder un algorithme, ils doivent réfléchir à l'algorithme qu'ils implantent, en termes d'efficacité, d'optimisation, de mémoire. La notion de langage informatique apparaît, ce qui laisse supposer que jusqu'à présent, les élèves utilisaient d'autres moyens pour programmer.

Key stage 4

All pupils must have the opportunity to study aspects of information technology and computer science at sufficient depth to allow them to progress to higher levels of study or to a professional career.

All pupils should be taught to:

- develop their capability, creativity and knowledge in computer science, digital media and information technology
- develop and apply their analytic, problem-solving, design, and computational thinking skills
- understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to identify and report a range of concerns.

Figure 14 : Programme scolaire britannique - Partie Informatique, Key stage 4

La figure 14 montre le programme du Key stage 4. Celui-ci comprend 2 années pour des élèves de 14 à 16 ans. Ici, aucune référence à la programmation, mais il apparaît que les élèves doivent avoir l'opportunité « d'étudier les aspects des sciences de l'informatique à un niveau suffisamment élevé pour leur permettre de progresser dans les études supérieures ou dans leur carrières professionnelles. ». Pour cela, il faudra qu'ils puissent développer et de mettre en application leurs capacités d'analyse, de résolution de problème de conception, de développer leur capacité et créativité, et qu'ils comprennent comment leur sûreté et sécurité peut être affecté par un infime changement. Le programme à ce niveau est donc plus libre, pour permettre à chaque élève un enseignement plus personnalisé.

Les quatre Key stage décrits comprennent chacun une partie programmation et une partie d'utilisation efficace et sûre du matériel et d'internet. Le fait de combiner ces deux points semblent pertinent. Ce programme est destiné à des élèves de moins de 16 ans. Ils sont donc encore jeunes et pourtant très exposés au danger d'un internet mal utilisé. Il est donc important qu'ils puissent apprendre à s'en servir de manière sûre. De plus, l'apprentissage de la programmation, des algorithmes, des concepts fondamentaux des ordinateurs leurs serviront dans leur vie professionnelle, mais également dans leur vie d'élève, car ces notions serviront à d'autres matières, comme par exemple les mathématiques. Le point 1 du key stage 3 leur permettra par exemple d'analyser des résultats d'expérience de population, dans le cadre de sciences économiques.

Les programmes sont pertinents. La liberté laissée aux enseignants permet de partir de cas simples et intéressants pour les enfants du key stage 1 avec, par exemple, des robots à programmer. Ensuite, au key stage 2, on peut imaginer utiliser Scratch^{xxxv}. Au key stage 3, on introduira au moins 2 langages informatiques, dont du pseudo-langage. Et un approfondissement sera effectué au key stage 4.

I.3. Pays Baltes

L'Estonie lance en 2012 un programme un programme pilote d'enseignement des techniques de programmation destinés aux enfants de 7 à 17 ans, ce qui correspond à la scolarité obligatoire^{xxxvi}. Elle généralise ensuite l'expérience à l'ensemble des écoles du territoire. La fondation Tiger's Leap, fondé en 1996, avait pour projet le développement et l'expansion de l'informatique (matériels et infrastructure), avec un enjeu particulier pour l'éducation.

Cette initiative commença d'abord par équiper les écoles d'ordinateurs et leur permettre d'accéder à internet^{xxxvii}. Ensuite, les élèves ont appris la programmation par un aspect ludique : la création de leur jeu^{xxxviii}.

En 2004, Valentina Dagiene^{xxxix}, directrice du département de méthodologie informatique à l'université de Vilnius en Lituanie, organise le concours Castor Informatique. Destiné aux élèves lituaniens, le concours s'internationalise dès l'année suivante pour atteindre aujourd'hui 37 pays organisateurs en 2015.

A la création du concours en 2004, 3470 élèves lituaniens avaient participé au concours^{xl}. En 2014, ils étaient 24 985, soit environ 7 fois plus en 10 ans.

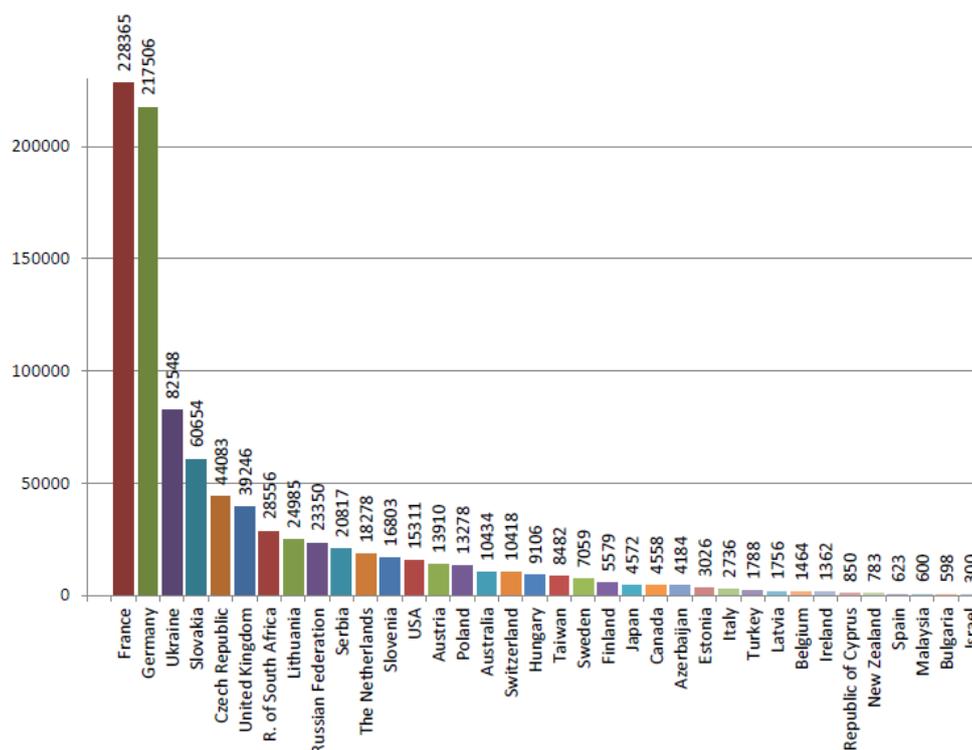


Figure 15 : Statistique du concours Castor Informatique en 2014

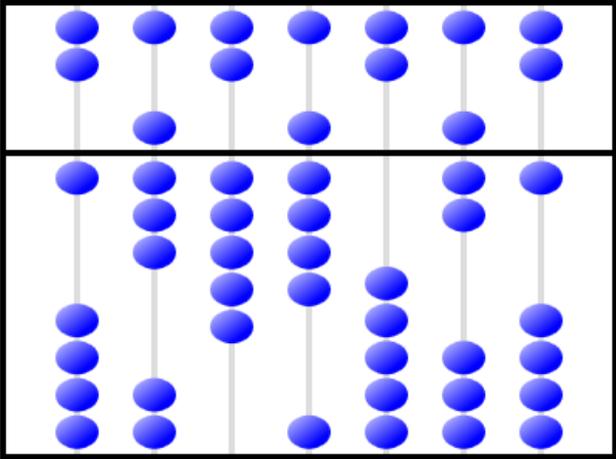
Si l'on prend l'exemple de la France, le premier concours Castor Informatique fut organisé en 2011 pour 46 346 élèves. En 2014 (cf figure 15), il y avait 228 335 participants français, et 340 000 en 2015^{xli}. L'accroissement du nombre de participants français est représentatif de l'engouement des élèves pour les sciences de l'informatique.

Le concours se présente sous la forme suivante : une fois par an, au début du mois de novembre, les élèves participants remplissent le questionnaire en salle informatique. Le temps pour chaque questionnaire est limité à 45 min, et plusieurs questionnaires sont disponibles en fonction du niveau des élèves. Par exemple en France, il y a 5 niveaux depuis cette année : CM1/CM2, 6°/5°, 4°/3°, Seconde, Première/Terminale^{xlii}. Pour chaque niveau, il y a 18 questions.

Positionnez les boules bleues de sorte à obtenir la valeur **1 859 071**.

Cliquez sur les boules pour comprendre ce qu'elles représentent.

Valeur courante : **1 859 071**.



Bravo ! vous avez réussi.

Votre score est maintenant : 8 sur 8.

Figure 16 : Question 1 du niveau Première/Terminale 2014

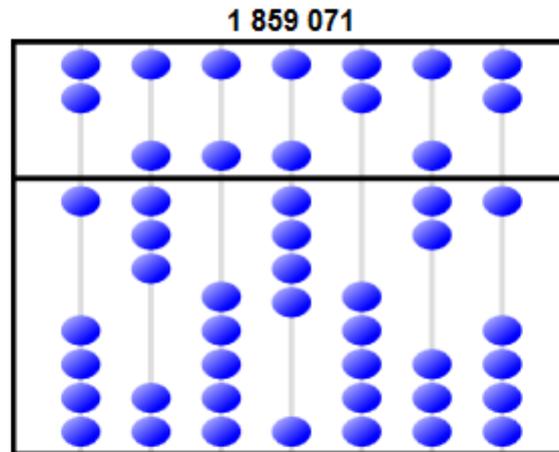
La figure 16 est un exemple de question posée au niveau Première/Terminale. A la fin du questionnaire, l'élève ne voit pas les réponses immédiatement.

Chapitre 2 : Analyse des pratiques

Dans la colonne de droite, chaque bille de la zone du bas qui est déplacée vers le milieu compte pour une unité, et chaque bille de la zone du haut qui est déplacée vers le milieu compte pour 5 unités. Ainsi, on peut obtenir n'importe quelle valeur entre 0 et 5 en n'utilisant que les billes du bas, et on peut obtenir n'importe quelle valeur entre 5 et 9 et utilisant une bille du haut et un certain nombre de billes du bas. On peut aussi aller jusqu'à 15 si on utilise toutes les billes, mais cela n'est pas utile.

Les billes de la seconde colonne en partant de la droite jouent un rôle similaire, sauf que leur valeur est multipliée par 10 : elles correspondent aux dizaines. Les billes de la troisième colonne ont une valeur multipliée par 100, et celles de la quatrième colonne ont une valeur multipliée par 1000, etc...

Ainsi, pour obtenir la solution, on pouvait positionner les billes de chaque colonne en fonction du chiffre correspondant dans la valeur à atteindre (le premier chiffre pour la première colonne, le second pour la seconde colonne, etc...).



C'est de l'informatique !

Le boulier (*abacus* en anglais) est l'un des plus anciens instruments d'aide au calcul et de mécanisation du calcul de l'histoire de l'humanité.

Lorsqu'on sait l'utiliser, le boulier permet d'effectuer des additions, soustractions, multiplications et divisions. Avec un certain entraînement, l'usage du boulier est aussi rapide que celui d'une calculatrice pour tous les calculs de la vie courante.

Le boulier est encore en usage aujourd'hui dans de nombreux pays d'Asie, preuve que l'informatique n'a pas (encore) tout remplacé !

Figure 17 : Correction de la question 1

Cependant, il a la possibilité de les voir une fois le concours terminé. Elles sont sous la forme de la figure 17. Un texte nous explique comment trouver la solution, la photo accompagnant le texte. Puis à la fin, un second texte nous expliquant en quoi ce problème est un problème informatique.

II. Pratiques en France

Les initiatives en France sont plus rares. Cela peut s'expliquer par le fait que son enseignement n'est pas encore prévu dans les programmes. Il le sera à la rentrée 2016. Nous pouvons cependant souligner les initiatives suivantes : les coding goûter, que l'on peut retrouver partout en France, et la ville d'Issy-les-Moulineaux, qui a mis en place des heures de code sur le temps périscolaire et qui a créé le GameCube. Nous étudierons donc ceci.

II.1. Les coding goûter

Présentation

Les coding goûter proposé par Jonathan Perret, fondateur de la société de conseil en informatique ut7 à Paris, ne sont pas des cours d'informatique^{xliii}. C'est-à-dire que les enfants découvrent par eux-mêmes, ils ne sont pas guidés par des professeurs. La formule proposée par Perret est un coding goûter de 3h, réunissant des participants d'âge variés. Le premier coding goûter était composé d'enfants de 6 à 14 ans, et d'adultes. Il y a donc plusieurs outils^{xliv} à disposition des participants en fonction de leurs envies, de leurs besoins. On retrouve par exemple, Scratch, GameSalad^{xlv}, RoboZZle^{xlvi}, etc.

Démarche

Le début de chaque coding goûter commence par un temps collectif à l'aide d'un vidéo projecteur avec RoboZZle. Le principe des coding goûter est basé sur l'entre-aide entre participants et non pas en un professeur qui fait un cours. Comme tous les participants sont novices, les échanges sont intéressants. Cette organisation est novatrice car elle met tous les participants au même niveau.

II.2. Issy-les-Moulineaux

La ville d'Issy-les-Moulineaux a mis en place plusieurs initiatives intéressantes. Elle propose des cours de programmation lors des temps périscolaire, et a créé en 2001 LeCube, premier centre de création entièrement dédié au numérique.

II.2.a. Périscolaire

Mis en place lors de la rentrée 2014, la ville d'Issy-les-Moulineaux propose des activités de programmation sur le temps périscolaire. Pour cela, elle met en place des activités à partir de Scratch sous le contrôle d'un intervenant en informatique^{xlvii}. Ces activités sont proposées aux enfants dès 6 ans. De plus, en partenariat avec la ville, Adelbaran^{xlviii}, société française de robotique, propose les mercredis après-midi des ateliers à la programmation robotique à ces mêmes enfants.

Dans le cadre des activités périscolaires, les enfants sont amenés à utiliser Scratch, pour apprendre les principes de la programmation : une instruction égale un effet, les boucles, les évènements, les conditions, etc.

II.2.b. LeCube

La ville d'Issy-les-Moulineaux crée en 2001 le centre de création numérique LeCube^{xlix}. Avant tout un lieu pour l'art numérique, le centre s'est diversifié pour intégrer la pédagogie numérique. C'est ainsi que pour l'année scolaire 2010-2011, le centre a mis en place le projet GamePlay dans 5 écoles de la Communauté d'agglomération Grand Paris Seine^l.

Le projet GamePlay propose aux classes de créer leur propre jeu, sur la base d'un serious game. La création d'un jeu passe par plusieurs étapes : choix du thème/du jeu, le scénario, les dessins (personnages, décors, etc), et la programmation du jeu. Ce projet est donc intéressant à mettre en place dans une classe, car il fait travailler plusieurs compétences. De la réflexion du jeu, au codage, en passant par l'écriture du scénario et les croquis nécessaires, toutes les parties du programme peuvent être abordées.

La partie code du jeu se fait à l'aide GameSalad, sur le même principe que Scratch à l'aide de bloc d'instruction à déplacer.

III. Bilan des pratiques

Grâce à l'analyse des différentes expériences dans le monde et en France, on peut remarquer deux approches :

- une approche privilégiant le code à l'aide de logiciel type Scratch sur PC ou tablette
- une approche sur matériel

Dans la plupart des initiatives, ces approches sont dissociées. Il semble pourtant intéressant de les associer, car elles se complètent.

Les élèves doivent pratiquer l'algorithmique. Pour cela, l'utilisation de logiciel type Scratch est pertinente.

Cependant, certaines notions de l'informatique, hors de l'algorithmique, mais nécessaire à l'informatique (comme la représentation des données ou leur stockage) ne peuvent être travaillé avec Scratch. Pour cela, du matériel déconnecté est adapté.

J'ai donc fait le choix d'associer ces deux pratiques pour fournir un enseignement complet.

Chapitre 3 : Propositions d'intervention

Il a été décidé de faire une intervention en classe à partir d'activités. Le choix de ces activités découle d'une réflexion basée sur les programmes 2016 du cycle 3. Il était indispensable de respecter le cadrage des programmes. L'introduction de l'algorithme apparaissant dans les programmes du cycle 3, cela m'a permis d'avoir à disposition un cadre. Les algorithmes apparaissent dans la partie Sciences et Technologies, dans le sous domaine Matière, Mouvement, Energie, Information :

Pratiquer des langages :

- P1 : exploiter un document constitué de divers supports (textes, schéma, croquis, tableau, algorithmes simples)
- P2 : utiliser différentes modes de représentations formalisées (schéma, dessin, croquis, tableau, graphique, texte)
- P3 : expliquer un phénomène à l'oral et à l'écrit

Initiation à la programmation :

- I1 : initiation faite à l'aide d'activité de repérage ou de déplacement (de robot ou de personnage sur écran), ou d'activité géométrique

Matériaux et objets techniques

- M1 : repérer et comprendre la communication et la gestion de l'informatique
- M2 : environnement numérique de travail
- M3 : le stockage des données, notions d'algorithmes (à l'aide d'applications visuelles et ludiques), les objets programmables (type robot)
- M4 : usage des moyens numériques dans un réseau
- M5 : usages de logiciels usuels

La notation effectuée sur les points précédents nous servira à faire des retours sur ceux-ci.

I. Les activités

L'analyse des programmes 2016 du cycle 3 et des différentes pratiques à l'étranger et en France met ainsi en évidence l'importance de logiciels ludiques et visuels pour l'introduction de l'algorithme. Concernant le stockage de données, il semble pertinent de mettre les élèves en situation, de travailler à l'aide de matériel plutôt que de faire un cours théorique, non pertinent à l'école. Nous avons donc décidé de faire deux types d'activités : des activités « déconnectées » et des activités « connectées ». En une intervention, nous ne pourrions couvrir tout le programme lié à ces apprentissages. Nous faisons donc le choix de nous focaliser sur les points suivants : P3, I1, M3. Nous couvrons ainsi les trois domaines, sans pour autant alourdir la séance.

Nous expliquerons ainsi le choix des activités et les décrirons.

I.1. Activités déconnectées

Nous avons décidé, avec mon directeur de mémoire M. Dadeau, de proposer des activités « déconnectées » à l'aide de matériels type flashcards. Cette utilisation de matériel m'est venue à l'idée lors des journées d'école avec mes maternelles. Lors de mes préparations de séquences pour ma classe, je privilégie beaucoup la manipulation. Il me paraît indispensable d'utiliser du matériel plutôt que de faire un cours théorique ou sur feuille. Cela est prouvé par les expériences internationales décrites précédemment et par la pédagogie Montessori^{li}. Nous avons choisies trois activités : une activité de représentation de nombres en binaire, une activité représentant le stockage de données et une activité de tri. Lors de nos discussions et recherches pour trouver des activités adaptés et cohérentes, nous avons découvert le site CSUnplugged^{lii} (Computer Science Unplugged signifie « Les sciences de l'informatique sans ordinateur »). CSUnplugged se décrit comme étant « une collection d'activités d'apprentissage gratuites qui enseignent les sciences de l'informatique à travers des jeux et des puzzles »^{liii}. Le site fourni un document^{liv} de 243 pages comprenant vingt-et-une activités classées dans six grands domaines de la programmation.

Toutes les activités sont présentées de la manière suivante :

- Un résumé de l'activité
- Les liens interdisciplinaires
- Les compétences
- L'âge requis des élèves
- Le matériel nécessaire

Ensuite, l'activité est décrite, et tout le matériel à imprimer est fourni.

Nous nous sommes donc inspirés de ce travail. Les activités de représentations de nombres en binaire et de représentation de données sont extraites de ce document. Nous avons choisi ces activités dans cet ordre-là car elles permettent de comprendre la représentation de données à l'aide de 0 et de 1.

I.1.a. Activité de représentation de nombres en binaire

Cette activité introduit la notation binaire, base de l'informatique. De plus, elle met en évidence que les données de l'ordinateur sont stockées et transmises sous la forme d'une série de 0 et de 1. A l'aide de cette activité, on montre que l'on peut ainsi représenter tous les nombres en utilisant seulement ces deux chiffres. Cette activité permet de répondre aux points P3 et M3 du programme.

De plus cette activité a de forts liens avec les mathématiques :

- Pour des petits nombres cibles, les élèves calculent mentalement
- Pour les grands nombres cibles, les élèves effectuent des calculs posés ou en ligne
- Les élèves continuent la séquence des nombres inscrits sur les cartes et trouvent seuls la règle qui la définit. Bien que les puissances de 2 ne sont pas étudiées au cycle 3, les élèves n'ont pas besoin de les connaître pour trouver comment les

nombre sont construits $2^0 = 1$; $2^1 = 2$; $2^2 = 4$; $2^3 = 8 \rightarrow 1 * 2 = 2$; $2 * 2 = 4$, $4 * 2 = 8$, etc.

Par groupe, les élèves reçoivent une carte avec une puissance de 2 inscrite dessus, et ils doivent, à l'aide de ces cartes, trouver comment faire par addition pour faire le nombre cible. Par exemple, le groupe est composé de 4 élèves. Ils reçoivent ainsi les cartes 1, 2, 4 et 8 correspondant à 2^0 , 2^1 , 2^2 et 2^3 . Le nombre cible est forcément inférieur ou égal à 15 car $2^0 + 2^1 + 2^2 + 2^3 = 1 + 2 + 4 + 8 = 15$. Avec le nombre cible 10, la solution est :

Tableau 3 : Résolution de la carte cible 10 avec les cartes des puissances de 2

8		2		=	10
---	--	---	--	---	----

Les cartes 2 et 8 seront face visible, et les cartes 1 et 4, face cachée (cf tableau 3). Pour privilégier l'autonomie, la solution sera disponible sans faire appel à un adulte.

L'activité se déroule en 2 temps (cf annexes 1 et 2):

- Un premier temps de découverte des cartes en faisant des calculs pour trouver le nombre cible, où la position des cartes n'est pas importante et où l'on utilise la valeur des puissances
- Un second temps où l'on enlève les cartes représentant les puissances et où les élèves n'ont que des 0 et 1. Ils doivent alors trouver le nombre cible en faisant attention à la position des cartes. La solution pour le nombre cible 10 serait ainsi :

Tableau 4 : Résolution de la carte cible 10 avec uniquement des 0 et des 1

1	0	1	0	=	10
---	---	---	---	---	----

On remarque que les cases contenant une carte 1 ne sont pas les mêmes lorsque nous utilisons les cartes 1, 2, 4 et 8 (cf tableau 4).

La figure 18 montre l'écriture des nombres en notation binaire.

$$\begin{array}{cccc}
 2^3 & 2^2 & 2^1 & 2^0 \\
 \downarrow & \downarrow & \downarrow & \downarrow \\
 8 & 4 & 2 & 1
 \end{array}$$

1 0 1 0 → 10

Figure 18 : Notation binaire

I.1.b. Activité de représentations de données

Cette activité met en évidence la façon dont sont représentées les images. Les enfants n'ont pas conscience qu'une photo n'est pas représentée telle qu'on la voit, mais à l'aide d'un code. Basée sur le mécanisme de l'impression, elle aide les enfants à en prendre conscience. Elle permet de travailler les points P3 et M3 des programmes.

Cette activité, basée sur le principe d'une imprimante, représente également le stockage de donnée. Une image est ainsi représentée pixel par pixel.

A partir d'une représentation en pixel art^{lv} de la lettre « a » en script, la représentation de données est mise en évidence (cf figure 19).

				1,3,1
				4,1
				1,4
				0,1,3,1
				0,1,3,1
				1,4

Figure 19 : Représentation du A script

L'information est représentée à l'aide de nombre, chaque nombre étant égale à un bloc de couleur alternativement blanche et noire. La première ligne est composée de 3 blocs de couleurs : 1^{er} bloc de taille 1 de couleur blanche, un 2^{ème} bloc de taille 3 de couleur noire, et un 3^{ème} bloc de taille 1 de couleur blanche. Cette ligne est donc codée « 1,3,1 ». La quatrième ligne semble commencer par une case noire. En fait, elle commence par un bloc de taille 0 de couleur blanche, puis un bloc de taille 1 de couleur noire, puis un bloc de taille 3 de couleur blanche et enfin, un bloc de taille 1 de couleur noire. Cette ligne est donc codée « 0,1,3,1 ». On remarque ainsi que le premier nombre représente toujours la taille d'un bloc de couleur blanche.

L'activité se présente en 3 étapes (cf annexes 1 et 3) :

- Présentation de deux lettres pour découvrir le principe. Les élèves construisent ainsi la règle ensemble à partir des exemples
- Décodage de lettres et dessins. Le fait de décoder des lettres et des dessins permet aux élèves de comprendre que le fonctionnement est le même, et que les lettres sont en fait des dessins, construits à l'aide de pixel blancs et noirs.
- Présentation d'un dessin en pixellisé en couleurs. Les élèves doivent, par discussion collective, imaginer un code pour représenter une image en couleur.

Cette activité peut être mise en lien avec les arts visuels et en Histoire des arts. Le pixel art est une forme d'art récente sous ce nom, mais qui est présente depuis l'Antiquité connue sous le nom de mosaïque.

I.1.c. Activité de tri

L'activité de tri est importante car les tris sont indispensables au bon fonctionnement des ordinateurs. Ils sont utilisés pour organiser des suites, et faciliter la recherche d'un élément dans un ensemble. On peut, par exemple, faire référence à l'ordre alphabétique, qui n'a vocation qu'à faciliter la recherche dans un dictionnaire^{lvi}.

Les enfants devront se trier en fonction de leur taille. Le fait de les faire se trier par taille met en évidence les comparaisons qu'ils devront effectuer.

A l'aide de 2 algorithmes, nous mettrons en évidence qu'un même problème peut se résoudre de plusieurs façons, et que le critère d'efficacité est à prendre en compte pour des professionnels.

L'activité se déroule en quatre temps (cf annexes 1 et 4) :

- Les élèves, alignés le long d'un mur, doivent se classer par ordre de taille sans autre consigne. Cette première étape permet aux élèves de se rendre compte qu'une tâche, apparemment simple, peut être source d'erreurs. De plus, les élèves se déplaceront tous en même, et cela augmentera le volume sonore.
- La deuxième étape, une fois les élèves revenus à leur place initiale avant tri, se fera à l'aide du questionnement suivant : est-on sûr que le premier élève est le plus petit ? Pour répondre à cette question, les élèves doivent dire qu'il faut comparer le premier élève avec tous les autres, un à la fois. Une fois ces comparaisons effectuées, nous serons sûr que l'élève le plus à gauche est bien le plus petit. Il faudra ainsi recommencer pour l'élève numéro 2, numéro 3, etc. pendant ces comparaisons et échanges, deux élèves notent le nombre de comparaisons et d'échanges effectués. Ce deuxième tri est le tri par sélection^{lvii}. Cet algorithme a été choisi car il est simple à mettre en place et à comprendre pour les élèves : comparer chaque élément avec tous les autres.
- La troisième étape, une fois les élèves revenus à leur place initiale avant tri, se fera à l'aide de la consigne suivante : comment faire pour « remonter » le plus grand en premier ? Les élèves devront alors proposer de comparer deux à deux chaque élément et de mettre à droite le plus grand. De même que pour le deuxième algorithme, deux élèves compteront le nombre de comparaisons et d'échanges. Ce troisième tri est le tri à bulles^{lviii}. Cet algorithme a été choisi pour les mêmes raisons que le premier. Dans celui-ci, les élèves peuvent comprendre aisément comment faire « remonter » le plus grand.

- La dernière étape se fera à partir des données recueillies par les élèves. Bien qu'ayant la même complexité¹, ces deux tris, pour l'exemple choisi, n'auront pas le même nombre d'échanges et de comparaison. Nous introduirons alors avec les élèves ce problème : effectuant le même travail, certains algorithmes prennent plus de temps. Il est donc important de réfléchir à l'algorithme choisi pour être optimal.

I.2. Activités connectées

Nous avons également décidé de proposer aux élèves une activité connectée sur tablettes/PC selon les moyens à l'aide de logiciel de type Scratch, Hour of Code, ... L'utilisation de tablettes/PC permet une approche ludique. Les élèves n'ayant probablement jamais pratiqué l'algorithmique, ils seront par ailleurs motivés par l'utilisation du numérique, peu utilisé en classe.

Nous avons décidé de prendre une application fournie par L'heure de code. Ces applications, à destinations des élèves, sont étudiées pour être accessibles par des enfants de l'école primaire. Chaque application se présente sous la forme d'une séance d'1h. Par souci d'organisation, nous avons prévu de ne laisser que 40 minutes aux élèves pour faire cette tâche. Chaque séance permet de découvrir les bases de l'algorithmique. Elles commencent par une vidéo explicative du thème, de l'univers, et du visuel. Les différents éléments sont présentés pour permettre une prise en main aisée. Les séances sont divisées en 10 à 25 étapes. Les notions abordées sont travaillées sur plusieurs étapes pour permettre une appropriation progressive par les élèves, et chaque étape est précédée d'une instruction, moins détaillée au fur et à mesure, qui donne un objectif pour valider l'étape. Il existe plusieurs séances, chacune reprenant des personnages et univers connus des enfants, tels que Minecraft, Elsa et Anna de La reine des neiges, Angry Birds, etc. Ces univers attirent les enfants.

Après réflexion, nous avons choisi de choisir l'application Lightbot^{lix}, proposée par l'Heure de code. Elle met en jeu un robot qui doit aller allumer des cases bleues sur un damier. Pour cela, le robot peut avancer, tourner à gauche, tourner à droite, sauter et allumer. A l'aide de ces cinq instructions et de vingt niveaux, l'application nous fait découvrir les bases de l'algorithmique comme l'appel de fonction ou les boucles sous forme d'appels récursifs. De plus, elle met en place une façon de programmer proche de celle des informaticiens : écriture du code, test puis debug si besoin.

L'application est ergonomique, et permet aux élèves de comprendre rapidement les pictogrammes utilisés. La consigne, accessible à tout moment depuis le bouton aide, nous explique les commandes et le but du niveau en cours. Les figures 20 et 21 mettent en évidence les explications fournies par l'application pour aider les élèves. Les mots clés (but de l'application : allumer les cases bleues ; explications des instructions) sont en bleu pour être soulignés.

¹ Le tri par insertion et le tri à bulle ont tous deux une complexité de $\Theta(n^2)$ dans le pire des cas. Cependant, on peut faire l'impasse sur cette notion de complexité (trop abstraite pour les élèves) et parler de nombre de comparaisons et d'échanges.



Figure 20 : Lightbot niveau 1-1 - But du niveau



Figure 21 : Lightbot niveau 1 - Explication des instructions

Chaque niveau est proposé sous la forme suivante :

- 1 : un *main* qui représente la partie principale du code où l'on met les instructions
- 2 : à partir du niveau 2, des procédures P1 et P2 pour faire des appels de fonction et de la récursivité

- 3 : les instructions disponibles pour le niveau en cours sous forme graphique. Cela permet à l'application d'être accessible à tous les élèves en se concentrant sur l'algorithmique plutôt que sur la lecture.
- 4 : une partie graphique pour visualiser les actions des instructions sur le robot
- 5 : des boutons de manipulation : exécution et affichage de la consigne

Les numéros permettent de voir sur la figure 22 l'emplacement des différentes parties.

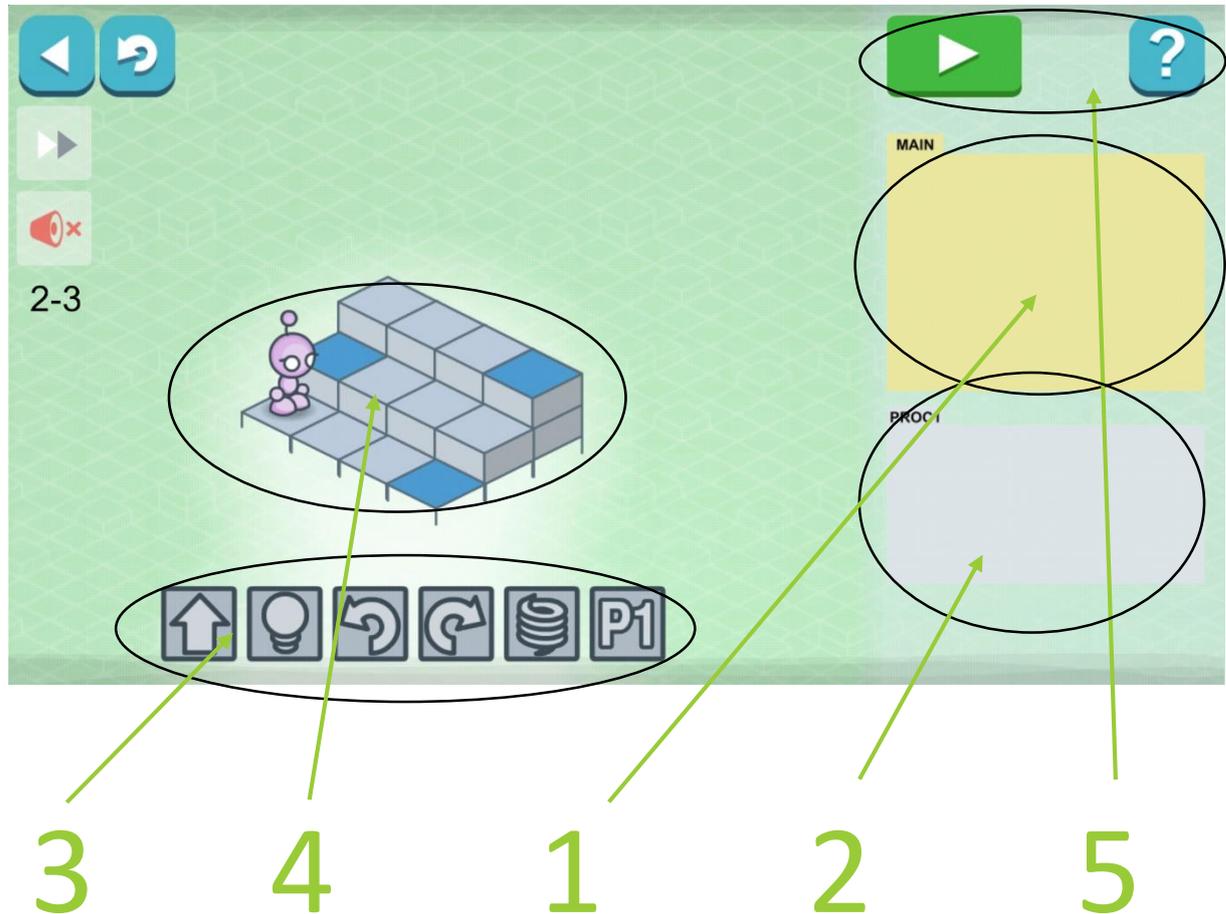


Figure 22 : Interface de Lightbot

Concernant la résolution des niveaux, l'application valide un niveau lorsque toutes les cases sont allumées, peu importe le nombre d'instructions. Cela est important car les élèves ne sont pas bloqués s'ils ne trouvent pas la solution parfaite, et deux solutions différentes mettent en évidence la possibilité d'améliorer sa solution en termes de nombres d'instruction. Cela est également mis en évidence sur la page d'accueil de l'application, où lorsque tous les niveaux ont été effectués, le score du joueur apparaît. Plus ce score est bas, plus les niveaux ont été résolus de façon efficace. Cela peut donc être une suite à une première séance : améliorer son score. Ce score correspond à l'addition du nombre d'instructions pour chaque niveau.

Enfin, une fois les trois niveaux terminés, le certificat de réussite, présenté précédemment, est attribué comme pour chaque application ayant le label Heure de code.

Le concepteur de l'application fourni une fiche reprenant des solutions « élégantes » (avec utilisations de *motif*) et des solutions simples (cf annexe 6).

II. Déroulement et Prévisions

Pour terminer la préparation des interventions, nous devons établir le déroulement de ces 2h. Ensuite, nous ferons des prévisions sur les résultats.

Nous avons prévu de faire une intervention de 2h. N'ayant qu'un nombre de tablette, il nous fallait faire travailler les élèves en groupe. La moitié de la classe serait sur tablettes, pendant que l'autre moitié travaillerait sur les activités déconnectées, et nous échangerions à mi-parcours. Le déroulement de ces 2h est présenté en figure 23.

	Groupe A	Groupe B
10 min	Présentation de l'intervention	
40 min	Activité de représentation des nombres binaires <i>en dirigé</i>	Activité connectée sur tablette <i>en autonomie</i>
	Activité de représentation des données <i>en dirigé</i>	
1 min	Echange	
40 min	Activité connectée sur tablette <i>en autonomie</i>	Activité de représentation des nombres binaires <i>en dirigé</i>
		Activité de représentation des données <i>en dirigé</i>
15 min	Activité de tris <i>en dirigé</i>	
15 min	Bilan de l'intervention	

Figure 23 : Planning de l'intervention

Faire travailler la classe en parallèle suppose que les élèves sur tablette sont autonomes. Cependant, l'activité sur tablette sera une découverte pour les élèves. Ils auront probablement besoin d'aide. M. Dadeau et le maître de chaque classe seront donc là pour les aider si besoin.

Chaque activité déconnectée est prévue pour durer 20 min (15 min pour l'activité de tri). Cet emploi du temps nous permet de faire 10 min de présentation de l'intervention, pour expliquer aux élèves la démarche de notre présence, et de faire 15 min de bilan pour voir ce que les élèves ont appris, ce qu'ils ont compris, et corriger ce qui a été mal compris.

Nous avons de faire travailler tous les élèves sur les tablettes avant de faire l'algorithme de tri pour qu'il y ait un lien. Les élèves auront déjà expérimenté les algorithmes sur la tablette avant de réfléchir à un algorithme de tri, plus complexe que ceux pour la tablette.

III. Les classes tests

Une fois les activités choisies, il a fallu choisir dans quelles classes nous allions intervenir. Il semblait intéressant de faire la même intervention dans trois classes de milieu différent. Il a donc été décidé d'intervenir dans une classe de centre-ville, une classe de quartier et une classe rurale. Ces trois classes représentent ainsi la diversité scolaire à laquelle nous faisons face.

Les classes choisies ont été :

- Une classe de CM2 à l'école Fourier. Située à Besançon dans le quartier Planoise, l'école est fréquentée par les élèves de Planoise, des Hautes-de-Chazal et de Chateaufarine. L'école est donc constituée d'élèves divers. Elle n'est cependant pas ni en REP ni en REP+.
- Une classe de CM1-CM2 à l'école Fanart. Située à Besançon dans le quartier Montrapon-Fontaine-Ecu, l'école est fréquentée par les élèves des rues alentour. Elle est située dans un quartier calme, proche du centre-ville.
- Une classe de CM1-CM2 à l'école des Terreaux. Située à Baume-les-Dames, c'est une école de centre-ville dans un milieu semi-rural.

Choisir trois classes de milieu social différent est important. L'objectif est de voir si l'informatique est une discipline qui creusera les inégalités sociales ou si, au contraire, les effacera.

La possibilité nous étant offerte, nous avons décidé d'inclure dans notre étude une classe spécialisée : l'ULIS de l'école des Terreaux à Baume-les-Dames. La classe comporte 11 élèves du CP au CM2. Cette intervention nécessita un travail avec la maîtresse de la classe pour adapter les activités.

Chapitre 4 : Expérimentations et résultats en CM1-CM2

Ce chapitre est consacré aux interventions dans les classes de CM2 et de CM1-CM2. Pour cela, nous décrirons chaque intervention, et analyserons les résultats. De ces descriptions et analyses, nous en tirerons un bilan.

I. Descriptions des interventions

Nous décrirons dans un premier temps l'intervention dans la classe de CM2 à Fourier, car ce fut la première. Ensuite, nous décrirons l'intervention dans la classe de CM1-CM2 à Fanart.

I.1. Classe de CM2 à Fourier

L'intervention s'est déroulée le jeudi matin 7 Avril. A notre arrivé, Joseph, le maitre de la classe, a proposé de faire des groupes en fonction des affinités. Les élèves se sont donc installés aux tables de groupes en fonction de cela dès leur arrivé. Joseph nous a prévenus que les élèves seraient perturbés par ces changements et qu'il faudrait donc les lancer rapidement dans les activités.

J'ai donc pris la décision de raccourcir la présentation de l'intervention à 5minutes. Les élèves étaient impatients de commencer.

Après avoir distribué les tablettes au premier groupe et expliquer le but et le premier niveau, j'ai laissé ces élèves en autonomie.

J'ai commencé par l'activité de représentation de nombres en binaire. Une fois les cartes distribuées, j'ai précisé aux élèves que l'ordinateur ne connaît que ces nombres-là, et je leur ai demandé pourquoi ceux-là. Ils ont de suite trouvé que la suite était construite à partir des doubles en partant de 1. Une fois ce point effectué, nous avons commencé les calculs, avec la carte cible 0. Ils ont eu des difficultés à proposer qu'on ne mette aucune carte pour faire 0. Ensuite, nous avons effectué la carte cible 3, avec la solution $2+1$.

Je leur ai ensuite distribué les cartes cibles, à prendre dans l'ordre. Je leur ai précisé que les solutions étaient derrière les cartes, et leur ai demandé de ne pas tricher pour ne pas fausser mes résultats de recherche.

Je leur ai également distribué une feuille (cf figure 24) avec comme aide des cases pour écrire leur résultats

Consigne : Tire une carte cible, résous-la, et inscris dans le tableau ton résultat

				→	
				→	
				→	
				→	

Figure 24 : extrait de la fiche pour la représentation de nombres en binaire

Les élèves ont bien travaillé, n'ont pas triché, et ont montré un entrain inattendu pour un jeu mathématique. Ils ont fait le rapprochement avec le jeu « le compte est bon ».

Je leur ai ensuite distribué les cartes suivantes de la suite, pour construire des nombres plus élevés.

Ensuite, nous avons changé d'activité en passant à la représentation de données. Les nombres sont une donnée particulière. Après avoir représenté un nombre, nous élargissons aux données en général.

Nous avons découvert la représentation du « a » en script et son code à côté. Ils ont donc trouvé par la discussion comment à partir du code, nous construisions le « a ». Ils ont ensuite « décodé » d'autres grilles individuellement (cf annexe 7). Chaque élève d'un groupe avait des grilles différentes. Cela leur permettait d'avoir toutes les lettres de l'alphabet et d'avoir plusieurs dessins. Ils ont des difficultés lorsqu'une ligne commençait par 0.

Pendant ce temps, les élèves sur tablettes sollicitaient leur maitre, M. Dadeau et moi. Je répondais donc à leur question. Le fait que les élèves me sollicitent également fait que je n'ai pas fait attention à l'heure. Nous n'avons donc pas eu le temps de faire l'activité de tri.

Au retour de la récréation, nous avons échangé les groupes. Mis à part une dispute dans un groupe, les élèves ont à nouveau bien travaillé.

Le bilan a été écourté par manque de temps. On peut se demander si les élèves ont fait les liens nécessaires entre les activités effectuées et le monde informatique. Joseph nous a proposé de prendre plus de temps dans les prochaines interventions pour améliorer ce retour sur activités, pour que les enfants puissent tous faire les liens.

Nous sommes donc restés toute la matinée au lieu des 2h initialement prévues.

I.2. Classe de CM1-CM2 à Fanart

Nous sommes intervenus dans la classe de CM1-CM2 à Fanart le mercredi matin 27 avril. Suite à l'intervention à Fourier, nous avons modifié notre planning:

- Nous avons décidé d'intervenir la matinée entière
- Nous avons décidé de supprimer l'activité de tri
- Nous avons décidé que je me concentrerai sur les activités déconnectées, M. Dadeau et Christophe, le maître de la classe, se chargeraient des questions des élèves sur tablettes.

J'ai pris 15 min pour présenter l'intervention, expliquer ma venue et faire une évaluation diagnostique de leur connaissance sur le sujet. Ensuite, je leur ai expliqué que nous allions apprendre à programmer à robot, et que nous allions apprendre à représenter les données comme dans un ordinateur.

J'ai ensuite agi de la même manière qu'à Fourier. Cependant, je me concentrais plus sur les activités déconnectées. Cela m'a permis de répondre plus efficacement aux demandes des élèves de ces activités-là, et de mieux gérer le temps. J'ai, de ce fait, proposer aux élèves une suite à chaque activité.

Pour l'activité de représentation des nombres binaires, les élèves de Fanart ont également fait le rapprochement avec un jeu mathématique : le mathador^{lx}. Nous avons décidé de supprimer la partie jusqu'à 1023. A la place, nous leur avons proposé de prendre des cartes 0 et 1 pour introduire la notation binaire.

Pour l'activité de représentation des données, je leur ai proposé une image en couleur pour laquelle ils devaient trouver une façon de la coder de la même manière que les images en noir et blanc proposé auparavant.

Les élèves ont appréciés ces deux activités supplémentaires.

II. Analyse des interventions

Les interventions effectuées dans les classes de Fourier et Fanart donnent les mêmes résultats. Les élèves des deux classes ont exprimé le même enthousiasme pour les trois activités, et se sont autant impliquées dans le travail.

Nous avons, de plus, observé les mêmes comportements chez les élèves :

- un fort attrait pour la tablette
- faire du lien avec des activités déjà pratiquées en classe pour les activités déconnectées (*mathador* et *le compte est bon* pour l'activité de représentation des nombres en binaire, et le pixel art et les mosaïques pour l'activité de représentation des données)
- une compétition entre élève : pour toutes les activités, les élèves avaient comme objectif d'en faire plus que les autres

- une envie de continuer les activités par une deuxième intervention ou en installant le jeu chez eux

Les résultats étant les mêmes, nous ne séparerons pas l'analyse de ceux-ci. Cependant, un aparté sera fait pour le jeu sur tablette, où une seule différence notable dans les résultats apparaît.

II.1. Activité déconnectée : écriture binaire

Cette activité est celle qui nécessitait le plus de prérequis. Il fallait que les élèves soient capables d'effectuer du calcul mental, bien qu'ils avaient droit de poser les calculs si besoin (cela fut peu utilisé). La mise en place de la procédure de résolution, cependant, se construisait au fur et à mesure des exemples.

Les élèves ont de suite compris comment la suite était construite, sans pour autant parler de puissance de 2. Ils étaient de plus capables trouver la suite, en effectuant du calcul mental.

Les prérequis étaient présents pour tous les groupes, et l'activité s'est déroulée avec peu de difficultés. Cependant, nous pouvons souligner les points suivants :

- Comment faire « 0 » ? La carte cible 0 était résolue collectivement. Les élèves avaient des difficultés à proposer la solution qui était de ne pas mettre de carte. Cela leur paraissant incorrect, car le contrat didactique, une fois les cartes des puissances de 2 distribuées, était, selon les élèves, qu'il fallait les utiliser. Or, les seules règles énoncées étaient : « *il ne faut utiliser que des additions et une carte ne peut pas être utilisée plus d'une fois* ».
- Stratégies de résolution : pour les cartes cibles basses, les élèves n'usent d'aucune stratégie, et effectuent les calculs rapidement mentalement. Chaque groupe avait 5, 6 ou 10 cartes. Les cartes cibles pouvaient ainsi aller respectivement jusqu'à 31, 63 ou 1023. Pour les cartes cibles inférieures à 15, les élèves avaient une bonne représentation mentale et trouvaient les réponses sans se tromper. A partir de 16, ils ont dû mettre une place une résolution ordonnée. Ils ont tenté en commençant du plus petit au plus grand mais cela ne fonctionnait pas. Ils ont donc commencé par le plus grand directement inférieur, et en additionnant les cartes de plus en plus petites. C'est en effet la stratégie de résolution utilisée dans le monde informatique pour convertir en base 2 un nombre en base 10.
- Cartes cibles jusqu'à 1023 : les élèves de la classe de Fourier ont eu deux séries de cartes cibles. La première séries allaient jusqu'à 31 et la seconde jusqu'à 1023. Les élèves ayant construit la stratégie de résolution dans la première série, cette série semblait redondante. Cependant, ils ne se sont pas démotivés, malgré la longueur des calculs parfois. On peut néanmoins noter que cette deuxième série de calculs n'étaient pas nécessaire pour la construction du savoir, et a ainsi été supprimé de l'intervention à Fanart.
- Cartes 0 et 1 : les élèves de la classe Fanart n'ont pas eu deux séries de cartes cibles. Ayant déjà effectué l'intervention à Fourier, nous avons retravaillé cette activité

pour qu'elle soit plus pertinente. Nous leur avons donc proposé après avoir fait la première série, dans les mêmes conditions qu'à Fourier, puis de faire les cartes cibles jusqu'à 63 avec les cartes 0 et 1, pour introduire la notation binaire. Ils avaient besoin de garder les premières cartes (celles de puissances de 2) et plaçaient en dessous les cartes 0 et 1 (cf figure 25).

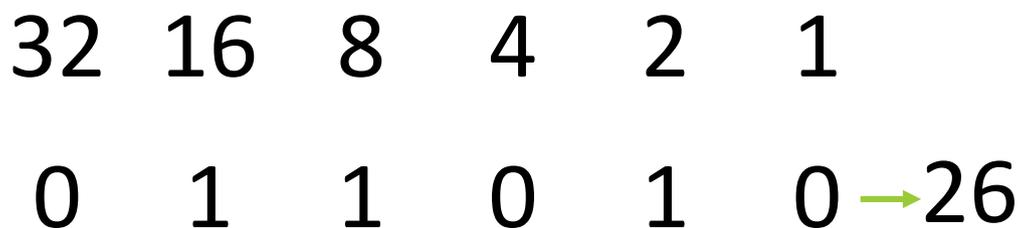


Figure 25 : Stratégie de résolution lors de l'utilisation des cartes 0 et 1

Les élèves n'étaient pas capables de reconstruire la suite (32,16,8,4,2,1) au moment des calculs. Leur concentration étant sur le calcul, il fallait leur faciliter la tâche en leur donnant visiblement les nombres de puissance de 2. Pour cela, j'avais préparé à nouveau des fiches avec les cases représentant les nombres binaires, marqués en haut de colonnes (cf figure 26).

16	8	4	2	1	→	Résultat
					→	
					→	
					→	
					→	

Figure 26 : extrait de la fiche pour l'activité binaire

Conclusion

L'activité n'a présenté aucune difficulté, car ils étaient habitués à la résolution de ce genre de problèmes (*mathador* et *le compte est bon*) et car ils ont fait preuve d'un entrain agréable et inattendu.

Le travail en groupe fut parfois compliqué. La formation des groupes est importante et doit être réfléchi en amont par le maître de la classe. Sur les deux classes, il n'y eut qu'un groupe qui posa soucis, suite à une dispute pendant la récré. Les autres groupes ont su travailler ensemble. Enfin, les groupes ne doivent pas être homogènes en termes de niveaux. Il est plus intéressant de travailler en groupes hétérogènes. Les élèves « forts » pourront aider

les élèves « faibles » dans un groupe hétérogène, contrairement à un groupe homogène d'élèves « faibles » qui pourraient rapidement se trouver bloqués et ainsi abandonner. Il faut cependant veiller à un vrai travail de groupe, et pas seulement à un travail d'élèves « forts ».

II.2. Activité déconnectée : représentation des données

Les élèves sont entrés rapidement dans l'activité une fois la procédure décrite : une suite de nombre de carreaux alternant blancs et noirs. Cependant, lorsque la suite de nombre commençait par un 0, les élèves étaient perturbés. Reprenons l'exemple en figure 19 du A en script et ajoutons le N en script.

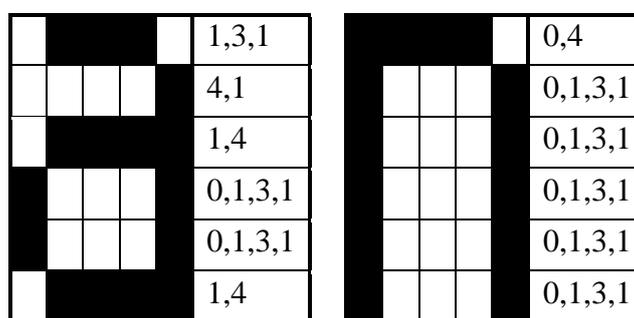


Figure 27 : exemple du A et du N en script

La figure 27 nous montre l'exemple correct pour deux lettres, ayant des 0 en début de ligne. La quatrième ligne commence par un 0. Cela signifie que nous laissons 0 cases blanches, ensuite nous mettons 1 case noire, 3 cases blanches et enfin 1 case noire. Ils interprétaient la ligne « 0,1,3,1 » comme « 0 cases blanches donc 1 cases noire, 1 case blanche, 3 cases noires, 1 case blanche ». Ils ont bien compris qu'on alternait les blancs et les noirs. Cependant, ce 0 en début de ligne leur posait problème, et l'interprétait comme « 1 case noire », et ainsi interprétait la suite de la ligne en reprenant l'alternance avec du blanc.

Le A et le N en script donnait donc comme sur la figure 28.



Figure 28 : A et N en script faux

Ils se rendaient vite compte qu'il y avait une erreur, car ils ne pouvaient reconnaître la lettre. Nous remarquons également que le dernier nombre de chaque suite correspond toujours

à un nombre de cases noires. Cela est également un indice qui indique qu'il y a une erreur. Il fallait donc reprendre avec eux l'exemple, et analyser à nouveau. Lors de la présentation de l'activité avec une lettre en format A4, pour travailler collectivement dessus, certains élèves n'osent pas prendre la parole pour dire qu'ils n'ont pas compris. Ils préfèrent attendre un moment individuel pour faire cela, ou se disent qu'ils comprendront seuls. C'est probablement pour cela que certains ont compris du premier coup mais que d'autres ont osé appeler en disant « je n'ai pas compris pour le 0 » lors du travail individuel.

Pour les élèves de Fanart, nous avons ajouté le travail de représentation de données en couleur à l'aide des deux pixel art (cf annexe 5). Ils devaient proposer des solutions pour coder les images selon le principe découvert en noir et blanc.

Ils ont proposé plusieurs deux solutions contenant chacune un code couleur que l'on respecte :

- Solution 1 : Une suite de nombre tel que : couleur, nombre de cases de cette couleur, couleur, nombre de cases de cette couleur, etc
- Solution 2 : Une suite de nombre avec n fois le nombre de cases de chaque couleur

Prenons par exemple le code couleur suivant :

Blanc = 1 ; Noir = 2 ; Rouge = 3 ; Bleu = 4 ; Jaune = 5

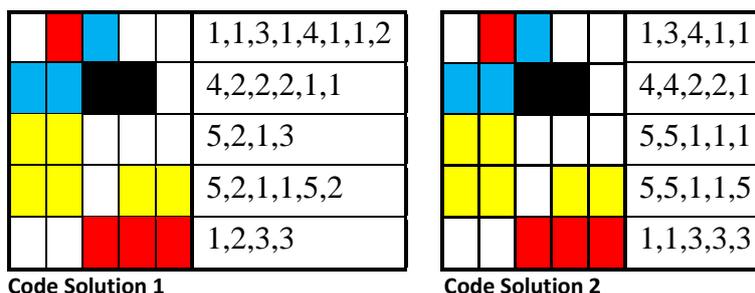


Figure 29 :

Code Solution 1

Code Solution 2

Leurs deux solutions sont correctes (cf figure 29).

La première solution prend exemple sur la démarche présentée en noir et blanc. Il est naturel que les élèves y aient pensé.

La seconde, en revanche, a été proposée car plus évidente, plus naturelle pour les élèves. A chaque case correspond une couleur, plutôt que de coder le nombre de cases pour chaque couleur. Nous pouvons noter que cette façon de faire est la technique utilisée pour compresser une image.

Conclusion

Les élèves ont vite compris le principe. Le 0 placé en début de ligne a résisté à certains élèves, mais une remédiation individuelle ou entre pairs à régler ce problème. Ils ont pu construire le code nécessaire pour représenter les images en couleurs, ce qui prouve qu'ils ont compris, et qu'ils peuvent ainsi transposer ce qu'ils ont appris en noir et blanc sur de la couleur.

II.3. Activité connectée sur tablette

Nous pouvons noter d'abord que les élèves se sont naturellement aidé entre eux, et ont développé un esprit de « compétition », bénéfique car il leur permettrait de se dépasser.

De plus, bien que n'ayant pas eu le temps de terminer tous les niveaux, ils ont souhaité que nous revenions pour poursuivre, et ont demandé le nom de l'application pour poursuivre chez eux.

Enfin, lors du bilan en fin de matinée, les élèves ont dit avoir joué. Ils n'ont pas eu l'impression de travailler, mais sont tout de même conscients d'avoir appris les bases de la programmation.

Partie 1 : les bases

Les trois premiers niveaux permettent aux élèves de prendre en main l'application en présentant les instructions et en proposant des niveaux simples à résoudre. Le premier niveau présente deux instructions : « avancer » et « allumer », le second niveau introduit les instructions « tourner à gauche » et « tourner à droite », et enfin le troisième niveau présente l'instruction « sauter ». Ce sont les cinq instructions de base.

En analysant les parties de chaque élève, on remarque que le niveau 1-1 (cf. figure 30) a été fait de façon optimal par tous.



Figure 30 : Lightbot niveau 1-1

Les 8 premiers niveaux ont été faits par 100% des élèves. Ils amènent les élèves à prendre en main l'application, à l'aide de niveaux simples à résoudre.

Cependant, nous remarquons que dès le niveau 1, les élèves, bien qu'ayant la solution optimale, sont peu à la mettre en place. En effet, à plusieurs reprises, ils sont beaucoup à insérer dans leur solution des instructions inutiles, telles que sauter sur place (le robot ne peut

sauter que pour monter ou descendre une marche), ou un avancer sur place (contre un mur ou dans le vide). Cette erreur est révélatrice de leur stratégie de réalisation : pour la réalisation des niveaux, ils devraient « se dérouler » dans la tête les mouvements du robot pour réussir, et une fois cela fait, écrire ces instructions dans l'application. Cependant, soit ils ne l'ont pas tous fait, soit ils l'ont fait mais ne peuvent se souvenir de toutes les actions. Dans les deux cas, ils insèrent des instructions inutiles et ne vérifient pas. L'application valide les niveaux une fois que toutes les cases bleues sont allumées, peu importe le nombre d'instruction mises en place. Pour leur faire prendre conscience de cela, il faudrait une deuxième séance avec comme objectif d'améliorer leur solution. Pour améliorer leur solution, il y a deux façons de faire :

- Repérer les instructions inutiles et les supprimer. Pour cela, ils doivent observer chaque niveau, et vérifier les actions du robot
- Optimiser la solution. Pour cela, une discussion collective sur les solutions possibles peut être envisagée.

L'intérêt de commencer par des niveaux simples est de permettre aux élèves de comprendre le fonctionnement et le but de l'application. Les niveaux simples ne les mettent pas en difficulté, et ont ainsi envie de continuer. Cependant, trois petites difficultés apparaissent dans la partie 1 :

- Les marches → le robot saute pour monter les marches. Les élèves ont de suite compris cette instruction. Cependant, naturellement, ils voulaient faire descendre le robot en le faisant avancer d'une case, alors qu'il faut le faire sauter pour cela. Néanmoins, une fois confronté à ce problème, ils trouvent rapidement la solution, et ne la refont pas.
- Droite/Gauche → bien que les élèves différencient tous leur droite de leur gauche, être capable de se décentrer pour trouver la droite (ou la gauche) du robot est plus compliqué. Lorsqu'ils devaient faire tourner le robot, les élèves réfléchissaient plus et certains faisaient le geste avec leur main pour savoir où tourner.
- Tourner sur place → certains pensaient que l'instruction « tourner à gauche » faisait tourner le robot à gauche sur la case suivante, alors qu'elle le fait tourner sur la même case. Cependant, les élèves ont vite compris.

Partie 2 : les procédures

Les procédures sont expliquées dans les deux premiers niveaux comme étant un espace supplémentaire pour écrire les instructions. Il y a donc deux solutions possibles pour ces niveaux :

- Soit nous utilisons P1 comme un espace supplémentaire (figure 31)
- Soit nous utilisons P1 comme une procédure, dans laquelle nous écrivons un motif qui sera appelé autant de fois que nécessaire (figure 32)

Ces deux solutions étaient possibles avec les trois premiers niveaux. Pour les trois derniers, il fallait absolument trouver le motif répété.

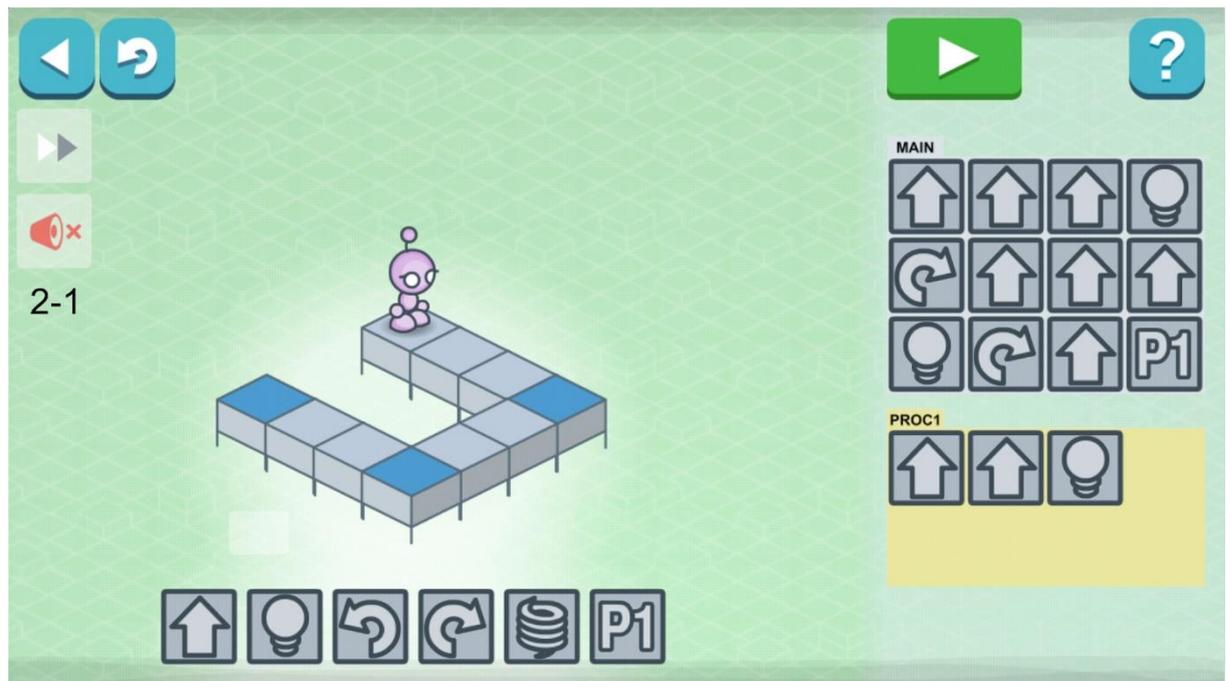


Figure 31 : Lightbot 2-1 solution non optimale

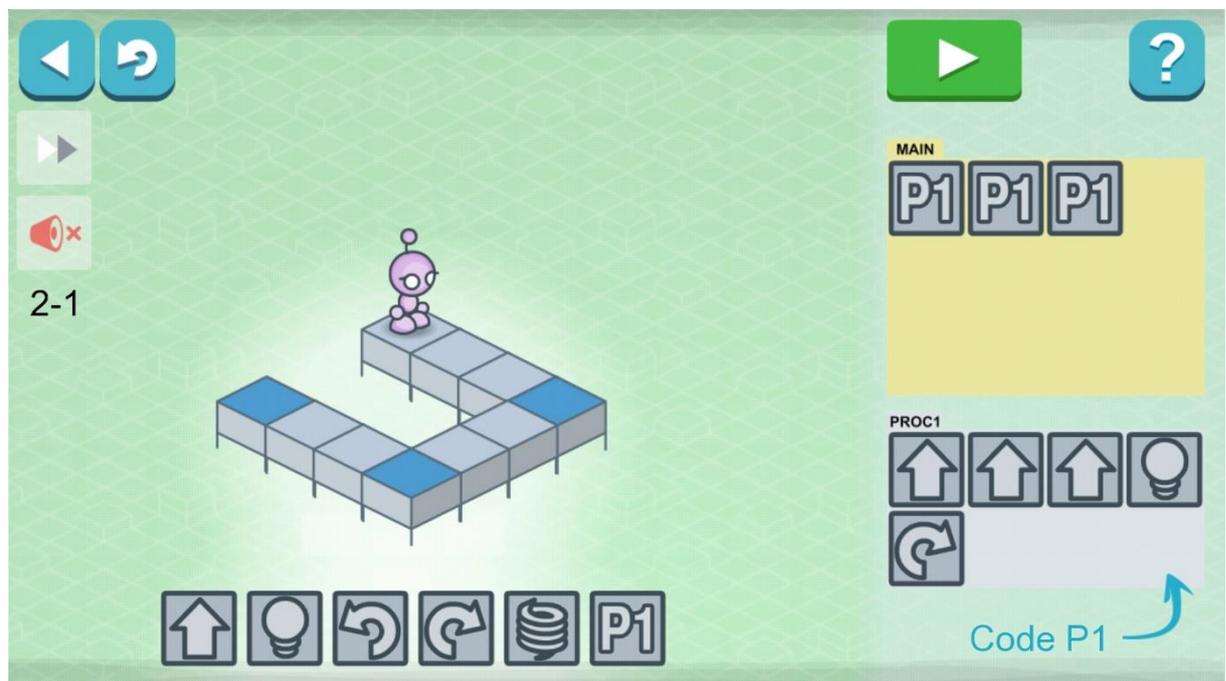


Figure 32 : Lightbot 2-1 solution optimale

Différences notables

Tableau 5 : pourcentage d'utilisation de P1 comme une procédure (et non comme un espace supplémentaire)

	Ecole Fourier	Ecole Fanart
Niveau 1	50%	0%
Niveau 2	75%	0%
Niveau 3	50%	12%
Niveau 4	100%	100%

Le tableau 5 montre le pourcentage d'élèves qui ont utilisé P1 comme une procédure. Nous pouvons noter que les élèves de Fanart n'ont pas pensé d'eux-mêmes que P1 pouvait être appelé à plusieurs reprises dans le *main*, alors que 50% des élèves de Fourier y ont pensé (ou ont été plus aiguillés à cette utilisation de P1). Nous remarquons également que les élèves de Fanart ne sont que 12% à utiliser P1 comme une procédure dans le niveau 3, alors que la consigne indique clairement que P1 est utile avec des *motifs*², une suite d'instruction réutilisable plusieurs fois, à conditions de l'appeler autant de fois que nécessaire dans le *main*. On peut donc en conclure que la plupart des élèves n'ont pas lu la consigne, l'ont mal comprise ou pas comprise, ou n'en ont pas tenu rigueur car le niveau était réalisable sans utilisation de motif. Enfin, il n'est pas évident de comprendre ce que l'on appelle retour de procédure³. Les élèves utilisant l'appel de procédure à P1 en fin de *main* dans les niveaux, il leur a été difficile de constater ce retour de procédure. Enfin, le lien *main* – procédure a été délicat. Les espaces du *main* et de P1 étant à la suite sur l'application, ils ont naturellement pensé qu'une fois toutes les instructions du *main* exécutées, celles de P1 s'exécuteraient. Pour que cela se produise, il faut préciser au *main* d'appeler P1.

Le niveau 4 n'était réalisable qu'en utilisant P1 et P2 comme des procédures contenant un motif. Certains ont dans un premier essayé de l'utiliser comme un espace d'écriture supplémentaire et se retrouvaient bloqués comme le montre la figure 33.

² En informatique, un motif (couramment appelé *patterns*) est une structure que l'on peut observer de façon répétée.

³ En informatique, le retour de procédure est le fait qu'à la fin de la procédure, le point d'exécution, une fois la procédure effectuée, « retourne » dans le *main* à la suite de l'appel de procédure. L'exécution du programme se continue donc avec les instructions à la suite de cet appel (s'il y en a).

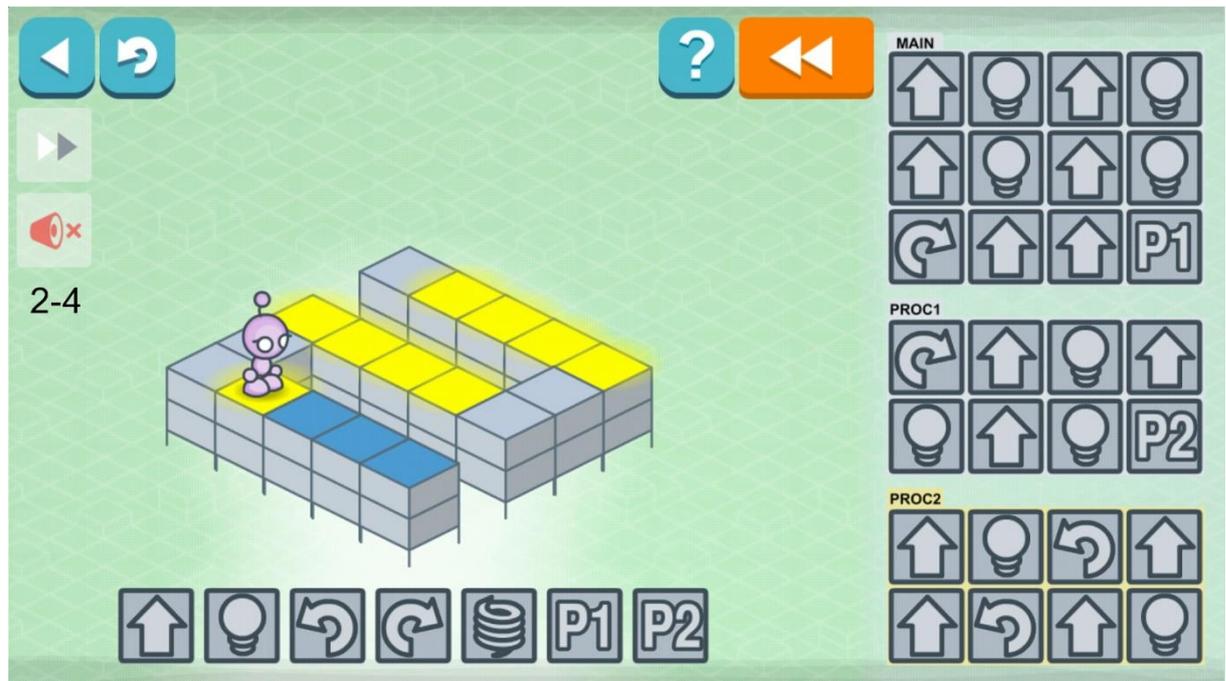


Figure 33 : Lightbot 2-4 robot bloqué

Il fallait donc les aider en leur précisant de trouver un motif répété. Les élèves remarquaient donc le motif suivant « avancer, allumer, avancer, allumer, avancer, allumer, avancer, allumer » qui se répétait trois fois. La solution était donc celle proposée en figure 34.

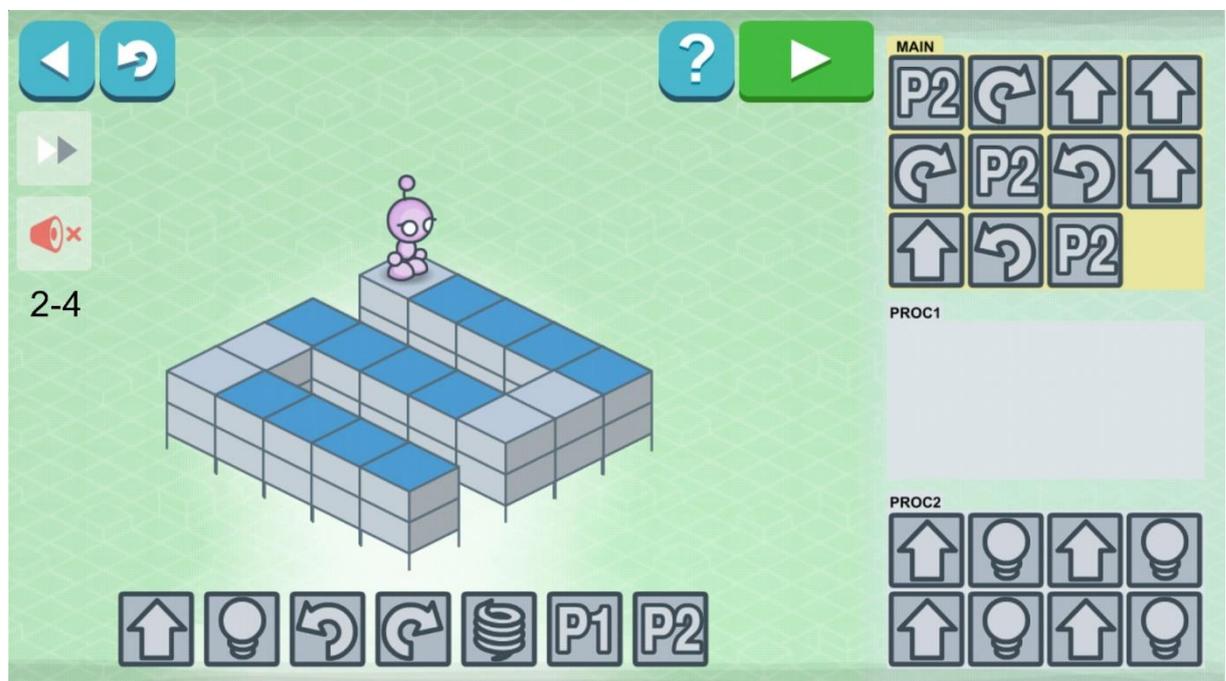


Figure 34 : Lightbot 2-4 solution avec motif non optimale

Cependant, dans le motif choisi par les élèves, on remarque que le motif « avancer, allumer » est lui-même répété quatre fois. La solution serait donc celle de la figure 35.

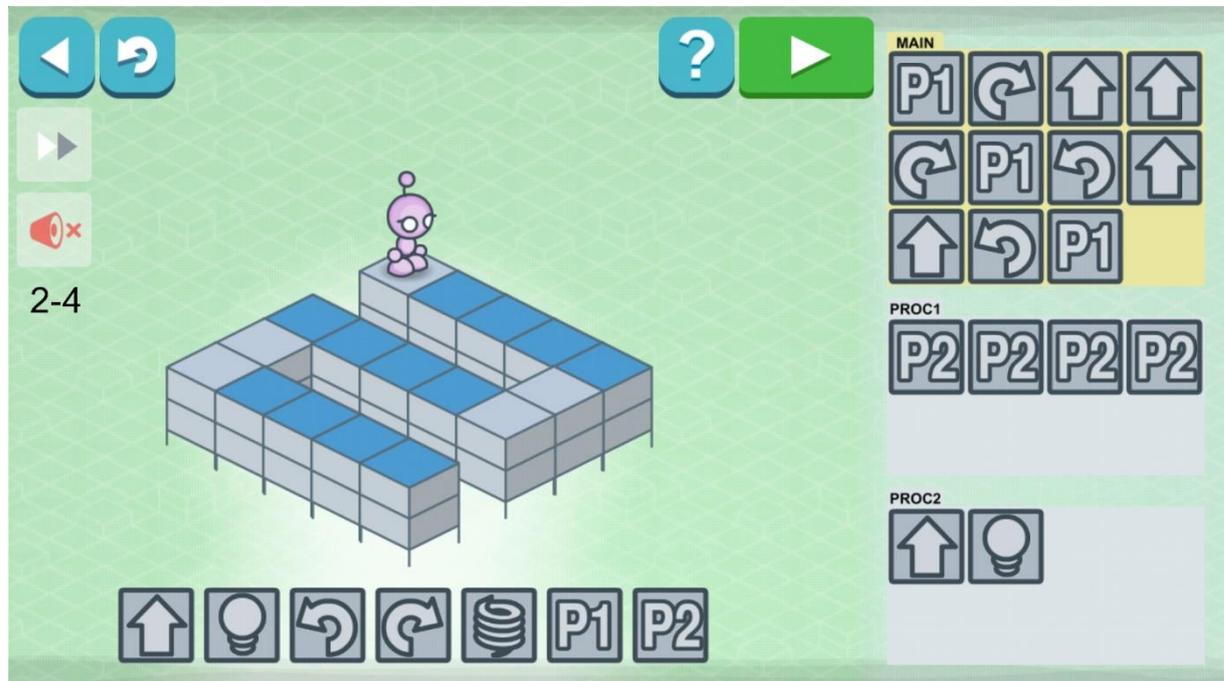


Figure 35 : Lightbot 2-4 solution optimale

On remarque ainsi que les élèves ont utilisé un motif plus évident au premier abord : « avancer, allumer, avancer, allumer, avancer, allumer, avancer, allumer ». Ceci pourrait être amélioré facilement si l'on faisait une deuxième séance, à l'aide par un exemple d'un tableau numérique pour mettre en avant ce motif « avancer, allumer ». Nous pourrions également leur proposer d'améliorer leur score sous forme de challenge, en leur précisant l'objectif à atteindre pour chaque niveau.

Le niveau 5 a posé problème aux élèves. Nous leur avons donc proposé de passer à la partie 3, puis de revenir sur ce niveau à la fin s'il leur restait du temps. Seul quelques élèves l'ont de ce fait essayé, et étaient proches de la solution. Mais aucun n'a réussi.

Les procédures sont importantes dans la programmation. Lorsqu'un programme commence à être un peu complexe, il est probable que l'on doive réutiliser des « morceaux » de programmes. Il faut donc déterminer ces « morceaux » que l'on appelle motifs (patterns). Dans l'application, cela est fait progressivement, en commençant par prétexter un manque d'espace, ce qui pourrait se rapporter à l'optimisation d'un programme en termes de lignes de programme. Ensuite, cela est amené en précisant que des motifs sont répétés, et qu'on peut faire appel à eux plusieurs fois plutôt que de les écrire le nombre de fois nécessaire. Chaque niveau étant réalisables de plusieurs façons, nous pourrions amener les élèves à faire la remarque suivante : un problème peut être résolu de plusieurs façons (par plusieurs algorithmes) mais la recherche d'optimisation est un élément à prendre en compte.

Partie 3 : les boucles

Les trois premiers niveaux restreignent les élèves en nombres d'instructions possibles, de telle sorte qu'ils comprennent le principe d'une boucle. Tous les élèves qui ont eu le temps d'arriver à ces niveaux-là ont, de ce fait, tous trouvé la solution optimale.



Figure 36 : Lightbot 3-1 consigne

La figure 36 montre la consigne du niveau 1. La consigne explique qu'en appelant P1 dans P1, le robot exécute P1 à l'infini.

Les deux derniers niveaux ont été moins effectués par les élèves, car la séance se terminait.

Le fait qu'ils soient restreints leur permet de mieux comprendre, et cela se ressent sur les niveaux qui suivent. Il semble que les élèves ont mieux compris et mieux réussi les boucles que les procédures. Pourtant, les boucles sont ici vues comme des procédures répétées de façon récursive.

Conclusion

Les élèves ont pris rapidement l'application en main. Ils ont, de plus, rapidement compris les instructions de bases.

Il est important de souligner les points suivants :

- Les élèves ont eu l'impression de jouer. Ils ont donc pu être concentrés sur la même activité un temps relativement long.
- Ils ont tendance à faire les niveaux sans lire les consignes. Ils font donc souvent appel à l'adulte lorsqu'ils sont bloqués. Pour remédier à cela, il faut instaurer quelques règles :

- Relire la consigne avant d'appeler un adulte
- Demander à un voisin si relire la consigne n'est pas suffisant
- Appeler l'adulte seulement si les deux premières règles ne sont pas suffisantes

Ces trois éléments impliquent une disposition en îlot pour favoriser les échanges entre élèves.

- Ils ne cherchent pas à améliorer leur score. L'activité étant présentée sous forme de jeu, ils veulent dans un premier temps réussir tous les niveaux. Une deuxième séance serait donc nécessaire pour mettre en évidence l'importance d'utiliser le moins d'instruction possible, et donc d'améliorer son « score », par la résolution collective sur vidéoprojecteur ou par le côté challenge.
- Les procédures sont assez mal comprises par la majorité des élèves. Pour cela, un accompagnement plus poussé serait pertinent. Avant de commencer la résolution des niveaux de la partie procédures, nous pourrions résoudre le niveau 1 collectivement en cherchant le *motif* répété, et en indiquant aux élèves que ce motif ne doit apparaître qu'une fois dans le code. Les programmes préconisent de faire travailler les élèves à partir de problèmes. Ceci est donc un bon exemple pour faire chercher les élèves ensemble.

III. Bilan

Nous pouvons maintenant tirer le bilan des interventions en termes de réussites et de difficultés.

III.1. Réussites

Les élèves ont été très motivés par les activités. L'activité sur tablette leur a énormément plu, ils ont eu l'impression de jouer, ce qui était le but. De plus, utiliser ce genre de matériel n'est pas courant dans une classe lambda, ils en ont donc pris soin comme demandé et se sont de suite mis au travail. Les activités déconnectées étaient ludiques et leur ont plu également. Avec ces activités, les élèves ont été attentifs et ont travaillé toute la matinée sur le même objectif, sans débordements.

III.2. Difficultés

III.2.a. Le matériel

Tablettes vs PC

Nous avons décidé de proposer l'activité connectée sur tablettes. Les écoles de Besançon sont équipées de trois PC par classe, et tout non pas de salle informatique. Il fallait donc prévoir assez de matériel pour que les élèves soit au maximum 2 par machine, l'idéal étant

qu'il soit seul. L'utilisation des PC n'était donc pas possible. Nous avons donc effectué une demande de prêt auprès de Canopé qui met à disposition des écoles des tablettes dans le cadre d'un projet. Cette demande ayant été refusé, il nous a fallu emprunter des tablettes à l'université. Cependant, cela n'aurait pas été possible en dehors du cadre du mémoire. C'est donc un point important à prendre lorsque l'on décide de faire une séquence sur l'algorithmique.

L'application

Pendant un temps, nous avons choisi l'application Run Marco^{lxi} sur tablette. Cependant, quelques jours avant la première intervention, une mise à jour fut effectuée par les développeurs, et elle ne fonctionna plus sur nos tablettes. C'est un point non négligeable à prendre en compte, et ainsi désactiver les mises à jour automatiques et toujours avoir une solution de repli.

III.2.b. Gestion des élèves et du temps

Les élèves

L'activité sur tablette est une activité autonome. Les élèves sont censés pouvoir avancer dans les niveaux sans sollicitation. Cependant, les élèves ne lisaient pas les consignes et lorsqu'ils arrivaient sur un problème résistant, ils appelaient de suite à l'aide. La non lecture des consignes et aides est représentatif de la vie de classe. Si les consignes ne sont pas lues pas avec les élèves, ils ne le feront pas seuls. Il faut donc insister sur le fait que les consignes sont là pour les aider, et que s'ils sont en difficulté, ils doivent les relire avant d'appeler l'adulte.

Le temps

Les sollicitations non prévues des élèves sur tablettes nous ont fait dépasser le temps prévu pour la première intervention. L'intervention était prévue pour durer 2h, elle a duré 2h30, toute la matinée. De plus, les activités prévues étaient trop longues pour ne durer que 15 minutes. Pour les interventions qui ont suivies, elles ont donc été retravaillées. Par ailleurs, l'activité de tri n'a pas pu être effectuée.

L'encadrement

Nous étions, chaque intervention trois adultes : le maître de classe, M. Dadeau et moi-même. Nous n'étions pas de trop pour gérer les élèves. L'intervention telle quelle serait difficilement transposable dans une classe sans intervenants. Il faudrait plus de tablettes pour permettre à toute la classe de travailler sur la même activité en même temps et un vidéoprojecteur pour faire les trois activités en même temps que les élèves dans un premier temps. Il faudrait ainsi au minimum deux séances.

Chapitre 5 : Intervention en ULIS

L'intervention telle qu'elle était prévue pour les classes de CM1-CM2 devait être adaptée à la classe d'ULIS.

Nous allons donc présenter les adaptations effectuées pour les activités dans un premier temps, puis nous décrirons et analyserons l'intervention. Enfin, nous en tirerons un bilan.

I. Adaptations sur les activités

Il est nécessaire d'adapter l'intervention. Pour cela, j'ai rencontré en amont Audrey, la maîtresse d'ULIS, pour discuter de la classe et des possibilités de ses élèves. A partir de cela, nous avons adapté les activités. Nous avons ainsi décidé, avec Audrey, d'enlever l'activité sur l'écriture binaire. La classe étant composée d'élèves de niveau CP au CM, et ayant de fortes différences de niveaux entre les élèves, il n'était pas pertinent de garder cette activité. Nous avons donc gardé l'activité sur tablette et l'activité de représentation de données en procédant cependant à quelques changements.

Dans un premier temps, nous présenterons les adaptations effectuées sur l'activité de représentation de données, puis nous présenterons celles sur l'activité sur tablette.

I.1. Activité déconnectée : représentation de données

Cette activité demandait peu d'adaptation. Nous avons décidé de mettre des dessins plutôt que des lettres, car cela supprimait la présence de 0 en début de ligne. Cependant, il était important de placer au moins une grille avec cette configuration. Chaque élève avait donc quatre grilles à faire. Cela représentait trois dessins et une lettre (qui introduisait la problématique du 0 en début de ligne).

I.2. Activité connectée sur tablette

Nous avons décidé avec Audrey de faire expérimenter les élèves par le corps. Pour cela, nous avons choisi de leur faire faire le jeu du robot « idiot »^{lxiii}, en reprenant les instructions du jeu Lightbot. A l'aide d'un quadrillage à leur taille dessiné sur les dalles du préau dans la cour de récréation, un élève jouait le rôle du robot, et ne devait effectuer seulement les instructions données par les élèves, il ne devait pas prendre d'initiative.

Cette activité se décomposa en deux temps :

- Un élève jouait le rôle du robot, et les autres lui dictaient les instructions

- Un élève jouait le rôle du robot et devait suivre les instructions au sol

Chaque élève a expérimenté sur les deux temps.

Lorsque les élèves jouaient le rôle du robot, ils devaient effectuer le bon geste suivant l'instruction donnée par les autres. Ils devaient donc réfléchir pour ne pas se tromper.

Lorsqu'ils jouaient le rôle de « l'informaticien », ils devaient réfléchir en amont pour donner les bonnes instructions, et vérifier que le robot effectuait les bons gestes.

Les élèves orientés en ULIS nécessitent un enseignement adapté. Le fait de faire cette activité préalable à l'utilisation de la tablette permet en effet à ces élèves de se représenter physiquement le but du jeu, et ses spécificités. En effet, la droite/gauche n'était pas acquise pas tous les élèves. De plus, bien que les autres les aient acquises, tous n'étaient pas capables de prendre un autre repère qu'eux pour la droite/gauche. L'activité sur quadrillage à leur taille était donc indispensable.

La figure 37 montre un des quadrillages utilisés. Nous avons choisi de matérialiser les cases bleues (cases à allumer) par une croix, et les marches par un rond. C'est une première abstraction.



Figure 37 : quadrillage à taille humaine

➔ Passage sur tablette plus quadrillage réduit et personnage

Ensuite, nous passerons sur tablettes. Nous avons prévu un quadrillage sur feuille avec des cubes pour représenter les marches et un personnage pour représenter le robot.



Figure 38 : Lightbot et quadrillage sur feuille

La figure 38 montre le quadrillage sur feuille avec les cubes.

II. Description de l'intervention

Nous sommes intervenus dans la classe le vendredi matin 29 avril. J'ai procédé comme dans les deux autres classes dans un premier temps : évaluation diagnostique et présentation de l'intervention et des activités.

Activité déconnectée : représentation de données

J'ai choisi de construire au tableau en direct l'image avec le code plutôt que de montrer l'image et son code associé. Ainsi, les élèves pouvaient suivre pendant que je faisais. J'ai dû adapter mon discours en changeant le code « noir et blanc » avec « vide et coloré », (les cases censées être noircies apparaissaient blanches au tableau). Ils ont rapidement compris comment le code et l'image étaient reliés. Je leur ai donc distribué les fiches pour qu'ils puissent mettre en pratique.

Activité connectée sur tablette

J'ai expliqué aux élèves le but de l'application : allumer toutes les cases bleues à l'aide du robot. Je leur ai dit que le robot ne connaissait que quelques instructions. Je leur ai donc montré chaque instruction en format A4 pour qu'ils puissent deviner ce que chaque pictogramme signifiait. Une fois tous les pictogrammes présentés, nous sommes allés dans la cour pour faire le jeu du robot idiot.

Nous avons effectué plusieurs niveaux sur les grilles :

- Un niveau simple où il fallait juste avancer
- Un niveau où il fallait tourner et avancer
- Un niveau où il fallait sauter, tourner et avancer

L'abstraction rond équivaut à une marche et crois équivaut à une case bleue a été bien accueillie et n'a pas posé aucun souci.

Les élèves ont fait le rapprochement avec des activités déjà pratiquées avec leur maitresse, pour le repérage dans l'espace.

Ensuite, nous avons repris le dernier niveau (nous avons écrit la solution au sol) et avons placé les élèves à un autre point de départ, en leur demandant de suivre les instructions au sol. Ils se rendaient compte qu'en suivant ces instructions, le robot n'arrivait pas à aller allumer la lumière. Ils ont dit qu'ils n'étaient pas sur le bon point de départ. Nous leur avons donc demandé de donner les instructions pour se placer au bon point de départ, puis d'effectuer les instructions au sol une fois sur ce point. Cette dernière activité permettrait d'introduire l'appel de procédure.

Après cela, nous sommes rentrés en classe. J'ai pris une tablette et ait effectué le premier niveau devant eux, pour mettre en contexte ce que nous avons vu précédemment. Les élèves ont ensuite travaillé seuls.

Certains élèves ont eu besoin de la grille papier avec les cubes et personnage pour matérialiser en 3D la grille. D'autres n'y ont pas fait appel.

III. Analyse de l'intervention

Il est important de souligner que pour cette intervention, nous étions quatre adultes : Audrey, M. Dadeau, Catherine, l'AESH de la classe et moi. L'activité de représentation de données et la partie sous le préau ne nécessitait pas forcément d'être quatre, mais l'activité sur tablette aurait été plus délicate avec un seul adulte. Pour cette intervention, nous avons choisi de ne pas faire deux groupes d'élèves travaillant parallèlement sur les activités. N'étant que onze dans la classe, nous avons choisi de les faire travailler tous en même sur la même activité. La gestion du temps et l'encadrement des élèves dans ces conditions furent donc plus aisés.

Nous allons maintenant analyser les résultats des chaque activité.

début de ligne était plus tardive que dans les autres classes, cela peut expliquer la raison de cette meilleure compréhension. On peut aussi penser qu'ils résolvait les grilles de manière moins automatique que les autres classes, qu'ils prenaient plus le temps de réfléchir et donc faisaient moins d'erreurs lorsqu'ils étaient confrontés au 0. Cependant, il est important de souligner que ceux qui ont compris de suite avec la question avaient probablement envisagé cette solution avant ma venue, mais soumis à un problème nouveau, ils demandaient l'approbation de l'adulte.

Enfin, nous remarquons un problème auquel nous n'avions pas pensé. Certaines grilles ont les nombres « 11 ». Certains élèves les ont interprétés comme « 1,1 ». Cette « erreur » que nous n'avions pas envisagée suggère que ces élèves ont besoin d'être soumis à toutes les possibilités avant d'effectuer le travail. Cependant, ce cas est arrivé chez peu d'élèves mais il est important de la souligner car cela met en évidence la manière de penser de ces élèves.

III.2. Activité connectée sur tablette

Les élèves sont de suite entrés dans les activités. La phase sur le quadrillage sous le préau leur a plu.

Les élèves qui savaient déjà distinguer leur droite de leur gauche ont profité de cette activité pour s'entraîner. Ceux qui ne les distinguaient pas ont expérimenté sous une autre forme que les activités proposées par leur maîtresse pour cela. Nous pouvons noter que ceux pour lesquels ces notions étaient acquises ont réussi grâce à cette activité à se détacher de leur repère personnel pour prendre un repère extérieur. Ceux pour lesquels ces notions n'étaient pas acquises sont maintenant plus sûrs d'eux.

Les élèves ont également fait le rapprochement avec des activités de motricités proposés par leur maîtresse.

Le passage sur tablette montre deux types d'élèves :

- des élèves à l'aise avec le passage à l'abstrait
- des élèves peu à l'aise avec ce passage

Pour ces derniers, le quadrillage en 3D à l'aide des feuilles et cubes leur était nécessaire. Cependant, nous avons remarqué, que même aidé du quadrillage, ils avaient besoin d'un adulte pour ensuite transposer leur solution dans l'application. Pour ces élèves, il aurait peut-être fallu expérimenter plus sur le quadrillage sous le préau et à l'aide du quadrillage sur feuille et amener la tablette dans une autre séance.

Nous pouvons noter également que lorsque ces élèves, peu à l'aise avec la passage à l'abstrait, résolvait les niveaux en pas à pas. C'est-à-dire qu'ils réfléchissaient au parcours du robot, et à chaque instruction qu'ils mettaient dans le *main*, ils exécutaient le code. Il semblerait qu'ils avaient besoin de placer le robot pour pouvoir réfléchir à l'instruction suivante. Ils avaient des difficultés à « se faire le chemin du robot » dans la tête.

On peut noter que ces élèves étaient bien souvent les plus jeunes de la classe, ce qui peut expliquer ces différences.

IV. Bilan

Nous pouvons donc noter le bilan suivant :

- Bien qu'impatients pour la plupart d'utiliser les tablettes, certains ont montré une certaine anxiété à l'approche de celle-ci. Cependant, cette angoisse ne les a pas empêchés de travailler dessus une fois rassurés.
- Les élèves, de niveaux CP jusqu'au CM2, avaient des niveaux très variés. Les résultats sont donc moins homogènes que pour les classes de CM1-CM2.
- Les tables étaient organisées en îlot de quatre, de façon à former des groupes hétérogènes de niveaux. Cette disposition a créée, comme dans les autres classes, une dynamique de groupe, amenant les élèves à se dépasser.
- On peut aussi noter que les élèves ont été concentrés toute la matinée sur les tâches, et ont réalisé un effort soutenu, et n'ont pas abandonné.
- Pour l'activité de représentation de données, quatre élèves se sont montrés performants et ont demandé peu d'aide.
- Les adaptations effectuées étaient nécessaires et bien réfléchies. Une intervention dans une classe spécialisée nécessite de travailler avec l'enseignant de cette classe, qui connaît le niveau de ses élèves et leur capacités.

Audrey m'a fait parvenir la fiche compte-rendu qu'elle a faite à destination des parents de ces élèves (cf annexe 8). Les élèves ont réclamé le nom de l'application pour pouvoir l'installer chez eux, et ont demandé à Audrey pour terminer les fiches dans la journée.

Conclusion

Nous pouvons noter dans un premier temps que les activités proposées aux élèves ont suscité un réel enthousiasme. Les activités étaient ludiques et avaient toutes un côté challenge. Les élèves ont fait de nombreux rapprochements entre les activités proposées et des activités pratiquées en classe auparavant. Cela conforte l'idée que l'introduction de l'informatique peut se faire à travers plusieurs disciplines. Les activités déconnectées sont aisément réalisables en groupe classe. Les élèves comprennent vite le problème et mettent en place des stratégies de résolution efficace. Cependant, celle sur tablette est difficilement transposable en dehors de ce travail. Pour nos interventions, il nous a fallu entre dix et quatorze tablettes. Certaines écoles ont été choisies pour être classes pilotes dans le plan numérique 2015 et ont donc à disposition une trentaine de tablettes. Cependant, la majorité des écoles n'en ont pas. Canopé (anciennement CRDP) met à disposition, en théorie, du matériel numérique^{lxiii}. J'ai donc effectué une demande pour cela. Je n'ai jamais reçu de réponse. Lorsque je me suis rendue sur place pour enfin obtenir une réponse, on m'a répondu que cela n'était pas un projet à leur sens. Le département informatique de l'université a accepté de nous prêter le nombre de tablettes nécessaires. Sans cela, nos interventions auraient été compromises. Les enseignants qui souhaiteraient ainsi faire ce genre d'activités en classe devront probablement insister auprès de Canopé pour obtenir le matériel nécessaire. De plus, pour chaque intervention, nous étions au minimum trois encadrants. L'activité sur tablette requiert un accompagnement poussé des élèves. Bien que la majorité des élèves soient enthousiasmés par l'utilisation des tablettes, certains peuvent faire preuve d'anxiété. Il faut donc pouvoir accompagner tous les élèves, que ce soit dans l'appropriation du matériel ou dans la résolution des niveaux. Un enseignant, seul dans sa classe, sera peut-être en situation difficile pour faire cela. Par ailleurs, le problème lié à la disponibilité du matériel ne doit pas être un obstacle à l'introduction de la programmation. Nous avons vu qu'il existait des activités « déconnectées ». Ces activités permettent, de plus, de travailler des notions abstraites, non travaillables sur tablettes. Elles ne doivent donc pas être négligées.

Ensuite, nous pouvons noter les apports qui ont été apportés aux élèves par les interventions :

- ***apprendre à décomposer un problème*** : dans l'application Lightbot, l'utilisation de *motifs* met en évidence cela. Pour résoudre les grilles du niveau 2, il faut déterminer un *motif*, un sous-problème. Cela est important car cela est transposable à d'autres disciplines. Cette capacité à analyser un problème pour en extraire des sous-problèmes est nécessaire dans la suite des apprentissages.
- ***mettre en place des stratégies de résolution*** : dans les trois activités proposées, les élèves doivent réfléchir avant de se lancer dans la résolution. Ils mettent donc en place des stratégies de résolution.

Conclusion

- **apprendre à être méthodique** : cette notion a été bien mise en évidence par l'activité de représentation des nombres en binaire. Pour la résolution des cartes cibles, ils commençaient toujours par additionner la carte directement inférieure avec celles inférieures et ainsi de suite. Ils faisaient preuve de méthode. Cela permettra aux élèves, dans d'autres disciplines, d'être mieux organisés dans la résolution de la tâche.
- **persévérer** : l'informaticien teste et debugg son programme. Les élèves effectuaient également ces phases lorsqu'ils étaient sur l'application Lightbot. Bien que certains réclamaient beaucoup d'aide, aucun n'a abandonné avant la fin. Ils ont, de plus, tous mis en place cette rigueur : implémenter un algorithme, le tester, le débogger si besoin. Cette rigueur est importante, comme la méthode, dans toutes les disciplines.
- **travailler des notions sous différentes formes** : les élèves ont naturellement fait des liens avec des activités déjà pratiquées. Ceci est important à souligner car beaucoup de critiques face à l'introduction de la programmation en primaire viennent du fait que les enseignants ne sont pas formés. En effet, cela pose problème. Cependant, les activités proposées ont fait écho chez les élèves. Les notions travaillées en informatique sont des notions d'autres disciplines, misent dans un contexte particulier.
- **Apprendre à travailler en équipe** : tout au long de leur scolarité, les élèves seront amenés à effectuer des travaux de groupe. Les activités proposées ont favorisé ce travail de groupe. Il est donc intéressant de proposer des activités où cette notion est nécessaire à la résolution du problème.

Nous pouvons également relever les apports pour les maîtres/maîtresses de primaire :

- **varier les situations d'apprentissage** : les maîtres et maîtresses des classes dans lequel je suis intervenue m'ont fait le même retour : « jamais je n'aurai pensé à proposer cela pour une séance d'informatique ». cela montre bien que n'étant pas formés, les enseignants ont des a priori sur cet enseignement. Or, ils ont pris conscience que des certaines de leurs activités pouvaient être proposées pour une séance d'informatique avec quelques ajustements. Par exemple, le jeu *Mathador* (ou *Le compte est bon*) serait repris pour la notation binaire et nécessiterai les adaptations suivantes : utiliser uniquement l'addition et les nombres de puissance de 2.
- **Désacraliser la programmation et changer son point de vue** : ce point rejoint le premier, les enseignants ont des a priori qui les empêchent de voir cet enseignement comme bénéfique. Les interventions ont permis à ces enseignants d'être accompagnés pour une première séance, et de se sentir plus à l'aise pour continuer avec une deuxième.

Cependant, il est évident que les enseignants, n'étant pas formés, ne pourront pas proposer un enseignement complet de la programmation. La formation mise en place lors de l'expérience de 58 lycées est intéressante et pertinente. Le gouvernement, à cette époque, avait compris l'importance et la nécessité de former les enseignants. Cela demande néanmoins beaucoup de moyens. On peut alors envisager de ne proposer que des formations courtes,

Conclusion

avec une initiation à la programmation par l'utilisation de logiciel type Scratch et des cours théoriques sur les notions générales de l'informatique. Enfin, pour compléter la formation, il serait intéressant de mettre en évidence les liens possibles avec les autres disciplines.

Enfin, nous pouvons noter les apports à cette intervention pour moi, en tant qu'enseignante et titulaire d'un Master en informatique :

- *L'envie de continuer* : il est évident que je proposerai à nouveaux ce genre d'activités et d'autres à mes élèves dans ma carrière. Ces interventions se sont toutes bien déroulées, et m'ont, de ce fait, donné envie de persévérer dans cette voie. Je ne m'arrêterai pas à ces classes tests.
- *La possibilité de mettre à profit mes compétences d'informaticienne dans mon nouveau métier* : bien qu'ayant choisi de quitter le monde de l'informatique, il me semble inconcevable de renier cette partie de ma vie. Je suis donc satisfaite de la volonté du gouvernement de proposer l'introduction de la programmation dès l'école primaire.

Suite à ces activités, une majorité d'élèves a souhaité continuer les activités proposées. Ils m'ont demandé de revenir pour poursuivre le travail (sur tablette et sur table) et ont demandé le nom de l'application. Il est évident qu'ils ont envie de connaître le fonctionnement des appareils qu'ils utilisent tous les jours. De plus, les activités étant présentées sous forme de jeux, ils ont appris, sans se rendre compte du travail effectué.

Les enseignants des classes dans lesquels je suis intervenue m'ont dit qu'ils n'auraient pas pensé à proposer ce genre de situations dans le cadre de l'enseignant de l'informatique. Avec l'arrivée de la programmation dans les programmes, ces enseignants auront déjà été initié à quelques activités possibles, et pourront les proposer à nouveau. Joseph, de l'école Fourier, m'a dit qu'il n'aurait jamais pensé à emprunter des tablettes à Canopé. Il en fera peut-être la demande à l'avenir. Christophe, de l'école Fanart, nous a précisé qu'une de ses collègues étaient également intéressée par notre venue. Nous allons voir pour programmer une intervention dans sa classe. Audrey, de la classe d'ULIS, nous a présenté la suite qu'elle allait effectuer dans sa classe : continuer les niveaux de l'application Lightbot avec la version flash^{lxiv} proposée sur le site Lightbot. De plus, l'étape sur quadrillage sous le préau ressemblant aux exercices de motricité, elle reprendra probablement l'idée pour varier les situations d'apprentissages.

Les retours des élèves et enseignants indiquent que l'introduction de la programmation dès l'école primaire est possible et bien accueillie. Il serait intéressant cependant de pouvoir faire cette intervention dans d'autres classes, et d'élargir aux cycles 1 et 2.

Concernant la mise en œuvre de cela, peut-on simplement fournir les fiches séquences et séances aux enseignants pour qu'ils mettent en œuvre l'intervention seuls ? Ou serait-il plus pertinent de les accompagner ? Nous pouvons imaginer la présence d'itinérants en informatique, comme il y a des itinérants en langue. Ou cela pourrait être une mission de l'animateur TICE ? Cependant, le rôle de l'animateur TICE étant déjà bien fourni, il serait envisageable de créer le poste de conseiller pédagogique en informatique.

Annexe

Annexe 1 : Séquence

Niveau : Cycle 3		Nom : Introduction à la programmation	Discipline : Sciences et technologie
Objectifs <ul style="list-style-type: none"> - Découvrir l'algorithme - Découvrir la représentation des données dans un ordinateur 			
Activité 1 : représentation binaire des nombres			
Connaissances	Capacités	Situation d'apprentissage groupes de 4-5	
+ un nombre s'écrit à l'aide de 0 et de 1 (notation binaire positionnelle)	+ être capable de donner la représentation binaire d'un nombre	+ chaque élève dispose d'une carte avec un nombre d'une puissance de 2 : 1, 2, 4, 8 (et 16) + tirer un nombre cible et trouver la démarche à mettre en place pour atteindre ce nombre cible à l'aide d'addition des cartes présentes + chaque élève dispose de carte avec au verso un « 0 » et au recto un « 1 », refaire les mêmes activités avec ces cartes → introduction de la nécessité de la position	
Les élèves ont découvert la notation binaire pour écrire un nombre.			
Activité 2 : représentation des données			
Connaissances	Capacités	Situation d'apprentissage groupes de 4-5	
+ la notation binaire est le format de représentation de l'information (à la fois stockage et transmission impression-fax)	+ être capable de décoder	+ présentation de l'activité : image dans les pc, comment sont-elles stockées ? Imprimées ? alors qu'on ne peut se servir que de nombres dans un pc ? + présentation de la lettre « a » avec son code, en expliquant que chaque carré = un pixel + présentation de la lettre « c » avec son code, pour mettre en évidence la taille du quadrillage, qui sera différent pour une lettre haute ou basse + décodage lettres et images	
Les élèves ont appris coder une image à l'aide de la notation binaire.			
Activité 3 : tri de taille			
Connaissances		Situation d'apprentissage groupes de 4-5	
+ pour un même objectif, on peut faire plusieurs solutions, mais on peut choisir une solution plus efficace (contrainte de l'ordi)		+ tri par sélection	

Annexe

	+ tri bubble sort
Les élèves se sont triés par taille à l'aide de plusieurs algorithmes et ont pu voir que certaines solutions sont plus rapides que d'autres	

Annexe 2 : Activité de représentation des nombres en binaire

Niveau : C3	Activité : 1 Représentation binaire des nombres	Nom : Introduction à la programmation	Discipline : Sciences et technologie	Durée : 20 min
Durée	Déroulement			Matériel
2 min	<p>Positionner les cartes dans l'ordre devant les élèves. Leur demander ce qu'ils remarquent sur ces cartes → chaque carte est le double de la carte située à sa gauche. Leur demander alors quel serait la prochaine carte.</p> <p>Demander aux élèves quel est le plus petit et le plus grand nombre possible que l'on peut faire à l'aide de ses cartes → 0 en ne mettant aucune carte, 15 (31) en les mettant toutes. Les élèves doivent donc penser à l'addition des cartes</p>			<p>+ cartes des puissances de 2 inférieures à 2^n, $n =$ nombres d'élèves du groupe</p>
9 min	<p>Les élèves tirent une carte cible et doivent trouver la solution en respectant la consigne : toutes les cases du tableau doivent être remplies d'une carte → ils doivent donc retourner les cartes en face cachée s'ils ne souhaitent pas en utiliser une.</p> <p>A ce stade, les élèves ne poseront pas forcément les cartes dans l'ordre.</p>			<p>+ cartes des puissances de 2 inférieures à 2^n, $n =$ nombres d'élèves du groupe</p> <p>+ tableau de 4 (5) cases avec un égal à la fin plus une autre case</p> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px;"></div> <p>+ cartes cibles ≤ 15 (≤ 31)</p>
9 min	<p>Avec les cartes 0-1, les élèves devront trouver une solution pour résoudre les problèmes : positionner les cartes en fonction du nombre qu'ils souhaitent.</p> <p>Intervenant naïf qui doit vérifier le calcul pour trouver la position</p>			<p>+ cartes avec au recto « 0 » et au verso « 1 »</p> <p>+ tableau de 4 (5) cases avec un égal à la fin plus une autre case</p> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px; margin-right: 5px;"></div> <div style="border: 1px solid black; display: inline-block; width: 20px; height: 20px;"></div> <p>+ cartes cibles ≤ 15 (≤ 31)</p>

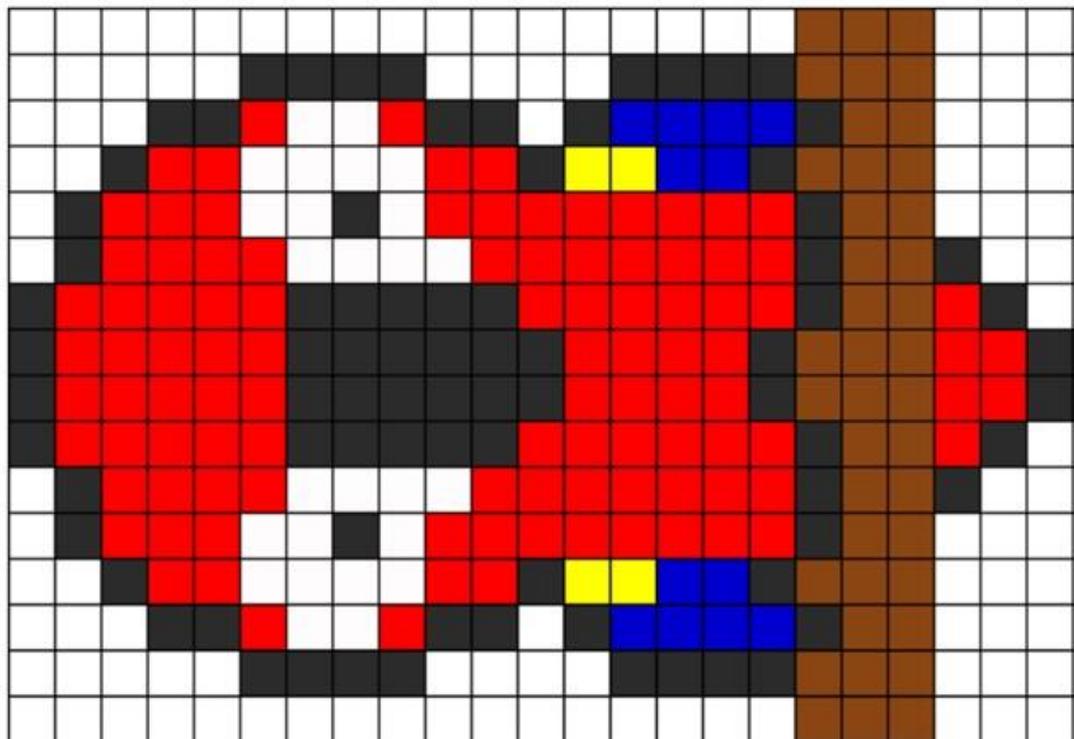
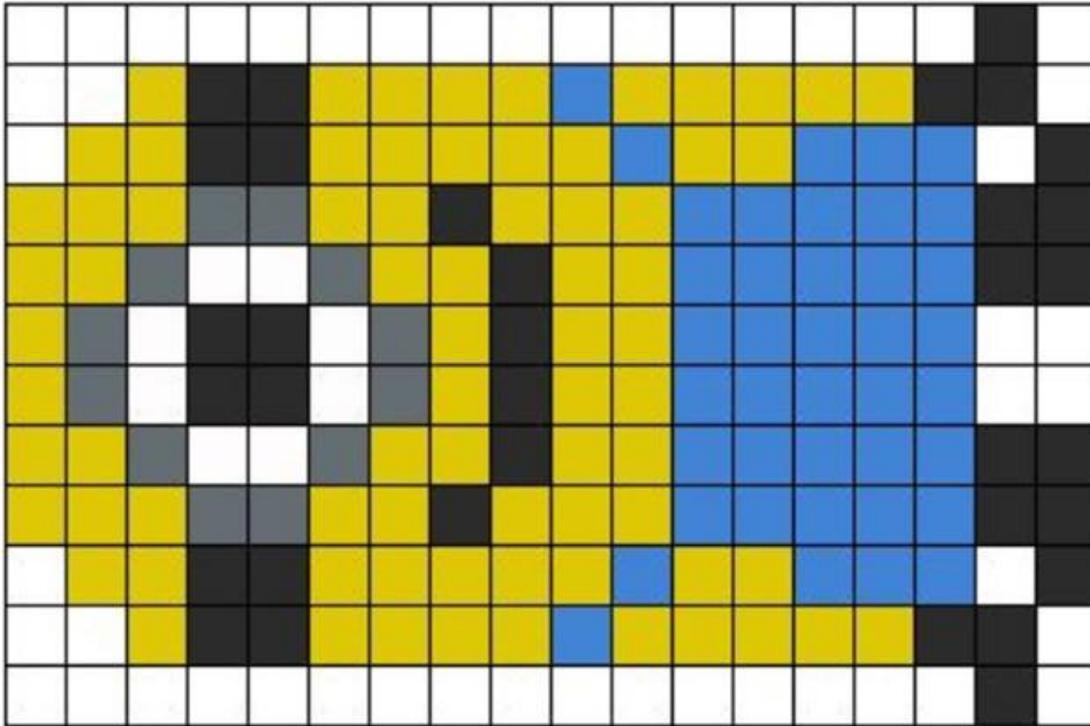
Annexe 3 : Activité de représentation des données

Niveau : C3		Activité : 2 Impression Fax	Nom : Introduction à la programmation	Discipline : Sciences et technologie	Durée : 20 min
Durée	Déroulement			Matériel	
1 min	<p>Consigne : « comment les images sont-elles représentées dans un ordinateur ? »</p> <p>Réponses possibles :</p> <ul style="list-style-type: none"> - Elles sont stockées telles quelles - Elles sont stockées à l'aide de 0 et de 1 (→ renvoyer à l'activité précédente) 				
2 min	<p>Présentation de l'image pixellisée du « a » minuscule avec son code</p> <p>Consigne : « Que voyez-vous ? »</p> <p>Réponses possibles :</p> <ul style="list-style-type: none"> - Un « a » - Une grille avec un « a » et des chiffres à droite <p>Consigne : « à quoi correspondent ses chiffres ? »</p> <p>Réponses possibles :</p> <ul style="list-style-type: none"> - Pour dessiner le « a » <p>Nous allons dessiner le « a » à partir du code pour vérifier que ces chiffres servent à ça.</p> <p>Sur une feuille plastifiée avec un quadrillage et le code, dessiner le « a » selon les consignes des élèves.</p>			Affiche plastifiée du « a » et son code	
2 min	<p>Présentation du « c » minuscule avec son code et mise en évidence de la hauteur et de la largeur, la même que pour le « a ».</p> <p>Demander aux élèves si toutes les lettres pourront alors entrer dans ces dimensions</p> <p>Réponses possibles :</p> <ul style="list-style-type: none"> - Non, les b, l, etc (lettres hautes) ne pourront pas - Les p, q, etc (lettres basses) ne pourront pas <p>Comment y remédier ?</p> <p>Réponse :</p> <ul style="list-style-type: none"> - Ajouter des lignes au-dessus (en dessous) 			Affiche plastifiée du « c » et son code	
15 min	Distribution de quadrillages avec le code pour qu'ils retrouvent les lettres			Affiches de quadrillages avec les codes des lettres	

Annexe 4 : Activité de tris

Niveau : C3		Activité : 3 Tri de tailles	Nom : Introduction à la programmation	Discipline : Sciences et technologie	Durée : 15 min
Durée	Déroulement			Matériel	
2 min	<p>Aligner les élèves le long du tableau (d'un mur ?), sans leur donner de consigne.</p> <p>Leur préciser qu'on aimerait qu'ils soient rangés par taille, du plus petit au plus grand. Comment pouvons-nous faire ?</p> <p>1^{ère} solution : les élèves vont tous se déplacer en même temps pour trouver leur place → erreurs possibles, brouhaha</p>			+ fiche pour marquer la place initiale des élèves	
5 min	<p>Demander aux élèves les points faibles de cette solution</p> <p>Demander aux élèves de reprendre leur place initiale, et leur préciser que cette fois, nous ne voulons pas qu'ils se déplacent tous à la fois, car cela fait trop de bruit.</p> <p>Les aider si besoin en leur demandant si le premier élève est bien plus petit que le deuxième élève. Pour cela, il faut vérifier. Leur demander alors de s'avancer, de comparer leur taille et de se placer en conséquence. Est-on sûr maintenant que le premier est le plus petit ? non, on doit alors le comparer avec chacun. Une fois comparé avec tous, peut-on être sûr que c'est le plus petit ? oui, alors on passe au deuxième, et ainsi de suite → tri par sélection</p> <p>Un élève compte le nombre de mouvements</p>			+ fiches pour compter les comparaisons et échanges	
5 min	<p>Refaire avec bubble sort</p> <p>Un élève compte le nombre de mouvements</p>			+ fiches pour compter les comparaisons et échanges	
3 min	<p>Parler de l'efficacité en comparant le nombre de mouvements pour les 2 algorithmes</p> <p>Pourquoi fait-on des algo de tri ? simplifier la recherche, besoin de présenter de façon triée</p>				

Annexe 5 : Représentation des données en couleurs



Annexe 6 : Solutions lightbot



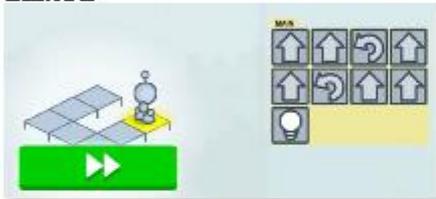
These solutions are meant to be used as a guide. Refrain from showing students a solution directly and encourage testing partial-solutions and using a "what-if-we-do-this" kind of hinting.

There can be multiple correct solutions for each level, here we provide common solutions.

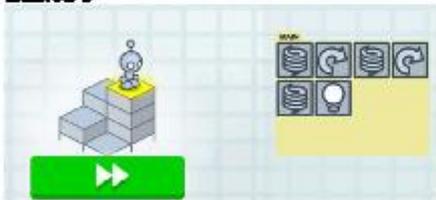
Basics 1



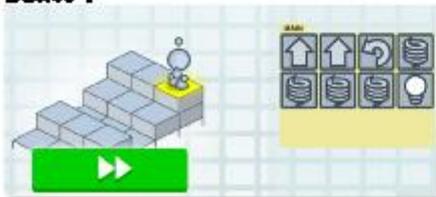
Basics 2



Basics 3

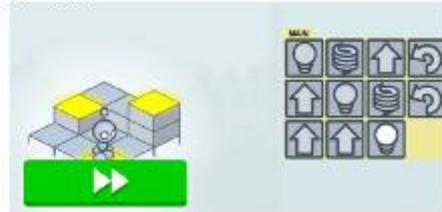


Basics 4

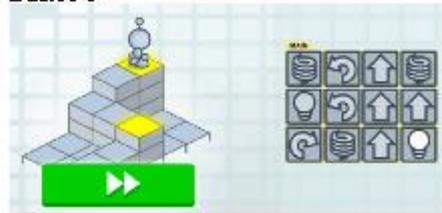


**Lightbot Hour of Code '14
Teacher Solutions**

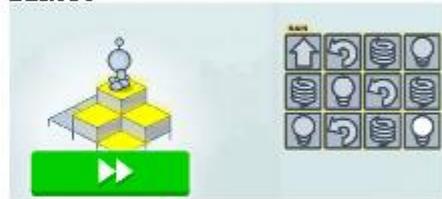
Basics 5



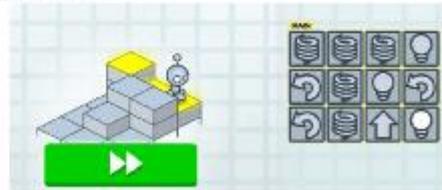
Basics 6



Basics 7

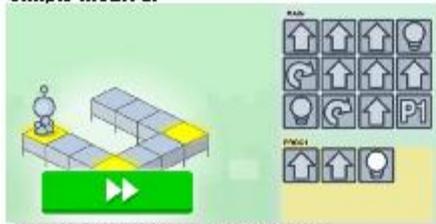


Basics 8

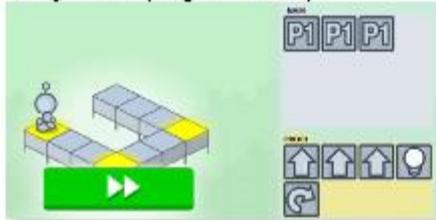


The first two levels only introduce Procedures without mentioning their usefulness for patterns. There are two types of solutions students may provide. 1) is just P1 used as an extension to MAIN, whereas 2) makes use of P1 multiple times as a pattern.

Procedures 1
Simple Method:



Using Pattern (Elegant Solution):



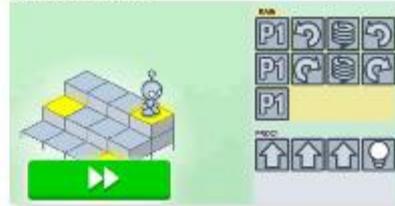
Procedures 2
Simple Method:



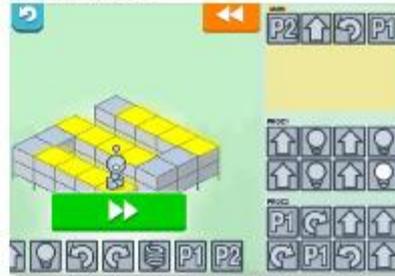
Using Pattern (Elegant Solution):



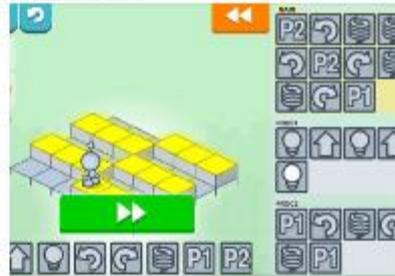
Procedures 3



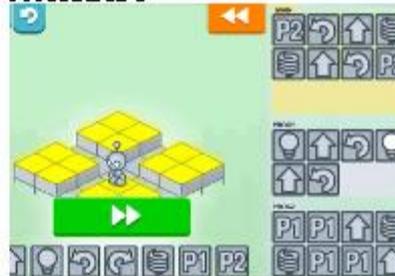
Procedures 4



Procedures 5



Procedures 6



lightbot.com/fr/oc2014.html

Annexe 7 : Fiche activités représentation de données

Sciences : Découverte de l'informatique – Représentation de données

Consigne : Décrypte le code pour faire apparaître les images

					1,3
					4,1
					1,4
					0,1, 3,1
					0,1, 3,1
					1,4

							0,3,1,2
							0,1,2,1, 2,1
							0,1,2,1, 2,1
							0,1,2,1, 2,1
							0,1,2,1, 2,1
							0,1,2,1, 2,1
							0,1,2,1, 2,1

				1,3
				1,1
				1,1
				0,4
				1,1
				1,1
				1,1
				0,2

			1,1
			1,1
			1,1
			1,1
			1,1
			1,1
			1,1

Annexe

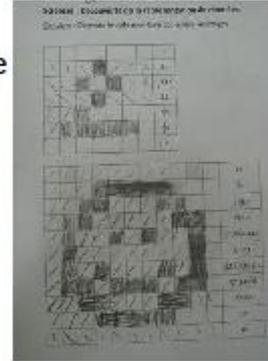
					1,1
					1,1
					0,3
					1,1
					1,1
					1,1, 2,1
					1,1, 2,1
					2,2

																			4,1 1
																			4,9 ,2,1
																			4,9 ,2,1
																			4,1 1
																			4,9
																			4,9
																			5,7
																			0,1 7
																			1,1 5

Vendredi 29 avril, nous sommes devenus des informaticiens !

Frédéric (maître de conférence en informatique à l'université de Franche-Comté) et Aude (professeur des écoles), sont venus nous faire une animation sur le métier d'informaticien. Nous avons essayé de comprendre comment « réfléchit » un ordinateur.

Nous avons commencé par une activité de décodage de données. Les données ce sont toutes les informations que l'on enregistre dans un ordinateur (photos, images, musiques, films...) et toutes ces informations sont codées. Nous nous sommes entraînés à les décoder pour découvrir les dessins mystères. Nous avons appris que tout était représenté en pixels et codé dans la mémoire de l'ordinateur, même le texte que l'on tape !



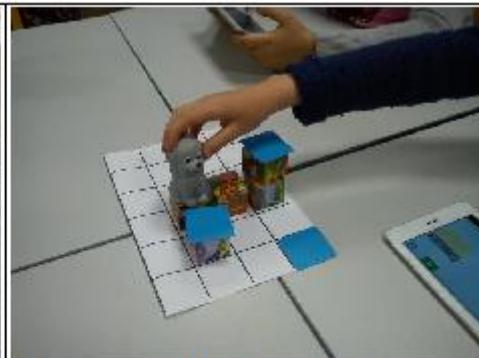
Ensuite, nous avons appris à programmer sous forme de jeu. Pour cela, nous avons dû utiliser des instructions qui sont les ordres que l'ordinateur comprend. Dans notre jeu, c'était « avancer, tourner à droite, tourner à gauche, sauter, activer »

Nous avons commencé par le « jeu du robot idiot » où un élève faisait le robot et devait suivre les instructions des camarades. Puis, le robot devait lire les instructions.

Ensuite, nous sommes passés sur tablettes numériques. Nous pouvions nous aider d'un quadrillage avec un vrai personnage pour mieux nous représenter les actions.



comprendre par le corps



Changer de point de repère

Nous avons appris que pour qu'un logiciel (jeu, traitement de texte...) fasse ce que l'on veut, il faut qu'un informaticien ait d'abord écrit un programme avec des instructions et des procédures (ensemble d'instructions qui peut être utilisé plusieurs fois).

(jeu gratuit à télécharger sur tablette et ordinateur : lightbot)

Bibliographie – Nétographie

- ⁱ The World Economic Forum : The Global Information Technology Report 2015 [en ligne] http://www3.weforum.org/docs/WEF_Global_IT_Report_2015.pdf [consulté le 13/05/2015]
- ⁱⁱ Wikipédia : Plan Calcul [en ligne] http://fr.wikipedia.org/wiki/Plan_Calcul [consulté le 13/05/2015]
- ⁱⁱⁱ LesEchos.fr : Le Plan Calcul, l'échec d'une ambition [en ligne] http://www.lesechos.fr/20/07/2012/LesEchos/21231-055-ECH_le-plan-calcul--l-echec-d-une-ambition.htm [consulté le 13/05/2015]
- ^{iv} MERCOUROFF Wladimir, 1983, Dix ans d'informatique dans l'enseignement secondaire : 1970-1980, La revue française de pédagogie, N°63, p. 90-91
- ^v Association Enseignement Public et Informatique : L'évaluation de l'expérience des 58 lycées [en ligne] <http://www.epi.asso.fr/revue/23/b23p068.htm> [consulté le 14/05/2015]
- ^{vi} Wikipédia Disque : Compact [en ligne] http://fr.wikipedia.org/wiki/Disque_compact#Histoire [consulté le 19/05/2015]
- ^{vii} Association Enseignement Public et Informatique : Présentation [en ligne] http://www.epi.asso.fr/association/epi_presentation.htm [consulté le 15/05/2015]
- ^{viii} MINISTERE DE L'EDUCATION NATIONALE : Plan Informatique Pour tous, Circulaire n°85-135 et n°85-208 du 29 mai 1985, B.O. n°24 du 13 juin 1985, p. 1737 à 1740
- ^{ix} Association Enseignement Public et Informatique : L'informatique pour tous [en ligne] <http://www.epi.asso.fr/revue/37/b37p023.htm> [consulté le 10/03/2015]
- ^x Association Enseignement Public et Informatique : Brochure « Informatique Pour Tous » [en ligne] <http://www.epi.asso.fr/revue/histo/h85-ipt.htm> [consulté le 06/05/2015]
- ^{xi} Wikipédia : Plan informatique pour tous [en ligne] http://fr.wikipedia.org/wiki/Plan_informatique_pour_tous [consulté le /05/2015]
- ^{xii} Association Enseignement Public et Informatique : Les stages Informatique pour tous [en ligne] <http://www.epi.asso.fr/revue/histo/h85-stageIPT.htm> [consulté le 19/05/2015]
- ^{xiii} Association Enseignement Public et Informatique : Option Informatique [en ligne] http://www.epi.asso.fr/fic_pdf/b46p032.pdf [consulté le 13/05/2015]
- ^{xiv} MINISTERE DE L'EDUCATION NATIONALE, 1985, Circulaire n° 85-0136 et n° 85-208 du 19 mai 1985
- ^{xv} Association Les Clionautes : Retour sur le « plan informatique pour tous » [en ligne] <http://www.clionautes.org/spip.php?article3051#VVszGUZCghR> [consulté le 13/05/2015]
- ^{xvi} Association Enseignement Public et Informatique : Le Nano Réseau USTL [en ligne] <https://hal.archives-ouvertes.fr/file/index/docid/30674/filename/b38p103.pdf> [consulté le 06/05/2015]
- ^{xvii} ARCHAMBAULT Jean-Pierre, 2005, 1985, vingt ans après..., Médialog, N°54 [en ligne] http://reseaux.blog.lemonde.fr/files/2013/03/IPT_Archambault.pdf [consulté le 13/05/2015]
- ^{xviii} Association Enseignement Public et Informatique : Les nouvelles technologies de l'information et de la communication dans l'enseignement [en ligne] http://www.epi.asso.fr/fic_pdf/b88p047.pdf [consulté le 18/05/2015]
- ^{xix} MINISTERE DE L'EDUCATION NATIONALE, DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE : B2i [en ligne] <http://eduscol.education.fr/cid46073/b2i.html> [consulté le 13/05/2015]
- ^{xx} MINISTERE DES FINANCES ET DES COMPTES PUBLICS : 57 objectifs prioritaires pour 2020 [en ligne] http://www.economie.gouv.fr/files/files/import/2011_france_numerique_consultation/2011_francenumerique2020objectifs.pdf [consulté le 14/05/2015]
- ^{xxi} MINISTERE DE L'EDUCATION NATIONALE, DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE : Concertation nationale sur le numérique pour l'éducation [en ligne] <http://ecolenumerique.education.gouv.fr/concertation/fin-de-la-premiere-etape/reponses-questionnaire#content> [consulté le 13/05/2015]

- ^{xxii} Association CIIP – Le café pédagogique : Hollande, ce qu'on sait de son plan numérique [en ligne] <http://www.cafepedagogique.net/lexpresso/Pages/2014/09/19092014Article635467071516385083.aspx> [consulté le 18/05/2015]
- ^{xxiii} Académie de Besançon – Dailymotion : Concertations, pratiques pédagogiques et éducatives [en ligne] http://www.dailymotion.com/video/x2h3xrs_concertation-pratiques-pedagogiques-et-educatives_school [consulté le 18/05/2015]
- ^{xxiv} Wikipédia : Scratch [en ligne] http://fr.wikipedia.org/wiki/Scratch_%28langage%29 [consulté le 20/05/2015]
- ^{xxv} Mairie d'Issy : Les petits Isséens s'initient au code informatique et à la programmation robotique [en ligne] <http://issy.com/vie-quotidienne/enfance/education/lecole-de-demain-deja-a-issy/apprentissage-du-code-informatique-a-l-ecole> [consulté le 06/05/2015]
- ^{xxvi} BlommborgBusiness : Soon, students may learn to code instead of taking French class [en ligne] <http://www.bloomberg.com/news/articles/2015-04-02/soon-students-may-learn-to-code-instead-of-taking-french-class> [consulté le 20/11/2015]
- ^{xxvii} Isabelle Collet, 2001, *Effet de genre : le paradoxe des études d'informatique*, TIC & Société [en ligne] <https://ticetsociete.revues.org/955> [consulté le 20/11/2015]
- ^{xxviii} Code.org : About us [en ligne] <https://code.org/about> [consulté le 20/11/2015]
- ^{xxix} Code.org : Computer Science Fundamentals for elementary school [en ligne] <https://code.org/educate/k5> [consulté le 20/11/2015]
- ^{xxx} Studio Code : Heure de code Angry birds [en ligne] <https://studio.code.org/hoc/1> [consulté le 09/12/2015]
- ^{xxxi} Studio Code : Cours 1 à destination des lecteurs débutants [en ligne] <https://studio.code.org/s/course1> [consulté le 09/12/2015]
- ^{xxxii} Le Figaro : Les sciences informatiques de retour à l'école britannique [en ligne] <http://www.lefigaro.fr/international/2012/01/11/01003-20120111ARTFIG00576-les-sciences-informatiques-de-retour-a-l-ecole-britannique.php> [consulté le 21/11/2015]
- ^{xxxiii} NextGenSkills.com : Campaign Objective [en ligne] <http://www.nextgenskills.com/about-us/> [consulté le 21/11/2015]
- ^{xxxiv} Departement of Education : The national curriculum in England [en ligne] https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/335116/Master_final_national_curriculum_220714.pdf pages 217 à 220 [consulté le 21/11/2015]
- ^{xxxv} Scratch : Page d'accueil [en ligne] <https://scratch.mit.edu/> [consulté le 21/11/2015]
- ^{xxxvi} Numerama : Des cours de programmation obligatoires à l'école, en Estonie [en ligne] <http://www.numerama.com/magazine/23615-des-cours-de-programmation-obligatoires-a-l-ecole-en-estonie.html> [consulté le 21/11/2015]
- ^{xxxvii} Université de Tartu : Tiger Leap program as a beginning of 21-st century educations [en ligne] <http://www.ut.ee/eLSEConf/Kogumik/Magi.pdf> [consulté le 21/11/2015]
- ^{xxxviii} BBC News : How Estonia became E-Stonia [en ligne] <http://www.bbc.com/news/business-22317297> [consulté le 21/11/2015]
- ^{xxxix} Université de Vilnius : Valentina Dagiené [en ligne] <http://ims.mii.lt/valentina/> [consulté le 21/11/2015]
- ^{xl} Brebas : Statistiques du nombre de participants de 2004 à 2014 [en ligne] <http://bebras.org/sites/default/files/Statistics%202004-2014.xlsx> [consulté le 21/11/2015]
- ^{xli} Brebas : Statistique du nombre de participants par pays [en ligne] <http://bebras.org/?q=countries> [consulté le 21/11/2015]
- ^{xlii} Castor Informatique France : Annales [en ligne] <http://concours.castor-informatique.fr/?tab=home> [consulté le 21/11/2015]
- ^{xliiii} Ils sont là : Premier Coding goûter de Paris. Des kids, du code, du cake [en ligne] <http://ils.sont.la/post/premier-coding-gouter-paris-kids-code-et-cake> [consulté le 09/12/2015]
- ^{xliv} Framasoft : outils utilisés lors des coding goûter <https://lite.framapad.org/p/codinggouter> [consulté le 09/12/2015]
- ^{xlv} GameSalad : Page d'accueil [en ligne] <https://gamesalad.com/> [consulté le 09/12/2015]
- ^{xlvi} RoboZZle : Page d'accueil [en ligne] <http://www.robozzle.com/> [consulté le 09/12/2015]

- ^{xlvi} Mairie d'Issy : Les petits Isséens s'initient au code informatique et à la programmation robotique [en ligne] <http://issy.com/vie-quotidienne/enfance/education/lecole-de-demain-deja-a-issy/apprentissage-du-code-informatique-a-l-ecole> [consulté le 08/12/2015]
- ^{xlviii} Wikipédia : Aldebaran Robotics [en ligne] https://fr.wikipedia.org/wiki/Aldebaran_Robotics [consulté le 08/12/2015]
- ^{xliv} LeCube : Le cube, centre de création numérique [en ligne] http://www.lecube.com/fr/le-cube-centre-de-creation-numerique_63 [consulté le 09/12/2015]
- ^l GamePlay : La carte du projet [en ligne] <http://gameplay.lecube.com/la-carte-du-projet/> [consulté le 09/12/2015]
- ^{li} Wikipédia : Pédagogie Montessori [en ligne] https://fr.wikipedia.org/wiki/P%C3%A9dagogie_Montessori [consulté le 08/05/2016]
- ^{lii} CSUnplugged : Page d'accueil [en ligne] <http://csunplugged.org/> [consulté le 07/05/2016]
- ^{liii} CSUnplugged : About [en ligne] <http://csunplugged.org/about/> [consulté le 07/05/2016]
- ^{liiv} CSUnplugged : Book of activities [en ligne] http://csunplugged.org/wp-content/uploads/2015/03/CSUnplugged_OS_2015_v3.1.pdf [consulté le 07/05/2016]
- ^{liv} Wikipédia : Pixel art [en ligne] https://fr.wikipedia.org/wiki/Pixel_art [consulté le 02/04/2016]
- ^{lvi} Wikipédia : Classement alphabétique [en ligne] https://fr.wikipedia.org/wiki/Classement_alphab%C3%A9tique [consulté le 07/05/2016]
- ^{lvii} Wikipédia : Tri par sélection [en ligne] https://fr.wikipedia.org/wiki/Tri_par_s%C3%A9lection [consulté le 07/05/2016]
- ^{lviii} Wikipédia : Tri à bulles [en ligne] https://fr.wikipedia.org/wiki/Tri_%C3%A0_bulles [consulté le 07/05/2016]
- ^{lix} Lightbot : Page d'accueil [en ligne] <https://lightbot.com/> [consulté le 15/04/2016]
- ^{lx} Mathador : Page d'accueil [en ligne] <http://www.mathador.fr/index.php> [consulté le 06/05/2016]
- ^{lxix} Run Marco : Page d'accueil [en ligne] <https://www.allcancode.com/web> [consulté le 21/04/2016]
- ^{lxxii} Pixees : Jouer à « robot-idiot » pour s'initier aux algorithmes [en ligne] https://pixees.fr/dis-maman-ou-papa-cest-quoi-un-algorithme-dans-ce-monde-numerique-%E2%80%A8/?mediago_cache_id=1899&mediago_engine_id=12290&mediago_ruuid=2213c4f0-a4f2-11e5-af10-f39fac4675bc [consulté le 08/05]
- ^{lxxiii} Canopé : Prêt de matériels [en ligne] <http://www.cndp.fr/crdp-besancon/index.php?id=materiels> [consulté le 10/05/2016]
- ^{lxxiv} Lightbot : Version Flash pour ordinateurs [en ligne] <http://www.lightbot.com/hocflash.html> [consulté le 10/05/2016]

Résumé

Ce document présente comment nous pouvons envisager l'introduction des sciences de l'informatique dès l'école primaire. Les programmes 2016 de l'école primaire et le Plan Numérique 2015 introduisent cela. Il est donc nécessaire de réfléchir à sa mise en œuvre. Pour faire cela, nous allons analyser les initiatives passées françaises. Ensuite, nous nous intéresserons aux initiatives actuelles, à l'internationale et en France. Enfin, nous proposerons un plan d'intervention que l'on testera dans plusieurs classes. Ce plan d'intervention, basé sur les programmes 2016, comprendra des activités de représentation des données et de nombres en binaire et deux activités mettant en place des algorithmes, à l'aide de tris et d'une application d'apprentissage de la programmation.

Mots-clés : sciences de l'informatique, représentation de données et de nombres, tri, application

Abstract

This paper presents how we can imagine to introduce computer sciences at primary school. The 2016 curriculum for primary schools and the "*Plan numérique*" introduce this. We have to think about his implementation. To do this, we will analyse the past French actions. Then, our focus will be on current actions in France and abroad. Finally, we will provide an action plan that we will try on different classes. This action plan, based on 2016 curriculum, will have data and binary representations activities and two activities of algorithmics, with sorts and a learning application.

Key-words: computer sciences, data and number representation, sort, application