



HAL
open science

La conscientisation de la transversalité des compétences travaillées en programmation

Solène Nail

► **To cite this version:**

Solène Nail. La conscientisation de la transversalité des compétences travaillées en programmation. Education. 2019. dumas-02308219

HAL Id: dumas-02308219

<https://dumas.ccsd.cnrs.fr/dumas-02308219>

Submitted on 8 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Master MEEF

« Métiers de l'Enseignement, de l'Éducation et de la Formation »

Mention premier degré

Mémoire

La conscientisation de la transversalité des compétences travaillées en programmation

Mémoire présenté en vue de l'obtention du grade de master

soutenu par

Solène Nail

le mardi 7 mai 2019

en présence de la commission de soutenance composée de :

Cendrine Mercier, directrice de mémoire

Céline Millet, PEMF membre de la commission

Résumé

Dans les nouveaux programmes de 206, l'initiation à la programmation est intégrée au champ disciplinaire des mathématiques dans le domaine « Espace et Géométrie ». Toutefois, les situations indiquées pour enseigner la programmation aux élèves sont peu nombreuses. Il existe un décalage entre la récente importance donnée à la programmation à l'école et la réalité dans les pratiques enseignantes. Nous sommes intéressés aux compétences développées chez les élèves lors des séances de programmation. Nous avons souhaités savoir si les élèves sont en mesure de conscientiser ces compétences et leur transversalité. À l'occasion d'un projet départemental longitudinal de programmation, nous avons mené des entretiens, à deux moments différents de l'année scolaire, auprès de quatre élèves aux profils scolaires et aux affinités avec la programmation différents pour analyser leur conscientisation de la transposabilité des compétences.

Mots clés : programmation – transversalité – conscientisation

In the new 206 programs, programming initiation is integrated into the subject area of mathematics in "Space and Geometry". However, there are few situations indicated to teach programming to students. There is a gap between the recent emphasis on school programming and the reality in teaching practices. We are interested in the skills developed by the students during the programming sessions. We wanted to know if the students were able to raise awareness of these skills and their cross-functionality. As part of a longitudinal departmental programming project, we conducted interviews, at two different times during the school year, with four students with different school profiles and affinities with programming to analyze their awareness of the transposability of skills.

Keywords: programming - cross-functionality - awareness raising

**Engagement de non
plagiat**

Je, soussignée Solène Nail

professeure stagiaire en MEEF à l'ESPE Académie de Nantes

- déclare avoir pris connaissance de la charte anti-plagiat de l'Université de Nantes,
- déclare être pleinement consciente que le plagiat de documents ou d'une partie d'un document publié sur toutes formes de support, y compris l'internet, constitue une violation des droits d'auteur ainsi qu'une fraude caractérisée.

En conséquence, je m'engage à citer toutes les sources que j'ai utilisées pour rédiger ce mémoire.

Date :

Signature :

TABLE DES MATIERES

INTRODUCTION	5
CADRE THEORIQUE	6
I. La programmation au cycle 3.....	6
A. Les instructions officielles et la programmation	6
B. Quels outils pour programmer ?.....	9
II. Transversalité des compétences en programmation	14
A. Apprentissages transposables en programmation	14
B. Compétences travaillées en programmation transposables	15
C. Conscientisation de cette transversalité	16
PROBLEMATIQUE	17
METHODOLOGIE	21
I – Cadre de la recherche.....	21
A. Population.....	21
B. Défi 2+4h kids	22
C. Transposition	26
II – Protocole de recueil de données.....	28
A. Cadre des entretiens	29
B. Sélection des participants	30
RESULTATS	32
I. Conscientisation des compétences de programmation	32
A. La verbalisation des compétences	32
B. La programmation en tant qu’enseignement.....	34
II. Transversalité de ces compétences.....	35
A. Compétences réinvesties en programmation	35
B. Réinvestissement des compétences de programmation.....	36
C. Prendre conscience d’une possible transposition	39
DISCUSSION	42
I. Interprétation et analyse des résultats.....	42
A. Transposition.....	42
B. Conscientisation	45
C. Verbalisation	46
II. Limites de la recherche.....	47
CONCLUSION	49
BIBLIOGRAPHIE	50
ANNEXES	52

INTRODUCTION

Les programmes de 2016 mettent l'accent sur un éveil et une initiation à la programmation durant la scolarité des élèves à l'école primaire. Dès le CE1 les élèves peuvent coder des déplacements à l'aide de logiciels de programmation pour que la notion d'algorithme soit assimilée à la fin du CE2. Au cycle 3, la programmation s'intègre au champ « Espace et géométrie » des mathématiques pour travailler la compétence suivante : « *(se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations* ». Les instructions officielles insistent sur la nécessité d'enseigner la programmation, qui s'inscrit dans tous les domaines du socle commun de compétences et de culture. La programmation permet de faire passer les élèves du statut de consommateurs du numérique au statut de créateurs avec le numérique. De nombreux outils éducatifs sont développés depuis quelques années pour permettre aux enseignants d'intégrer l'initiation à la programmation à leurs pratiques.

La programmation est au service des compétences du socle commun, mais également de compétences de plusieurs disciplines, ce que nous développerons dans le cadre théorique. Le travail de recherche qui suit s'est penché sur la transversalité des compétences travaillées en programmation, sur la façon dont les élèves sont capables de transposer les compétences acquises lors de séances de programmation sur des situations d'apprentissages d'autres disciplines. Nous nous sommes ensuite questionnés sur la conscientisation de cette transversalité. Nous formulons l'hypothèse que les élèves sont capables de comprendre et verbaliser leurs capacités à transposer ce qui est appris lors des séances d'un projet longitudinal de programmation sur des situations ne relevant pas de cet enseignement.

Nous vous présenterons dans un premier temps le cadre théorique sur lequel nous nous sommes appuyés pour formuler la problématique de cette recherche. Nous développerons ensuite le cadre dans lequel ce travail de recherche a été effectué et la méthodologie utilisée pour recueillir les données utiles pour répondre à notre hypothèse. Enfin, nous analyserons les résultats obtenus au vu des données recueillies et nous intéresserons aux limites de cette recherche.

CADRE THEORIQUE

I. La programmation au cycle 3

La programmation est un enseignement obligatoire de l'école élémentaire puisqu'elle figure dans les programmes depuis 2016. La société étant de plus en plus informatisée, depuis les années 1980 des chercheurs tentent d'ancrer dans le système éducatif l'importance d'éduquer les élèves, dès le plus jeune âge, à la logique informatique. C'est le cas de Seymour Papert¹ (1981), qui a eu une grande influence sur d'autres chercheurs dans les domaines de l'éducation et de l'informatique. Il définit la logique informatique comme un « *ensemble d'attitudes et d'acquis universellement applicables que tous, et pas seulement les informaticiens, devraient apprendre et maîtriser.* » De nombreux pays ont été influencés par ces recherches en intégrant davantage le numérique dans l'univers scolaire, pour finalement très récemment y intégrer la programmation et l'apprentissage du code dans les instructions officielles, comme les Etats-Unis, l'Angleterre, la France, le Canada et la Suède.

A. Les instructions officielles et la programmation

1. Une place importante en « Espace et Géométrie »

Dans le programme actuel de cycle 3, l'enseignement de la programmation est développé dans un court paragraphe dans le domaine « Espace et Géométrie », en mathématiques. Il est bien précisé qu'il s'agit d'une initiation, que l'on mène à travers des activités de déplacements simples de robots dans l'espace scolaire ou de personnages sur des écrans, ou d'activités géométriques avec par exemple des programmes de construction de figures géométriques simples ou complexes connues des élèves. L'idée d'initiation est renforcée par le repère de progressivité suivant dans les programmes « *au CM1, on réserve l'usage de logiciels de géométrie dynamique à des fins d'apprentissage manipulatoires* ». Les logiciels propices aux situations

¹ Seymour Papert (1928-2016) : mathématicien, informaticien et éducateur au Massachusetts Institute of Technology

d'apprentissage en géométrie, GeoGebra² par exemple, doivent alors être utilisés pour illustrer des notions que les élèves sont amenés à manipuler par la suite hors logiciel, comme la conservation des propriétés de figures géométriques ou de droites. En suivant les repères de progressivité, c'est à partir du CM2 que les fonctionnalités permettant la construction et la déconstruction seront utilisées par les élèves.

La programmation, sans être nommée, participe cependant à part entière dans l'acquisition de la compétence « *(se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations* ». Bien qu'il soit ensuite indiqué qu'il s'agit d'une simple initiation, son usage semble implicitement être très encouragé pour construire des situations d'apprentissage permettant d'acquérir la compétence citée ci-dessus. Les élèves doivent savoir décrire ou exécuter des déplacements sur un plan ou sur une carte, accomplir et coder des déplacements dans des espaces familiers, et programmer les déplacements d'un robot ou d'un personnage sur un écran. Les enseignants peuvent alors se diriger vers des éléments du quotidien : travailler dans des espaces de travail de taille différente mais trouvés dans l'environnement proche de l'élève ou partir de plans schématiques auxquels les élèves sont ou seront également confrontés. Il est ensuite indiqué, en dernier, qu'il faut également travailler « *avec de nouvelles ressources comme les systèmes d'information géographique, des logiciels d'initiation à la programmation [...]* ». La programmation semble donc être un des moyens permettant d'acquérir cette compétence, pas le moyen obligatoire, mais son enseignement en tant qu'initiation est obligatoire. On peut ainsi se demander quelles compétences sont alors travaillées en programmation, autres que le repérage et le déplacement dans l'espace.

2. Les compétences travaillées en programmation

L'initiation à la programmation permet, si elle est menée régulièrement avec les élèves, de travailler des compétences de chaque domaine du socle commun de connaissances, de compétences et de culture : « Les langages pour penser et communiquer », « Les méthodes et outils pour apprendre », « La formation de la personne et du citoyen », « Les systèmes naturels et les systèmes techniques » et « Les représentations du monde et de l'activité humaine ».

² Logiciel gratuit

La programmation permet en effet de pratiquer des langages spécifiques et des langages adaptés à des situations. C'est un domaine qui nécessite un lexique spécifique, parfois difficile à assimiler dès le CM1. Les élèves sont d'abord sensibilisés par l'enseignant qui se doit d'utiliser, à l'oral puis à l'écrit, le lexique adapté pour chaque élément de programmation. Les élèves peuvent ensuite progressivement emprunter ce bagage lexical qui leur est de plus en plus familier. Le langage est ensuite un élément clé pour rendre compte de ce qui est fait en programmation, les phénomènes vécus en situation doivent pouvoir être expliqués l'oral – ou à l'écrit, pour pouvoir débloquer certaines situations ayant mis les élèves en difficulté ou au contraire pour partager les réussites réalisées.

En outre, la programmation permet également aux élèves de s'emparer de méthodes et d'outils nouveaux. La connaissance et la maîtrise des outils numériques se construisent en technologie, notamment avec l'appropriation d'un environnement numérique propre à chaque élève, puis à la classe, mais aussi en utilisant différents logiciels de traitement de données numériques. Les outils numériques sont également à intégrer dans différentes disciplines, telles que les mathématiques comme cela a été précédemment évoqué, en français pour l'utilisation des outils numériques d'écriture, en langues vivantes ou encore dans le domaine des arts à travers des logiciels de 4D par exemple. Les élèves apprennent aussi, à travers la programmation, à coopérer et collaborer dans l'élaboration de projets de groupe.

L'enseignement de la programmation permet de développer chez les élèves des compétences précises et pourtant transposables dans différentes situations, telles que³ :

- L'investigation : les élèves mènent une activité avec des temps d'observation et de réflexion, ce qui répond à une démarche scientifique.
- La décomposition : les élèves sont d'abord confrontés à un problème complexe, qu'ils découpent en sous-problèmes, en sous-tâches selon les stratégies qu'ils décident d'adopter.
- La reconnaissance de schéma : les élèves reconnaissent des schémas, des configurations, des invariants, des répétitions et mettent en évidence des interactions.

³ Source : Lazzarotto, F. et Cosson, M. La programmation dans les nouveaux programmes

- La généralisation et l'abstraction : les élèves doivent réussir à repérer les enchaînements logiques et les traduire en instructions conditionnelles, traduire les schémas récurrents en boucles, concevoir des méthodes liées à des objets qui traduisent le comportement attendu.
- La conception algorithmique : les élèves savent comment écrire des solutions à un problème donné, réutiliser des algorithmes déjà programmés, programmer des instructions déclenchées par des événements, concevoir des algorithmes.

L'enseignement de la programmation permet d'accompagner différemment certains principes mathématiques, mais participe également à varier les modalités de résolution de problèmes.

La programmation est donc établie comme un enseignement obligatoire dans les instructions officielles, à l'état d'initiation, mais permet de travailler de nombreuses compétences si elle est pratiquée davantage dans les classes. Cet enseignement est mis en avant en France car de nombreuses études ont montré l'importance de l'enseignement de la logique informatique pour la formation des futurs citoyens. En effet, cela permet aux élèves de mieux comprendre le monde technologique dans lequel ils évolueront, mais aussi d'être préparés aux nombreux métiers actuels et futurs qui continueront d'être influencés par le numérique.

B. Quels outils pour programmer ?

La programmation peut être enseignée sous diverses formes et à l'aide de nombreux outils. Il s'agit d'un domaine qui se développe énormément ces dernières années, en démontrent les nouveaux matériaux de programmation et de robotique programmée commercialisés et accessibles à tous. Cependant, la programmation ne se fait pas systématiquement par le biais d'ordinateur, tablette ou robot. On parle alors d'activités débranchées, lorsqu'il n'y a pas d'utilisation de dispositif informatique, ou de travail branché⁴. Les activités débranchées sont très variées (labyrinthe, rôle de robot, codage de déplacements...), nous allons cependant nous intéresser aux

⁴ Source : <https://canope.ac-besancon.fr/>

logiciels et aux robots pouvant être utilisés en cycle 3 lors d'activités de programmation branchées.

1. Les logiciels d'apprentissage de programmation

Tout d'abord, de multiples outils informatiques peuvent être utilisés en programmation, comme les ordinateurs portables, les ordinateurs fixes, les tablettes... Ensuite, les outils de programmation sont très nombreux. Il existe plusieurs manuels, sites, défis, concours, projets accessibles à tous les enseignants, dans chaque académie. Cependant, peu sont utilisés massivement, puisque les nombres de participants, d'inscrits ou les taux d'engagement sont faibles. Les enseignants semblent souvent préférer se lancer dans des projets recommandés par l'académie dans laquelle ils exercent. Cela permet d'avoir des contacts plus faciles avec les référents des projets. Les outils adaptés sont de plus en plus nombreux et peuvent être intégrés progressivement dans le cursus scolaire des élèves. Il existe des manuels, comme « 1, 2, 3 Codez⁵ » de la fondation *La main à la pâte*, qui permet d'enseigner la programmation à tous les niveaux avec des séances clés en main, ou des concours qui permettent également de travailler en programmation tout en suivant un parcours déjà établi (« concours Castor⁶ », du CM1 à la terminale).

Les élèves qui utilisent des logiciels d'apprentissage de la programmation comprennent davantage les concepts et la logique de programmation après leur utilisation⁷, il s'agit de logiciels simples et adaptés à l'âge des élèves. Scratch, logiciel gratuit, a par exemple une extension pour les plus jeunes, Scratch Junior, un logiciel de programmation visuel permettant d'introduire des compétences de codage dès l'âge de 5 ans. Ces logiciels adaptés permettent l'engagement des élèves dans les tâches demandées, ils sont davantage motivés et prennent du plaisir à progresser en programmation.

⁵ <http://www.fondation-lamap.org/123codez>

⁶ <https://concours.castor-informatique.fr/>

⁷ Source : Moreno-León, J., Robles, G. et Román-González, M. Code to learn: Where does it belong in the K-12 curriculum?

nécessite davantage d'opérations mentales mais permet de se rapprocher des possibilités de codage de Scratch.

Scratch est quant à lui le logiciel le plus répandu dans l'univers scolaire. Eduscol, le portail national de ressources des professionnels de l'éducation, propose de nombreux guides pour encourager et aider les enseignants à utiliser Scratch en classe. La réforme française du collège de 2015 insiste également sur l'utilisation de ce logiciel pour aborder les notions de codage et d'algorithme, des questions portant sur des algorithmes réalisés sur Scratch peuvent d'ailleurs être posées au brevet des collèges. Scratch a été développé en 2006 par le MIT et connaît un succès dans de nombreux pays depuis une dizaine d'années. Les élèves peuvent créer et modifier des programmes informatiques en déplaçant des blocs de couleurs et en les organisant selon leurs besoins. Les élèves peuvent y créer des jeux vidéo, des histoires interactives et donner vie à des scénarios déterminés à l'avance. Scratch permet aux élèves de devenir des auteurs et acteurs de leurs histoires, et donc de leurs apprentissages¹².

2. Des robots pour programmer

Il existe également de nombreux robots destinés aux enfants. Ces robots sont de plus en plus commercialisés et ont beaucoup de succès. Cependant, il y a souvent une confusion entre commander et programmer un robot. La plupart des robots commercialisés, et donc que possèdent les élèves chez eux, sont à commander. Souvent à l'aide d'une manette, ou bien simplement avec des touches disposées sur le robot, les enfants peuvent commander à distance ou non les déplacements et actions de leur robot. Cependant, les robots utilisés dans le milieu scolaire, dès le cycle 1, sont à programmer, pour qu'il n'y ait pas de confusion plus tard entre commander et programmer. La programmation d'un robot inclut un temps de codage de ses futurs déplacements et un temps d'exécution des déplacements codés préalablement. La programmation nécessite une anticipation des déplacements à effectuer pour se rendre à un endroit ou pour éviter des obstacles.

¹² Source : Thierry Karsenti, Julien Bugmann, Pourquoi apprendre à coder à l'école ?

Il existe ainsi des robots pouvant être programmés de manière intuitive et pédagogique. L'un des premiers robots à programmer à avoir eu du succès dans le milieu scolaire est le robot américain Dash, programmé grâce à des applications.



*Robot Dash*¹³

Aujourd'hui, de nombreux robots semblables à Dash viennent soutenir l'enseignement de la programmation pour donner davantage de réalisme à l'apprentissage du code¹⁴. Beaucoup sont financièrement abordables et de plus en plus présents dans les écoles, comme les robots Beebot, Mbot, Ozobot, Thymio, ou encore des robots pédagogiques humanoïdes, comme le robot canadien NAO. Il peut être intéressant de lier ces robots aux projets menés sur Scratch.



*Beebot*¹⁵



*Mbot*¹⁶



*Ozobot*¹⁷



*Thymio*¹⁸



*NAO*¹⁹

¹³ Source de l'image : <https://www.generationrobots.com>

¹⁴ Source : Thierry Karsenti, Julien Bugmann, Pourquoi apprendre à coder à l'école ?

¹⁵ Source de l'image: <https://www.generationrobots.com>

¹⁶ *Ibid.*

¹⁷ *Ibid.*

¹⁸ Source de l'image : <https://www.my-etechno.com>

¹⁹ Source de l'image : <http://eduscol.education.fr/>

Les outils pour créer des situations d'apprentissage en programmation sont donc nombreux et variés. Tous ces outils permettent aux enseignants de créer des projets conséquents avec les élèves. Les séances débranchées peuvent être utiles pour commencer, pour acquérir une bonne compréhension des principes de base de la programmation. Cependant dans un second temps, il est possible de procéder par pédagogie de projet en faisant appel à tous les projets déjà créés et proposés au sein ou à l'extérieur des académies. Ces projets sont souvent interdisciplinaires, liant une ou plusieurs disciplines à la programmation. Au-delà de la transdisciplinarité des projets possibles, la programmation fait elle-même appel à des compétences transdisciplinaires chez les élèves et crée de nouvelles connaissances transposables dans d'autres disciplines.

II. Transversalité des compétences en programmation

La plupart des compétences acquises à l'école sont transversales, puisque réutilisées dans des situations relevant d'autres disciplines. Certaines compétences sont acquises dans telle matière et sont réutilisables dans telle autre, mais l'inverse peut également être possible. Dans le développement de cette seconde partie sur la transversalité des compétences, nous découvrirons que l'acquisition de compétences transversales peut être faite en programmation ou dans une autre discipline.

A. Apprentissages transposables en programmation

Comme cela a été développé dans la première partie, la programmation fait appel à des compétences de français, notamment le langage écrit pour réaliser le codage, et le langage oral pour progresser en verbalisant les procédures, en nommant les difficultés et les réussites. En mathématiques, les apprentissages de repérage dans l'espace sont utiles pour avoir une meilleure appréhension lors de l'entrée dans l'enseignement de la programmation (quadrillage, plan, déplacement dans l'espace, description de parcours...). Les notions de géométrie facilitent l'utilisation de logiciels

tels qu'AlgoBlocs²⁰ qui permettent de créer des figures géométriques en codant leurs tracés, ce qui nécessite une connaissance préalable des propriétés. Des compétences de numération sont également engagées, pour se repérer plus facilement dans ces logiciels incluant beaucoup de données numériques. Enfin, la notion d'angle abordée en Grandeurs et Mesures est également importante sur l'interface Scratch, puisque les rotations des personnages sont exprimées en degrés dans les blocs de déplacement.

B. Compétences travaillées en programmation transposables

La programmation est un enseignement riche et varié qui permet de travailler différents domaines. Tout d'abord, le travail effectué avec les élèves en programmation permet de mieux illustrer certaines notions de technologie. Dans le thème « *Matériaux et objets techniques* », les élèves doivent approfondir leur connaissance de l'environnement numérique de travail, en comprenant les notions d'algorithme et d'objets programmables, ce qui est facilité par les manipulations faites en programmation. Dans un autre thème de technologie, intitulé « *Signal et information* », les élèves doivent savoir identifier un signal et une information en découvrant plusieurs types de codage d'information. Les activités faites en programmation peuvent déjà leur donner des exemples concrets d'application de code pour une utilité précise, ici le codage de mouvement pour animer un personnage sur un écran ou un robot. Le code peut être réutilisable dans de nombreuses situations scolaires, mais aussi dans la vie quotidienne de l'élève. Les algorithmes peuvent également être pris comme modèle pour mettre en œuvre des algorithmes de calcul posé pour les différentes opérations apprises à l'école élémentaire²¹. La programmation, toujours grâce à la notion algorithmique, a un rôle essentiel pour la logique de l'élève et sa mise en œuvre dans différentes situations. Pour Piaget (1981), l'algorithme est un outil constitutif de la logique, c'est un langage précis et un outil d'analyse et de codage qui permet de travailler ses compétences d'abstraction et de cohérence dans des situations problématiques. La programmation a, grâce à ce développement de l'esprit logique chez les élèves, des bénéfices en mathématiques, principalement dans la résolution

²⁰ Logiciel gratuit

²¹ Source : <https://pedagogie.ac-guadeloupe.fr/>

de problèmes. Le codage et l'algorithme peuvent également être des connaissances utiles en éducation musicale, pour aborder les notions de rythmes, de boucles, de carrures, ce que l'on peut observer en mettant en lien des notes combinées ensemble et des algorithmes.

Tout ce que les élèves apprennent en programmation peut probablement être transposable dans toutes les autres disciplines, la programmation n'est d'ailleurs pas une discipline enseignée comme telle. Il s'agit donc bien d'un outil pour réinvestir des compétences relevant des disciplines scolaires, ou au contraire de développer des compétences nouvelles.

C. Conscientisation de cette transversalité

Pour qu'un élève soit conscient des compétences qu'il acquière, il faut qu'il soit entièrement acteur de ses apprentissages. Les enseignants doivent essayer de rendre les élèves conscients de leur propre processus cognitif, c'est-à-dire d'avoir un recul sur ce qu'il sait, en être conscient. L'enseignant cherche à développer sa capacité de gérer de manière autonome les différentes fonctions métacognitives : savoir comment on mémorise, comment on peut résoudre un problème, comment on contrôle ses émotions²²... Une fois que l'élève sait quelles compétences lui sont acquises, il peut faire appel à ce savoir pour déterminer quelles compétences utiliser dans différentes situations. L'enseignant peut participer à cette conscientisation des savoirs en explicitant bien avec les élèves tout ce qui est appris au cours d'une activité, d'une séance, d'une séquence... Les compétences doivent être explicitées par les élèves eux-mêmes, il faut qu'il y ait une verbalisation. Vergnaud (2011) montre l'importance de la mise en mots pour les élèves. C'est en mettant des mots sur des compétences que les élèves peuvent conscientiser ce qu'ils savent. Ainsi, les élèves auront davantage de facilité à réinvestir leurs savoirs.

²² Source : Gagné, Leblanc et Rousseau, *Apprendre, une question de stratégies*

PROBLEMATIQUE

Au vu des éléments apportés dans le cadre théorique, nous nous sommes donc questionnés sur comment vérifier la conscientisation des compétences travaillées en programmation et de leur transversalité par les élèves.

La programmation est un enseignement obligatoire de l'école primaire mais elle s'inscrit de façon peu lisible dans un champ disciplinaire défini. Les programmes officiels inscrivent cet enseignement dans le thème « Espace et Géométrie », et en technologie, mais développent très peu les exemples de pratiques en classe. Cependant, c'est un enseignement qui se développe et paraît être de plus en plus présent dans les pratiques professionnelles puisque des formations pour les professeurs des écoles sont menées, des initiations pour les futurs professeurs des écoles dans les ESPE sont menées également, des intervenants spécialisés pour certains projets de programmation sont appelés dans les écoles, des ouvrages pédagogiques et articles sur la programmation commencent à être de plus en plus disponibles et connus, les outils pratiques et le matériel pour mettre en œuvre en classe cet enseignement se développent également énormément... Nous sommes donc face à un enseignement, la programmation, ayant un cadre institutionnel concernant les attendus et compétences peu défini, qui toutefois semble progresser dans les pratiques enseignantes et tend à l'être de plus en plus dans les années à venir.

Nous avons questionné une vingtaine d'enseignants²³ sur leur façon d'aborder la programmation avec les élèves de cycle 3. Une grande majorité de ces enseignants incluent l'enseignement de la programmation dans leur programmation de géométrie, souvent en fin d'année, durant quelques séances branchées ou débranchées. Plusieurs enseignants ont admis ne pas faire de programmation dans l'année avec leurs élèves, en estimant ne pas avoir le temps, le matériel, les compétences ou les ressources nécessaires. Trois enseignants font de la programmation avec les élèves de façon détachée des autres matières, avec des activités branchées dédiées à cet

²³ Professeurs des écoles enseignant en Sarthe ou en Mayenne ayant ou ayant eu des élèves de cycle 3

enseignement. Lorsque nous leur avons demandé sur quel temps disciplinaire était décompté le temps passé sur ces activités de programmation, ils ont admis ne pas ancrer ce temps dans une discipline précise car il s'agit pour eux d'une initiation, mais qu'ils pourraient justifier ce temps en l'associant à des compétences présentes en « Espace et géométrie ».

Ces échanges avec les enseignants nous ont permis de constater que la programmation peut être un outil au service de compétences mathématiques, mais il s'agit d'un outil qui peut prendre beaucoup de temps. En effet, la familiarisation des élèves, dans un premier temps, aux ordinateurs peut parfois se faire très progressivement dans certains cas où les pratiques des écoles n'y sont pas habituées. Dans un second temps, la familiarisation au principe même de programmation peut être compliquée pour certains élèves. Il s'agit d'un enseignement qui requière des connaissances lexicales précises (bloc, boucle, code, lutin²⁴, script...), des compétences complexes (anticipation, repérage dans l'espace, stratégie...) et une aisance avec l'espace numérique. Les logiciels de programmation sont parfois très riches, à l'instar de Scratch qui présente de nombreuses informations dès l'arrivée sur l'interface initiale, et énormément de possibilités d'action que les élèves découvrent petit à petit. Cependant, il est nécessaire que plusieurs temps d'explication soient faits progressivement avec les élèves sur toutes les possibilités que proposent Scratch pour créer des programmes (lutins, évènements, contrôles, déplacements, sons, paroles, arrière-plans...). Durant notre travail de recherche, nous avons remarqué que les élèves jouant aux jeux vidéo chez eux étaient remarquablement à l'aise avec toutes ces possibilités, l'interface semblait leur être plus ergonomique, ils avaient davantage de facilité à naviguer entre tous les onglets des propositions d'actions. Cependant, pour la plupart des élèves, plusieurs séances ont été nécessaires pour découvrir la quasi-intégralité des possibilités de programmation sur Scratch. De ce fait, il est possible qu'une séquence de programmation de quelques séances ne suffise pas aux enseignants pour explorer avec les élèves toutes les offres de Scratch ou d'un autre logiciel. Il existe toutefois plusieurs projets à disposition des professeurs des écoles pour enseigner la programmation avec des outils clés en main. Nous avons choisi d'inscrire notre classe, sur laquelle sera menée l'étude de ce mémoire, à un projet de

²⁴ Terme pour désigner les personnages ou objets programmables sur Scratch

programmation se tenant sur sept mois. Ce projet est conséquent en termes de temps, puisqu'il dure plusieurs mois et nécessite approximativement une séance de quarante-cinq minutes par semaine. Il s'agit donc d'un projet longitudinal qui permet une bonne familiarisation d'un logiciel de programmation pour ensuite développer plusieurs compétences chez les élèves.

La question des compétences peut ainsi être entièrement mise au cœur du questionnement professionnel de l'enseignant, puisque le projet offre des séances clés en main et des objectifs déjà prédéfinis. À travers cette recherche, nous nous intéresserons donc aux compétences travaillées en programmation, mais surtout à la conscientisation de ces dernières par les élèves. Nous souhaiterions savoir ce que les élèves connaissent de leur propre processus cognitif concernant les compétences travaillées dans le cadre des séances de programmation. En outre, nous nous questionnons également sur la capacité des élèves à être conscients que la programmation nécessite l'acquisition de compétences propres à d'autres disciplines, telles que le français ou les mathématiques par exemple. Nous questionnons ainsi les capacités des élèves à conscientiser et transposer leurs compétences. Nous allons donc tenter, à travers cette étude, de répondre à la problématique suivante :

De quelles façons, dans le cadre d'un projet de programmation, rendre conscients les élèves des compétences travaillées et développées et de leur transversalité ?

Tout d'abord, nous souhaitons savoir si les élèves perçoivent que la programmation est un enseignement comme les autres, c'est-à-dire qu'elle est au service d'apprentissages. Les élèves peuvent avoir tendance à considérer le temps de programmation comme extérieur au temps considéré « de travail » de par ses caractéristiques propres. Il apparaît généralement que ce soit un temps sur ordinateur pour la plupart des séances, souvent en groupe ou en binôme, avec des activités ludiques. Tous ces paramètres peuvent obstruer l'idée que la programmation reste un temps d'apprentissage. Au-delà de la considération de cet enseignement, il s'agit ensuite de prendre conscience des compétences en jeu dans ces activités. Pour évaluer cette conscientisation, nous devons également demander aux élèves de verbaliser ; verbaliser sur ce qu'est la programmation, sur ce que cela leur apprend,

sur ce qu'ils savent faire et connaissent... Se pose alors la question de la verbalisation, certains élèves peuvent réussir ce travail de conscientisation des compétences travaillées, mais ne pas parvenir à le verbaliser.

Intervient ensuite la problématique de la transversalité des compétences travaillées en programmation, de la transposition de ce qui est appris dans cet enseignement dans d'autres disciplines. Comment réutiliser des compétences travaillées en programmation ? Comment être conscient de cette réutilisation, cette transposition ? À travers plusieurs disciplines extérieures, nous proposerons aux élèves des situations d'apprentissage faisant appel à des compétences ou connaissances travaillées antérieurement en programmation. Nous cherchons tout d'abord à savoir si les élèves sont capables de voir des similitudes entre des situations de programmation et des situations ayant lieu lors de séances d'autres matières. Il peut s'agir de situations faisant appel au code, à l'algorithme, au programme... Nous voulons ensuite observer la capacité des élèves à réutiliser les compétences travaillées en programmation, et surtout à être conscients qu'ils font appel à ces compétences. Cette transposition peut également être questionnée sur des situations extérieures au temps d'apprentissage, nous analyserons si les élèves peuvent citer quelques situations se rapprochant de ce que l'on fait en programmation et d'expliquer pourquoi.

METHODOLOGIE

I – Cadre de la recherche

A. Population

Nous menons notre étude dans l'école primaire publique de Saint Pavace, village périurbain appartenant à la circonscription de Le Mans-Coulaines. La municipalité est très investie dans la vie scolaire et beaucoup de moyens sont donnés à l'école. Il s'agit d'une école dans laquelle les enseignants travaillent fortement en équipe, avec de nombreux décloisonnements.

Le numérique est au cœur de l'école, la très grande majorité des enseignants y utilisent énormément les outils numériques à disposition. Ces pratiques se sont intensifiées il y a cinq ans avec l'arrivée d'un ERUN (Enseignant Référent pour les Usages du Numérique) présent un ou deux jours par semaine dans l'école qui apporte de nombreux conseils et outils aux enseignants. L'école a eu plusieurs blogs informatiques, et utilise beaucoup l'interface e-primo²⁵, anciennement One. Les enseignants communiquent avec les élèves et les parents dessus, les élèves peuvent également communiquer entre eux et les classes entre elles. Les fonctionnalités pédagogiques y sont nombreuses et sont exploitées dans la majorité des classes (cahiers numériques, blog, pad²⁶, cahier de texte...). Plusieurs outils numériques sont à disposition : malles d'ordinateurs, des micros, un robot Thymio...



Robot Thymio²⁷



Interface e-primo²⁸

²⁵ Espace numérique de travail des écoles de l'académie de Nantes

²⁶ « mur virtuel » collaboratif sur lequel il est possible d'afficher toute sorte de documents

²⁷ Source de l'image : <https://www.thymio.org/>

²⁸ Source de l'image : <https://www.pedagogie.ac-nantes.fr/>

Nous avons récolté les données de cette recherche dans une classe de CM1 composée de 26 élèves, 13 garçons et 13 filles. Deux professeures se partagent la classe à temps égal. Le niveau de la classe est, comme toutes les classes, hétérogène puisqu'il y a des différences de capacités, connaissances, compétences entre chaque élève. Quelques élèves sont en difficulté en termes d'apprentissages, des aménagements au sein de la classe et en décloisonnement sont effectués.

Tous les élèves de cette classe ont eu, hormis une élève arrivée cette année dans l'école, une initiation à la programmation durant leur parcours scolaire. Depuis la grande section, ces élèves de CM1 ont eu des enseignants effectuant des activités de programmation, branchées ou débranchées au sein de leur classe. Les élèves n'étaient pas tous dans la même classe l'année dernière, en classe de CE2. Un tiers des élèves se trouvait dans une classe de CE2-CM1 avec l'enseignante titulaire actuelle et ont eu quelques séances de programmation en fin d'année. Les deux autres tiers des élèves étaient avec deux enseignants très proches du numérique dans leur pratique, avec une forte appropriation de l'espace de travail numérique, des expériences sur plusieurs logiciels (Tuxbot, Géotortue²⁹, Scratch...) de programmation et des activités effectuées avec le robot. Les élèves ont déjà de fortes affinités avec le logiciel de programmation Scratch et connaissent beaucoup de ses fonctionnalités.

B. Défi 2+4h kids

1. Concept

Nous avons inscrit cette classe au défi « 2+4h kids – Chronocode »³⁰ en début d'année. Ce défi est proposé aux classes de CM1-CM2 de la Sarthe depuis trois ans. Il est organisé par la DSDEN de la Sarthe, l'ESPE de l'académie de Nantes et le réseau CANOPE. L'objectif général de ce défi, pour les élèves, est de réaliser « *un support de communication par l'intermédiaire d'un logiciel de programmation, valorisant le patrimoine historique local et rendant compte des apprentissages réalisés en histoire tout au long de la scolarité* »³¹. Il s'agit d'un projet d'initiation à la programmation et à

²⁹ Logiciel gratuit pour enseigner les mathématiques et l'algorithmique

³⁰ <http://www.24hkidsandco.fr/>

³¹ Source : <http://www.24hkidsandco.fr/>

la pensée informatique. Le choix du logiciel de programmation est libre, nous avons choisi d'utiliser Scratch qui était déjà connu des élèves et qui semblait être un logiciel complet et adapté pour effectuer ce défi. Les enseignants sont accompagnés pour mener ce défi en classe par le biais d'animations pédagogiques et d'un espace de mutualisation sur la plateforme de formation continue de l'Education nationale M@gistère³².

Les élèves ont pour défi de créer un support de communication valorisant le patrimoine local à l'aide de Scratch. Cette année, le défi a pour thème : « *Itinéraire d'un poilu durant la Grande Guerre* », les classes devront alors retracer l'itinéraire d'un poilu de leur commune durant la Première Guerre mondiale. Les productions des élèves seront valorisées en étant projetées à l'évènement rencontre 2+4h kids organisé à la CCI du Mans par les organisateurs du défi. Cependant, pour réaliser ce défi final, les élèves vont devoir réaliser en amont différentes missions reçues tout au long de l'année permettant de travailler progressivement différentes compétences en programmation. La production finale fera appel à toutes ces compétences acquises lors des missions.

2. Missions

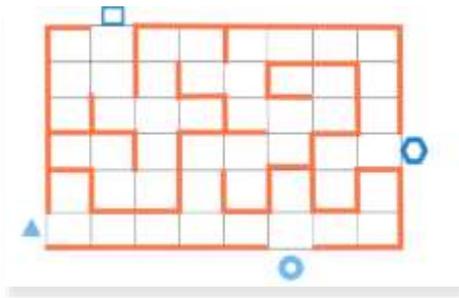
Du mois d'octobre au mois de mars, nous avons mené avec les élèves les cinq missions proposées avant de commencer la préparation du défi final. Les missions doivent être effectuées pendant plusieurs séances.

La première mission s'intitule « Le labyrinthe » et s'organise en trois séances débranchées de quarante-cinq minutes qui visent les objectifs suivants pour les élèves :

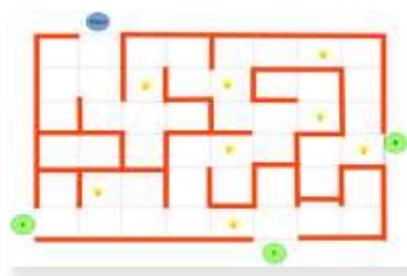
- être capable d'écrire une suite d'instructions (un programme) pour permettre la réalisation d'un parcours par un tiers,
- être capable d'écrire une suite de déplacements, de décisions et d'actions pour permettre la réalisation d'un parcours dans un labyrinthe sur feuille,

³² <https://magistere.education.fr/>

Les trois séances ont sensiblement le même objectif technique pour les élèves, la situation initiale est toujours la même : permettre à la figurine-robot de sortir du labyrinthe, mais la situation se complexifie au cours des séances. Durant la première séance, il s'agit de réussir à faire sortir le robot le plus vite possible, pendant la deuxième séance, de faire sortir le robot en ajoutant des actions spéciales utilisables une seule fois, et lors de la troisième séance de faire sortir le robot en utilisant obligatoirement des sources d'énergies au bout de 6 ordres, instructions.



Labyrinthe des deux premières séances



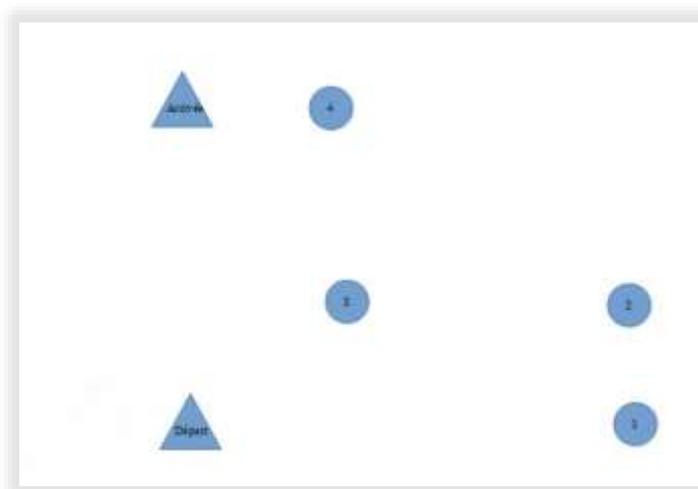
Labyrinthe de la troisième séance

À travers cette première mission composée de trois séances, les élèves ont ainsi pu se remémorer des éléments de programmation et de codage utiles pour les séances branchées :

- Il faut anticiper les moyens les plus courts avant de coder les instructions.
- Pour réduire le nombre d'instructions, il faut utiliser le coefficient multiplicateur lorsque le déplacement est identique plusieurs fois.
- Le programme le plus efficace est le programme le plus court en ligne de code.

Nous avons, lors de la première séance, établi qu'un programme était une suite d'instructions à effectuer.

La seconde mission est cette fois-ci branchée et propose aux élèves d'écrire un programme sur Scratch pour permettre au lutin de réaliser un parcours. Dans la première séance, le parcours est imposé par un arrière-plan fourni avec la mission et pour la deuxième séance ce sont des parcours que les élèves doivent eux-mêmes dessiner et importer en arrière-plan sur Scratch. Cette mission a pour but de remobiliser les connaissances des élèves sur Scratch.



Arrière-plan imposé pour la première séance



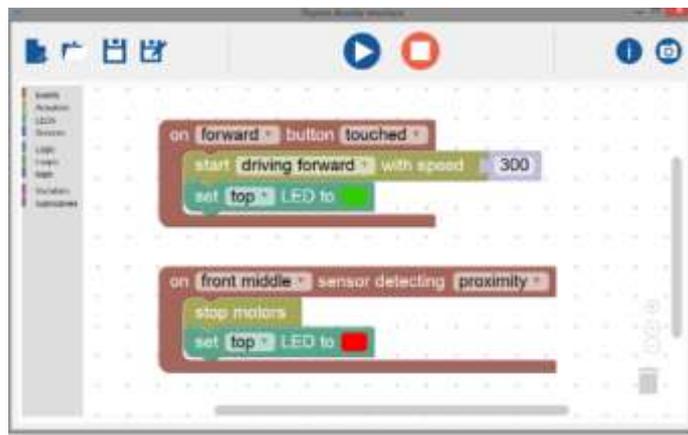
Exemples d'arrière plan réalisés pour la deuxième séance

La troisième mission propose aux élèves de créer une carte d'identité interactive en anglais. Les élèves doivent créer un script permettant de présenter un personnage à travers différents lutins apportant des informations en anglais lorsque l'on clique dessus. Cette mission permet aux élèves d'apprendre à associer des lutins à des enregistrements audio et simultanément à des bulles écrites.

La quatrième mission propose quant à elle de réaliser un scénario mettant en scène la visite d'un musée avec la présentation successive de deux œuvres vues en classe. A la fin des deux séances, les élèves doivent être capables d'écrire un programme pour permettre d'associer plusieurs lutins, enregistrements audio, bulles de conversation, arrière plan et costumes sous la forme d'un scénario en continu dès le programme lancé. Cette mission permet d'aborder plus précisément le projet final avec les élèves puisqu'ils devront également réaliser un scénario continu.

Enfin, la dernière mission s'effectue cette fois-ci avec le robot Thymio. Nous avons ainsi pu apprendre aux élèves qu'il était possible de programmer le robot avec

son propre logiciel de programmation, Blockly³³, puisqu'ils ne l'avaient utilisé qu'avec ses propres fonctionnalités de commande durant les années scolaires précédentes. À travers cette mission, les élèves s'approprient l'interface Blockly permettant de créer des programmes pour le robot. Ils apprennent à programmer un robot afin de lui permettre de réaliser un parcours.



Interface Blockly

Nous avons effectué toutes ces missions pendant plusieurs mois à raison d'une séance de quarante-cinq minutes par semaine. Nous prenions cinq minutes après chaque séance pour verbaliser ce que les élèves pensaient avoir appris. Nous posions la question suivante : « *Qu'avez-vous appris aujourd'hui durant cette séance de programmation ?* », cela a permis aux élèves une verbalisation des compétences techniques de programmation.

C. Transposition

Pour répondre à notre questionnement concernant la transposabilité des compétences travaillées en programmation et sa conscientisation, nous avons proposé des situations d'apprentissage réinvestissant des compétences techniques propres à la programmation. Ces situations d'apprentissage ont été menées à différents moments de l'année.

³³ <https://www.thymio.org/fr:blocklyprogramming>

1. Education musicale

Durant la première période scolaire, nous avons mené une séquence autour du rythme musical avec l'apprentissage d'une « *cup song* »³⁴, un rythme effectué avec les mains et un gobelet, et un chant l'accompagnant. Nous avons décidé de mêler cette notion de rythme au codage pour la deuxième période scolaire.

Nous avons tout d'abord demandé aux élèves, par groupe, d'inventer un morceau rythmique avec les supports qu'ils souhaitaient. Les élèves travaillaient, à travers ce temps de recherche, des compétences propres à l'éducation musicale telles que :

Echanger, partager et argumenter

- ✓ Ecouter et respecter le point de vue des autres et l'expression de leur sensibilité
- ✓ Argumenter un choix dans la perspective d'une interprétation collective

Explorer, imaginer, créer

- ✓ Expérimenter les paramètres du son et en imaginer des utilisations possibles
- ✓ Inventer une organisation simple à partir de sources sonores sélectionnées

Nous avons par la suite souhaité développer la compétence suivante « *Imaginer des représentations graphiques pour organiser une succession de sons* » en demandant aux élèves de représenter à l'écrit leur rythme inventé, comme une partition, pour que d'autres groupes puissent l'interpréter, le reproduire. Par cette situation, nous nous sommes intéressés aux capacités des élèves à réutiliser ce que nous avons fait en programmation quelques semaines auparavant lors de la première mission du défi, la mission débranchée avec le labyrinthe. Les productions des huit groupes d'élèves ont été très variées (annexe 2). Quatre groupes ont réutilisé la notion de coefficient multiplicateur lorsqu'une même action était répétée plusieurs fois. Un groupe a utilisé une légende pour que sa production soit reproductible, que le code utilisé soit compris des autres. Quatre groupes ont utilisé des symboles pour coder leur rythme plutôt que des phrases écrites, et trois groupes ont associé des symboles à quelques mots écrits. À la fin de la séance, nous avons consacré cinq minutes pour verbaliser ce que nous avons fait, et plusieurs élèves ont relevé les ressemblances entre la tâche demandée aujourd'hui et la tâche effectuée en programmation lors des séances débranchées.

³⁴ Traduction : chanson au gobelet

2. Technologie

Nous avons mené une séquence de technologie sur le thème « Signal et information » durant la troisième période scolaire. Nous avons défini avec les élèves le signal comme étant un message codé permettant la communication. Les codes communs de communication de la classe étant l'alphabet et la langue française. Dès la première séance, les élèves étaient à l'aise avec la notion de signal et de codage d'informations, certains ont d'ailleurs fait remarquer spontanément le lien avec la programmation à l'oral. Nous avons proposé des situations de codage et d'encodage d'informations (annexe 3) en nous appuyant successivement sur le code binaire, le code morse, le braille ou des codes créés par les élèves. Les élèves ont été nombreux à verbaliser des ressemblances avec des activités de programmation.

3. Géométrie

Nous avons également proposé des situations d'apprentissage en géométrie pouvant demander une remobilisation de ce qui a été appris en programmation. Il est important de préciser que nous utilisons en classe le terme « coder » pour faire figurer les propriétés des figures par des notations. Nous « codons » les angles droits par des carrés rouges, nous « codons » les côtés égaux par deux traits parallèles... Le lexique de la programmation reste donc présent dans cette discipline. Les élèves se sont ensuite vus proposer plusieurs situations d'apprentissages où nous leur avons proposé de réaliser, compléter ou rédiger des programmes de constructions complexes. Les programmes de construction font partie des compétences à acquérir par les élèves, toutefois nous avons pu observer une certaine aisance de la part des élèves pour rédiger ou énoncer ces programmes.

II – Protocole de recueil de données

Les missions de programmation et les situations d'apprentissages propices à la réutilisation des compétences travaillées en programmation nous ont permis d'observer si les élèves parvenaient à faire appel et transposer des compétences pour

accomplir des tâches ne relevant pas de la programmation. Toutefois, nous voulons savoir si les élèves conscientisent leur appropriation de ces compétences et la possibilité de transposer ces dernières dans d'autres disciplines.

A. Cadre des entretiens

Pour répondre à ce questionnement, nous avons mené des entretiens individuels auprès de quatre élèves à deux moments de l'année. Nous avons tout d'abord mené quatre entretiens au début du mois de décembre, puis quatre entretiens avec les mêmes élèves à la fin du mois de février. Ces échanges ont été effectués sur le temps du midi ou sur le temps des récréations et enregistrés avec un dictaphone. Lors de ces entretiens, nous avons posé plusieurs questions sur la programmation, certaines questions ont été communes puisque définies à l'avance et d'autres questions ont découlé des réponses des élèves pour demander des précisions, pour approfondir un point ou pour essayer de faire verbaliser davantage l'élève lorsque nous le sentions capable. Voici la liste des questions posées aux élèves lors de chaque entretien, ainsi que quelques exemples de questions posées pour préciser ou rebondir sur les réponses des élèves (de couleur plus claire) :

Depuis quand fais-tu de la programmation ?

Que faisiez-vous avant ?

Cette année, qu'avons-nous fait et que faisons-nous en programmation ?

La programmation se fait-elle toujours sur ordinateur ?

À quoi sert la programmation ?

Qu'apprenons-nous en programmation ?

Qu'est-ce qu'on travaille en programmation ?

Qu'est-ce qu'un programme ?

Pouvons-nous faire des programmes ailleurs que sur Scratch ?

Avons-nous besoin d'avoir des connaissances de d'autres matières pour faire de la programmation ?

Pourquoi as-tu besoin de savoir calculer ?

Pourquoi as-tu besoin de savoir lire et écrire ?

Dans quelle autre matière pouvons-nous avoir besoin des connaissances que nous avons grâce à la programmation ?

Pourquoi est-ce que tu trouves que cela se ressemble ?

Pourquoi trouves-tu que c'est un programme ?

Pourquoi dis-tu qu'il s'agit d'un code ?

Rappel à la séance musique grâce aux traces écrites conservées

Pourquoi cela te fait penser à la programmation ?

Rappel à la séance de technologie grâce aux exercices faits

Pourquoi cela te fait penser à la programmation ?

Est-ce qu'une recette de cuisine est un programme ?

Pourquoi ?

Nous avons choisi de mener les entretiens avec les mêmes élèves et les mêmes questions à deux moments différents de l'année pour observer d'éventuelles évolutions.

B. Sélection des participants

Pour réaliser ces entretiens, nous avons sélectionnés quatre élèves aux profils assez différents, ayant des affinités avec la programmation différentes et ayant des difficultés et réussites scolaires différentes également. Pour une question d'anonymat, nous nommerons ces quatre élèves : élève A, élève B, élève C, élève D. Nous allons décrire brièvement les profils de ces quatre élèves pour pouvoir analyser par la suite les données en prenant en compte les profils des élèves en question.

Elève A : Cette première élève fait partie des deux tiers des élèves ayant eu beaucoup d'activités de programmation l'année dernière en CE2. C'est une élève avec seulement quelques difficultés relatives au langage écrit. C'est une petite parleuse qui ose rarement prendre la parole devant le groupe classe mais n'hésite pas à faire appel aux enseignantes lorsqu'elle rencontre des difficultés. Elle est très motivée par les séances de programmation et a développé très vite sa connaissance de la plateforme Scratch et ses possibilités.

Elève B : Cette deuxième élève était également dans la classe de CE2 ayant beaucoup fait d'initiation à la programmation. Elle est arrivée dans l'école l'année dernière, elle n'a donc pas eu le parcours des autres élèves face à la programmation (depuis la Grande Section). Il s'agit d'une élève ayant un haut potentiel intellectuel (HPI), avec quelques difficultés de socialisation et de gestion des émotions. Elle a eu un temps de familiarisation plus long avec le logiciel Scratch et ses fonctionnalités, avec parfois de grosses difficultés de compréhension qui ont pu entacher sa motivation lors des premières séances. Une fois la plateforme bien appréhendée, elle s'est fortement impliquée dans les missions.

Elève C : Cet élève était dans la classe n'ayant eu que quelques séances de programmation l'année dernière. C'est un « bon élève », ayant acquis de nombreuses compétences dans toutes les disciplines. Il joue beaucoup aux jeux vidéo, en parle énormément et en utilise les termes techniques régulièrement. Il s'agit d'un des élèves de la classe étant le plus à l'aide avec la programmation. Il n'a pas eu beaucoup à se familiariser longuement avec Scratch en début d'année et a commencé à utiliser des fonctionnalités avant que cela soit demandé avec notre accord pour différencier les missions auprès des élèves en grande réussite.

Elève D : Cette dernière élève est une élève studieuse et moteur dans la classe. Les compétences évaluées sont toujours acquises et souvent dépassées. Toutefois, elle présente des difficultés lors des situations faisant appel à l'imagination, à la créativité. Cette élève est capable de donner les réponses attendues en respectant tous les critères demandés, mais a parfois du mal à verbaliser procédures engagées et justifier ses choix. Nous avons observé un fort attachement aux bonnes réponses, aux réussites dans les exercices et évaluations et un léger blocage lorsqu'il n'y a justement pas de bonne réponse attendue.

Nous allons désormais vous exposer les différents résultats obtenus après le recueil de données auprès des élèves que nous vous avons présentés.

RESULTATS

I. Conscientisation des compétences de programmation

Nous avons posé différentes questions aux quatre participants concernant l'enseignement de la programmation. À travers ces questions, nous cherchons à savoir s'il les élèves considèrent les séances de programmation comme des séances d'apprentissage, leur évaluation de l'utilité de l'enseignement de la programmation, ce que les élèves estiment apprendre lors de ces séances. Nous avons ainsi posé les questions suivantes pour répondre à ce premier champ d'interrogation :

« Que faisons-nous en programmation ? »

« À quoi sert la programmation ? »

« Qu'apprenons-nous en programmation ? »

Les questions n'ont pas toujours été posées de cette façon et dans cet ordre-ci, nous menions les entretiens en fonction des réponses apportées par les élèves.

A. La verbalisation des compétences

Nous avons dans un premier temps demandé aux élèves ce que nous avons fait et ce que nous faisons en programmation cette année. Puis, nous leur avons demandé ce que nous apprenions en programmation. Les réponses ont été parfois été sensiblement les mêmes pour ces deux questions.

Lors des premiers entretiens qui ont eu lieu au début du mois de décembre, pour la question « Que faisons-nous en programmation ? », les quatre participants ont répondu que nous faisons du scratch.

Elève A : « *Des trucs sur Scratch.* »

Elève B : « *On fait du Scratch par deux cette année.* »

Elève C (élève très à l'aise avec la programmation) : « *On fait encore du Scratch et pour ceux qui vont vite, on peut avoir le niveau 2, quand tu nous donnes des trucs plus compliqués à faire.* »

Elève D : « *On fait Scratch.* »

À la question « Qu'apprenons-nous en programmation ? », les réponses ont cette fois-ci été plus diverses. La question a été comprise de différentes manières.

Elève A : « *On apprend à faire du Scratch.* »

Elève B (élève HPI) : « *À créer des choses, à développer son intelligence, son imagination, sa créativité.* »

Elève C : « *À utiliser Scratch.* »

Elève D : « *À déplacer des personnages.* »

Durant les seconds entretiens, qui se sont tenus à la fin du mois de février, les élèves interrogés ont apporté des réponses plus développées pour les deux questions.

Réponses à la question « Que faisons-nous en programmation ? » :

Elève A : « *On apprend à faire des programmes pour nos personnages.* »

Elève B : « *On travaille sur Scratch et ceux qui ont fini Scratch ont commencé à faire des choses sur le robot.* »

Elève C : « *On fait un projet sur [...] les poilus et on va faire quand il est né, quand il est à parti à la guerre...* ». Question supplémentaire : Et avant ce défi final, qu'avons-nous fait durant l'année ? « *On a fait un truc sur Scratch avec les arrière-plans et avant on a fait des parcours pour que Scratch avance, et avant ça on a fait un truc un peu dur où on devait changer d'arrière-plan, parler... comme un scénario.* »

Elève D : « *On a fait beaucoup de missions sur Scratch, un peu de robot et là on va faire le projet sur le poilu de Saint-Pavace.* »

Réponses à la question « Qu'apprenons-nous en programmation ? » :

Elève A : « *On a appris plein de choses parce que avant moi je savais juste bouger les personnages. Bah là je peux le faire parler, lui donner un nouveau costume, lui mettre un arrière-plan différent même si ça marche pas toujours.* »

Elève B : « *On apprend à travailler à deux, comme [nom d'élève] et moi, et on dit ce qu'on préfère, et on partage pour que ça soit pas toujours la même qui décide tout.* »

Elève C : « *On apprend à faire tout ce qui est possible presque avec Scratch et le robot, sauf les catégories où il faut choisir les réponses avec des questions.* »

Elève D : « *On a appris à faire des bulles et les faire parler. On a appris à changer de fond et à changer de lutin. On apprend à utiliser le robot aussi.* »

Les réponses ont été plus détaillées lors des seconds entretiens et les élèves ont mis moins de temps à trouver leurs mots pour les formuler. Ils ont apporté des réponses différentes aux deux questions et ont su donner des exemples pour illustrer leurs propos.

B. La programmation en tant qu'enseignement

Pour essayer de comprendre comment les élèves perçoivent l'utilité de la programmation, pour savoir s'ils considèrent cela comme une matière, un enseignement, et donc une source d'apprentissages, nous leur avons posé la question suivante : « À quoi sert la programmation ? ». Les réponses suivantes ont été données :

Réponses à la question lors du premier entretien :

Elève A : « *À savoir se déplacer [...] avec les ordinateurs, avec des feuilles...* »

Elève B : « *À mieux connaître Internet ?* » Question supplémentaire : On va sur Internet parfois avec Scratch ? « *Euh non. C'est plutôt pour essayer... Quand on sera plus grand ça nous servira.* »

Elève C : « *Ça sert à savoir utiliser un ordinateur, à savoir utiliser des logiciels, à plein de trucs.* »

Elève D : « *Ça sert à quelque chose mais je sais pas quoi...* »

Réponses à la question lors du deuxième entretien :

Elève A : « *Ça sert à créer plein de choses, comme des programmes, inventer des choses pour faire comme un mini film...* »

Elève B : « *À savoir donner des indications, à être précis dans les mots qu'on utilise pour que quelqu'un d'autre nous comprenne.* »

Elève C : « *Ça sert à apprendre à faire des programmes, à savoir faire les blocs et les répétitions et ça peut être utile pour d'autres trucs ça.* » Question supplémentaire : Pour quoi cela peut être utile ? « *Je sais pas comment expliquer... Mais on réfléchit plus...* »

Elève D : « *Ça sert à plein de choses. C'est du travail donc on apprend sinon on en ferait pas avec les maîtres et les maîtresses.* »

Nous avons, une fois de plus, reçu des réponses différentes en fonction de la date de l'entretien et donc de l'avancée dans le projet de programmation.

II. Transversalité de ces compétences

Nous avons ensuite posé plusieurs questions pour interroger la notion de transversalité des compétences travaillées en programmation auprès des élèves et de leurs représentations.

A. Compétences réinvesties en programmation

Tout d'abord, nous nous sommes intéressés aux prérequis pour faire de la programmation. Nous avons demandé aux élèves s'ils avaient besoin d'avoir des connaissances provenant d'autres disciplines pour faire de la programmation. Nous interrogeons ici la transposition des compétences travaillées dans les autres matières aux séances de programmation.

Réponses à la question lors du premier entretien :

Elève A : « *Faut savoir lire parce que pour faire un programme t'as déjà des trucs qui sont marqués sur Scratch et faut savoir les lire pour savoir les utiliser.* »

Elève B : « *C'est pas obligatoire d'être fort en maths ou en français pour faire du Scratch. Par exemple si un élève est pas très très fort en maths ou en français ça peut l'aider à développer son intelligence et d'autres choses. Ça peut peut-être aussi aider par exemple à mieux écrire, à trouver plus vite...* »

Elève C : « *Oui parce que c'est du travail quand même la programmation. Il faut savoir bien lire et parler aussi.* »

Elève D : « *Je sais pas.* »

Réponses à la question lors du deuxième entretien :

Elève A : « *Faut savoir lire et écrire pour lire et écrire les bulles. Et il y a des programmes où il faut calculer par exemple comme quand on faisait les programmes pour le robot, il y avait marqué par exemple 1+1 ou 40+40 ou alors fallait deviner la vitesse mais fallait ajouter des 0 pour convertir.* »

Elève B : « *Faut savoir un peu l'orthographe pour savoir bien écrire et orthographier notre bulle si on veut faire des bulles pour parler. Mais j'ai pas trop d'exemples, là, tout de suite.* »

Elève C : « *Il faut savoir se repérer, par exemple avec la gauche et la droite ou le haut et le bas pour savoir où emmener le lutin, parce que si nous on sait pas bah lui non plus il pourra pas savoir où aller.* »

Elève D : « *Bah il faut savoir les additions pour le robot, et aussi les angles mais c'était compliqué, mais il fallait écrire les chiffres avec le petit rond (le degré) pour dire comment le lutin doit tourner.* »

Les réponses se ressemblent et peuvent être regroupées. Les élèves estiment que pour effectuer des activités de programmation, il faut savoir lire et écrire. Entre le premier et le deuxième entretien, les élèves ont été face à des situations de programmation faisant appel aux notions de degrés d'angle et de multiplications (sur Blockly, logiciel de programmation du robot Thymio). L'élève A et l'élève D s'appuient sur ces situations pour justifier la nécessité d'avoir des connaissances en mathématiques pour faire de la programmation. L'élève C évoque également une compétence de mathématiques, le repérage dans l'espace, une compétence que les programmes associent aux situations de programmation, mais aussi à des situations de repérage et déplacement sur des plans ou des cartes.

B. Réinvestissement des compétences de programmation

Nous avons voulu interroger les élèves sur la transposition possible d'une compétence propre à la programmation, la réalisation d'un programme. Nous avons défini ensemble, lors des missions, le programme comme étant une suite d'instructions, d'ordres à effectuer.

Dans un premier temps, nous les avons questionnés sur leurs conceptions du programme à travers la question suivante : « Qu'est-ce qu'un programme ? ».

Réponses à la question lors du premier entretien :

Elève A : « *C'est ce que tu fais avec des personnages pour qu'ils avancent, pour qu'ils reculent, pour qu'ils aillent à gauche, à droite.* »

Elève B : « *Un programme... on peut faire le programme de la journée, plein de choses.* » Question supplémentaire : Et sur Scratch, qu'est-ce qu'un programme ? « *Le programme là, sur Scratch, ça va être de savoir faire plus de choses qu'avant.* » Question supplémentaire : Et le script, qu'est-ce que c'est ?

« Ah oui bah là on apprend à écrire, à inventer une histoire, à présenter des choses... »

Elève C : « C'est quelque chose qu'on fait, c'est plein de trucs qu'on met à la suite pour que par exemple le lutin bouge. »

Elève D : « C'est faire faire des choses au lutin. »

Réponses à la question lors du deuxième entretien :

Elève A : « C'est tout ce que tu dis de faire à ton lutin ou à ton robot, tu les mets dans le script là et quand tu cliques sur le drapeau bah le lutin il fait tout ce que t'avais mis. »

Elève B : « Un programme... c'est ce qu'on prévoit de faire. »

Elève C : « C'est plein de blocs qu'on met ensemble, comme quand on fait les scripts des lutins ou aussi des arrière-plans, c'est tout ce qu'on veut qu'ils fassent à la suite. »

Elève D : « C'est tout ce qu'on met dans [le script] pour lui faire faire ce qu'on veut, c'est plein d'actions ou de choses à dire. On fait le programme des lutins et aussi des arrière-plans. »

Puis, dans un second temps, nous leur avons demandé s'il était possible de réaliser des programmes autrement. Nous souhaitons ici observer leur capacité à transposer ce savoir-faire technique dans d'autres situations. Nous avons posé la question suivante : « *Pouvons-nous faire des programmes ailleurs que sur Scratch ?* »

Réponses à la question lors du premier entretien :

Elève A : « Oui, sur Tuxbot ! Mais aussi on peut faire des programmes pour bouger comme le personnage-aimant qu'on avait fait avec le parcours et toi au tableau. »

Elève B : « Tu peux aussi l'utiliser sur les tablettes, les téléphones portables, partout. »

Elève C : « Oui sur Tuxbot et aussi sur d'autres trucs comme Minecraft³⁵ mais c'est compliqué, mais sur plein de trucs comme les ordinateurs et aussi pas sur les ordinateurs comme on avait fait avec le labyrinthe. »

Elève D : « Je ne sais pas. »

Réponses à la question lors du deuxième entretien :

Elève A : « Oui tu peux faire des programmes sur d'autres sites sur l'ordinateur et sur des feuilles et comme on avait vu sur les feuilles encore avec la musique, pour faire les gestes avec les gobelets. »

³⁵ Jeu vidéo

Elève B : « *Oui moi j'ai un programme le matin : je me lève, je m'habille, je me coiffe, je mange, et après je peux me poser dans le canapé parce que du coup je vais plus vite avec ce programme.* » Question supplémentaire : Et si tu donnes ce programme à exécuter à quelqu'un d'autre ? « *Bah il fera le même matin que moi avant d'aller à l'école.* »

Elève C : « *Oui sur les autres logiciels. Et aussi on peut faire des programmes de choses, comme des choses à faire, pour faire une danse ou des choses à faire dans une journée, c'est notre programme de la journée.* »

Elève C, plus tard dans l'entretien, après la question sur la recette de cuisine : « *Des fois on fait des programmes de construction avec le compas, la règle et l'équerre en géométrie. En fait quelqu'un dit ce que les autres doivent faire, comme une figure, mais les autres la voient pas parce qu'il la cache et on écoute ce qu'il dit, on fait tout et à la fin il nous montre et on regarde si on a pareil que la vraie figure. Et après on explique pourquoi on a pas pareil, comme quand [nom d'élève] avait dit rayon alors que c'était diamètre, bah faut utiliser les bons mots pour qu'on fasse la bonne figure. Ou alors faut dire les choses dans le bon ordre parce que des fois on a pas assez de place sur la feuille après.* »

Elève D : « *Oui pour le robot par exemple, ou pour un humain.* »

Enfin, nous avons sélectionné une situation de la vie courante pouvant être considérée comme un programme. Il s'agit alors de savoir si les élèves sont capables de justifier pourquoi cette situation leur fait penser à un programme, si c'est le cas. Nous avons posé cette question en dernière pour ne pas influencer les réponses de la question précédente. Nous leur avons ainsi demandé si une recette de cuisine est un programme.

Réponses à la question lors du premier entretien :

Elève A : « *Je sais pas trop... je sais pas.* »

Elève B : « *Oui parce que ils demandent de faire des choses et par exemple si on te dit « mets un œuf, plus trois millilitres de lait, plus trois cents grammes de farine », ça c'est un programme parce qu'on assemble des choses en consistant des matières pour pouvoir créer un seul truc à la fin.* »

Elève C : « *Bah oui.* »

Elève D : « *Oui ?* » Question supplémentaire : Pourquoi ? « *Parce qu'on nous dit de faire des choses.* »

Réponses à la question lors du deuxième entretien :

Elève A : « *Oui parce qu'on nous dit de faire ça, et ça, et ça... tous les aliments, et en fait faut le faire comme ils disent, faut faire dans le même sens et faut tout faire, faut pas en oublier.* »

Elève B : « *Oui, c'est un programme, c'est ce qu'on te dit de faire pour avoir le gâteau à la fin, il faut bien tout faire.* »

Elève C : « *Bah oui parce que regarde, si je dis « ajoute un œuf » c'est comme quand je dis « avance de dix pas » à mon lutin, il le fait, là aussi il faut ajouter un œuf. Il faut faire comme c'est écrit dessus pour avoir le gâteau par exemple à la fin de la recette. C'est comme si toutes les phrases de la recette c'est des blocs.* »

Elève D : « *Oui, parce qu'on nous dit de faire des actions et on les suit et comme ça on réalise le programme que la personne qui a écrit la recette voulait.* »

L'appropriation des élèves du terme « programme » a évolué au fur et à mesure des missions effectuées en classe, les réponses aux questions ont ainsi beaucoup évolué entre les deux entretiens. Certains élèves réussissent à trouver des situations de la vie quotidienne pouvant être catégorisées de programme. Tous les élèves réussissent à justifier pour quelles raisons ils considèrent la recette de cuisine comme étant un programme.

C. Prendre conscience d'une possible transposition

Les élèves ont réussi à transposer des compétences travaillées en programmation lors de séance d'éducation musicale et de technologie, présentées dans notre méthodologie. Cependant, nous avons cherché à savoir si les participants étaient conscients de la transposition qu'ils ont effectuée, notamment en étant capables de verbaliser des similitudes entre les situations d'apprentissages de technologie et d'éducation musicale et les séances de programmation.

Nous avons tout d'abord posé une première question aux élèves pour savoir si, sans situation donnée, ils étaient capables spontanément de trouver des disciplines où il est possible de faire appel à des connaissances acquises en programmation. La question était la suivante : « Dans quelle autre matière pouvons-nous avoir besoin des connaissances que nous avons grâce à la programmation ? »

Réponses à la question lors du premier entretien :

Elève A : « *Je ne sais pas.* »

Elève B : « *Plus tard par exemple on aura peut-être besoin de faire de la programmation dans notre travail et au moins maintenant on sait tous utiliser les ordinateurs.* »

Elève C : « *Je ne sais pas.* »

Elève D : « *Je ne sais pas.* »

Réponses à la question lors du deuxième entretien :

Elève A : « *Quand on fait de l'anglais avec le personnage qui disait son âge et où il habitait, il fallait faire un programme mais c'était de l'anglais.* »

Elève B : « *Bah on a bien progressé quand on tape sur les touches. Alors en APC pour faire le blog j'écris plus vite avec l'ordinateur par exemple, alors c'est grâce à la programmation.* »

Elève C : « *Quand on fait des choses où il y a besoin de coder comme en technologie la dernière fois, les codes pour les nuages de fumée, c'est un peu comme quand on faisait le labyrinthe, c'était des dessins.* »

Elève D : « *Je ne sais pas.* »

Juste après cette question, lors des premiers entretiens, nous avons montré aux participants des traces écrites des séances effectuées en éducation musicale faisant appel au code et au programme. Nous avons attendu les réactions des élèves sans poser de question. Nous avons réalisé la même opération avec des exercices faits lors des séances de technologie travaillant le codage et le décodage pendant les entretiens de février.

Réactions aux traces écrites de la séance de musique :

Elève A : « *Ah oui on devait mettre comment on faisait les gobelets ou les mains, par exemple on tapait avec le bord du rond pour boire là, et après on tournait à gauche et fallait noter ça sur la feuille.* » Question supplémentaire : A quoi cela te fait penser ? « *A une sorte de programme en fait, un peu mais pas trop.* » Pourquoi ? « *Bah tu dois noter ce que t'as fait, et si c'est bon bah... c'est bon par exemple si le gobelet tu dois le mettre à gauche et que tu le notes à droite ça sera pas bon.* »

Elève B : « *Ça ressemble un petit peu à de la programmation ça.* » Pourquoi ? « *Parce qu'on fait des schémas et on essaye d'inventer quelque chose comme on fait avec les gobelets comme taper puis claquer des doigts puis trois*

fois... C'est un peu comme un programme pour que après les autres ils puissent essayer de reproduire si ils ont envie ou pas. »

Elève C : « Ah oui ! C'est comme un programme parce qu'en fait des partitions c'est des feuilles, mais c'est aussi un programme qu'on fait mais sur feuille, pas sur ordinateur. »

Elève D : « Ah oui ça c'est ce qu'on avait fait là en musique, ça ressemble à ce qu'on avait fait avec les labyrinthes où il fallait sortir. On a fait des symboles pour dire par où aller ou quoi faire et comme là on avait des symboles pour dire aux autres quoi faire. »

Réactions aux traces écrites des séances de technologie :

Elève A : « Ça me fait penser à quand on avait fait les musiques avec les gobelets et qu'on devait dessiner ce qu'il fallait faire. »

Elève B : « C'était le code ça. »

Elève C : « Ça c'était quand on devait comprendre des signaux pour déchiffrer des informations, comme dans l'évaluation avec le braille, c'était écrit Harry Potter et il fallait connaître le code parce que sinon c'est impossible de deviner. Et aussi avec les nuages de fumée, aussi c'est nous qui avons inventé notre code. »

Elève D : « Bah là on a appris qu'une information était transmise avec des codes comme le français, le morse, le langage des signes. »

Les résultats ont pu évoluer entre les deux périodes d'entretiens grâce au projet de programmation. Nous observons des évolutions de schèmes chez tous les élèves concernant plusieurs aspects de la transposition des compétences travaillées en programmation.

DISCUSSION

I. Interprétation et analyse des résultats

Nous allons désormais nous intéresser aux résultats obtenus lors des deux séries d'entretiens menées auprès des quatre participants. Nous allons tenter d'interpréter et analyser les résultats obtenus en fonction de trois variables : la transposition, la conscientisation et la verbalisation.

A. Transposition

L'un des objectifs de cette recherche était de savoir si les élèves étaient capables de transposer des compétences travaillées en programmation dans d'autres disciplines, lors de situations d'apprentissage.

Nous avons tout d'abord demandé aux élèves ce qu'était un programme et s'il était possible d'effectuer des programmes ailleurs que sur Scratch. Nous avons ici questionné la transposition d'un savoir-faire, d'une compétence technique. Les programmes sont un apprentissage principalement fait en programmation, nous avons d'ailleurs défini ce qu'était un programme avec les élèves lors de séances de programmation. Sur cette transposition d'un savoir-faire, les réponses ont beaucoup évolué entre les premiers entretiens et les suivants, les élèves avaient cité uniquement des différents types de programmes faits lors de séances de programmation. Lors des seconds entretiens, les réponses ont été très diverses. Nous proposons une analyse en fonction du profil scolaire des élèves et de leur affinité avec la programmation.

L'élève C, un élève en grande réussite scolaire étant très intéressé par la programmation, a su apporter plusieurs situations scolaires ou de la vie quotidienne pouvant être considérées comme étant des programmes. Dans un premier temps, il explique que les programmes peuvent aussi se faire sur d'autres logiciels. Nous ne considérons pas cette réponse comme une reconnaissance de situation différente faisant appel au programme puisqu'il s'agit du même type de programme que sur

Scratch, un programme informatique, un ensemble d'instructions. L'élève a ensuite proposé une situation de la vie quotidienne : le programme de la journée.

« On peut faire des programmes de choses, comme [...] des choses à faire dans une journée, c'est notre programme de la journée. »

Il s'agit ici effectivement d'une situation étant un programme, un ensemble d'activités prévues pour la journée. L'élève cite, plus tard dans l'entretien, une situation scolaire où le groupe classe a déjà effectué des programmes ailleurs que sur Scratch. Il se rappelle d'activités où les élèves écrivaient des programmes de construction de figures géométriques.

« On fait tout [...] et après on explique pourquoi on a pas pareil [...] faut utiliser les bons mots pour qu'on fasse la bonne figure ou alors faut dire les choses dans le bon ordre. »

Il s'agit également de programmes qui proposent une suite d'instructions à suivre pour réaliser une tâche finale. L'élève a pu se souvenir de cette situation en faisant une analogie entre les termes utilisés – nous utilisons tous ensemble les termes de « *programme de construction* », ou bien une analogie entre les situations elles-mêmes, en remarquant qu'il s'agit dans les deux cas de suites d'instructions données.

L'élève A et l'élève D sont des élèves à l'aise avec la programmation, elles ont cité des situations relevant du programme informatique. L'élève B, quant à elle, a proposé une situation de la vie courante.

« Moi j'ai un programme le matin : je me lève, je m'habille, je me coiffe, je mange. »

Durant les entretiens, elle a plusieurs fois apporté des réponses s'écartant du sujet de la programmation, pas seulement pour cette question. Cependant, sa réponse s'explique par sa conception du programme. Lors de la question « Qu'est-ce qu'un programme ? », nous remarquons qu'elle considère le programme comme l'ensemble des activités qui sont prévues, et non comme le programme informatique.

Toutefois, les élèves ont tous été capables d'expliquer pourquoi une recette de cuisine pouvait être considérée comme un programme : « *Oui c'est un programme c'est ce qu'on te dit de faire pour avoir un gâteau à la fin, il faut bien tout faire* », « *C'est comme si toutes les phrases de la recette c'est des blocs.* » « *Oui parce qu'on nous dit de faire des actions et on les suit [...]* ». Les élèves ont plus facilement su répondre pour expliquer pourquoi ils observaient des ressemblances entre la recette et le programme, que pour trouver spontanément des éléments ressemblant à des

programmes. La transposition de ce savoir-faire technique a été possible avec un support. Les élèves sont capables de transposer leurs connaissances d'un support à un autre, mais il est plus difficile pour eux de trouver un support de transposition seuls.

Nous avons ensuite cherché à savoir quelles étaient les compétences travaillées dans d'autres disciplines dont ils pouvaient avoir besoin pour faire de la programmation, qu'ils pouvaient réinvestir. Tous les élèves considèrent que la maîtrise du français est indispensable pour effectuer des activités de programmation. Lors du premier entretien, toutes les réponses données n'évoquent que des compétences propres à la maîtrise du langage : le terme « *lire* » est donné deux fois, le terme « *écrire* » deux fois également et le terme « *parler* » une fois. Les élèves évoquent la nécessité de savoir lire, l'importance de savoir écrire et orthographier correctement les mots. Pour le second entretien, les termes « *lire* » et « *écrire* » reviennent à nouveau, mais sont complétés par des compétences propres aux mathématiques. En effet, deux élèves évoquent la nécessité de savoir « *calculer* » et « *faire des additions* », l'une d'elle évoque aussi la transposition de connaissances des mesures angles pour pouvoir faire effectuer des rotations au lutin sur Scratch. La seconde évoque, quant à elle, des situations de programmation nécessitant de savoir convertir des mesures de longueur.

Enfin, nous avons questionné les élèves sur la transposition de compétences travaillées en programmation à des situations d'apprentissage ayant lieu dans d'autres disciplines. Nous avons évoqué dans le cadre théorique des compétences de repérage dans l'espace, d'anticipation, de codage et de décodage... Lorsque nous avons montré aux élèves les traces écrites de la séance de musique pour qu'ils se remémorent ce que nous avons fait, trois utilisent des exemples pour exprimer le fait qu'il s'agisse de situations de codage et une explique qu'il s'agit d'une programme à suivre. En ce qui concerne la séance de technologie, trois élèves affirment également qu'il s'agit de situations de codage, la quatrième élève paraît également en être consciente puisqu'elle répond en citant une autre situation faisant appel au code (la séance de musique) et en signalant une analogie entre les deux.

B. Conscientisation

Les élèves sont capables de transposer des compétences techniques de programmation sur des situations scolaires ou de la vie quotidienne, mais semblent moins percevoir la possibilité de transposer des compétences transversales travaillées lors de séances de programmation, comme le repérage dans l'espace. Les compétences transversales sont travaillées, développées dans plusieurs disciplines, tandis que les compétences techniques sont travaillées dans une discipline. Les compétences transversales sont difficiles à nommer par les élèves. Seul l'élève C a cité une compétence transversale pour la question « Avons-nous besoin d'avoir des connaissances d'autres matières pour faire de la programmation ? » :

« Il faut savoir se repérer, par exemple avec la gauche et la droite ou le haut et le bas pour savoir où emmener le lutin, parce que si nous on sait pas ça bah lui non plus il pourra pas savoir où aller. »

Les autres élèves ont tous cité des compétences techniques, des savoir-faire, lors des questions sur la transposition. Nous remarquons cela par les tournures de phrases suivantes : « *On a appris à...* », « *On apprend à...* » et non « On a appris », « On apprend... ».

Pourtant, l'élève D semble consciente de ses connaissances, ses savoirs, puisqu'elle explique que lors de la séance de programmation : « *on a appris qu'une information était transmise avec des codes comme le français, le morse, le langage des signes...* ». Ce savoir semble être totalement conscientisé puisqu'elle le verbalise facilement. Il s'agit ici d'un savoir acquis dans le cadre d'une autre discipline, avec une leçon à apprendre pour retenir ces savoirs. L'élève D est une élève studieuse qui acquière très vite de nouveaux savoirs.

Nous constatons qu'il est compliqué pour les élèves d'avoir conscience de ce qu'ils savent ou savent faire. Un schème de pensée est l'organisation d'une image ou d'une notion. Pour Piaget (1961), nous combinons une représentation à une opération mentale qui permet d'organiser la nouvelle représentation avec nos représentations précédentes, ou de réorganiser ces dernières en fonction de la nouvelle. Il est difficile pour les élèves de mobiliser ces schèmes pour répondre lors des entretiens, en transposant leurs représentations à des situations analogues. La conscientisation de

ce processus peut être accompagnée par l'adulte pour que les élèves gèrent de façon autonome l'utilisation de telle ou telle fonction métacognitive.

C. Verbalisation

La verbalisation a été l'un des facteurs de compréhension dans cette recherche. Nous basons nos analyses de résultats sur les réponses données par les élèves. Toutefois, la verbalisation ne permet pas de rendre compte du processus de conscientisation des compétences puisqu'elle n'est pas la même pour tous les participants.

Nous avons tout d'abord observé d'importantes différences entre les premiers entretiens au début du mois de décembre et les seconds à la fin du mois de février. Nous avons obtenu des réponses beaucoup plus longues, avec plus de détails. À cela, plusieurs raisons peuvent être mises en lumière. Durant les trois mois séparant ces deux séries d'entretiens, plusieurs missions de programmation ont été faites, les élèves ont eu plus d'expérience avec cet enseignement, ils ont donc eu plus d'exemples sur lesquels s'appuyer pour répondre. De plus, les verbalisations lors des séances sont devenues de plus en plus riches, tous les élèves se sont appropriés le lexique technique et ont verbalisé ou ont entendu être verbalisés les savoir-faire acquis lors de chaque séance de programmation. En outre, nous pouvons également considérer le fait que les élèves connaissaient déjà les questions, puisque nous avons posé les mêmes lors des deux entretiens, ainsi le temps d'assimilation et de compréhension des questions pouvait être moins long et compliqué.

Il faut également prendre en considération le fait que les quatre participants n'ont pas verbalisé de la même manière, n'ont pas autant justifié ou argumenté leurs réponses. L'élève A, petite paroleuse dans le groupe classe, a été très à l'aise pour répondre et illustrer ses propos avec des exemples. L'élève B, HPI et rapport difficile à la programmation en début d'année, a parfois eu du mal à saisir le sens des consignes et a souvent apporté des réponses incohérentes avec la question. L'élève C, bon élève à l'aise avec la programmation, a été apte à développer ses réponses de manière très satisfaisante. L'élève D, studieuse mais craignant les mauvaises réponses, a eu des difficultés à répondre aux questions ouvertes ou à argumenter ses réponses.

II. Limites de la recherche

Nous avons, durant toute notre recherche, relevé différents éléments pouvant être améliorés, notamment d'un point de vue méthodologique.

Tout d'abord, nous n'avons effectué nos analyses qu'au regard de données récoltées auprès de quatre participants. Il aurait été intéressant de faire participer davantage d'élèves à ces questionnaires, avec des profils scolaires et des affinités à la programmation divers. Ce nombre de participants a été motivé par une volonté d'effectuer des entretiens d'une dizaine de minutes. Cependant, nous ne disposions pas de ressources de temps nécessaires pour en effectuer davantage. Un plus grand nombre d'élèves nous aurait peut-être amenés à des résultats différents, et donc à une analyse différente. Il aurait également été intéressant d'effectuer davantage d'entretiens, notamment une série d'entretiens en fin d'année scolaire pour rendre compte des évolutions.

Nous pensons également que la verbalisation a été un point de différenciation entre les quatre participants. Le choix de procéder par entretien a été décidé pour pouvoir demander des précisions ou des exemples plus facilement aux élèves pour développer leurs propos. Toutefois, nous pourrions envisager des questionnaires écrits avec la possibilité de répondre à l'écrit, ou en dessinant pour certaines réponses. Cela permettrait peut-être aux élèves de prendre plus de temps pour réfléchir à leurs réponses, de revenir plus facilement sur les réponses apportées aux questions précédentes.

De plus, pour vérifier la capacité des élèves à transposer leurs compétences dans des situations d'apprentissage relevant d'autres disciplines, nous nous sommes appuyés sur deux matières, éducation musicale et technologie. Toutefois, il est possible de mener davantage de situations faisant explicitement appel à des compétences de programmation, et ainsi réaliser des entretiens directement après ces situations pour faciliter le travail de remémoration des élèves.

Enfin, nous estimons que les questions posées font également partie des limites de cette recherche. En effet, les tournures des questions ont parfois été compliquées à comprendre pour les élèves, nous les avons d'ailleurs reformulées à quelques reprises. Si les questions ouvertes ont pu bloquer une des participantes, les trois questions fermées posées dans ces entretiens peuvent peut-être bloquer les élèves dans l'apport de justifications. Dans le cadre d'une même recherche menée auprès d'un plus grand nombre de participants, il peut être intéressant de favoriser les questions ouvertes.

CONCLUSION

Les résultats de ce travail de recherche démontrent que les élèves sont capables d'effectuer des transpositions de leurs compétences acquises lors des séances de programmation. Les élèves ont pu développer des compétences techniques, propres à la programmation, telles que le codage, le concept de programme ou l'utilisation de coefficient multiplicateur, mais également des compétences transversales, c'est-à-dire pouvant être travaillées dans n'importe quelle discipline et exploitées dans différentes situations.

Toutefois, nous observons qu'il est plus facile pour les élèves de verbaliser la transposition de compétences techniques, d'autant plus lorsque cette verbalisation s'appuie sur des exemples de traces conservées des séances d'apprentissage ayant permis de réaliser ces transpositions. Les élèves réussissent cette tâche par analogie. Nous interprétons donc les résultats de la façon suivante : les élèves sont conscients de la possibilité de réinvestir des compétences techniques de programmation, puisqu'ils la verbalisent.

En revanche, il est plus difficile pour les élèves interrogés de verbaliser les compétences transversales travaillées en programmation, la conscientisation doit donc l'être également. Nous ne pouvons cependant pas, à travers cette étude, connaître les schèmes de pensée opérés par les élèves quant aux compétences transversales.

Les élèves ont plus de difficultés à percevoir les savoirs que les savoir-faire, cela peut être dû à la vision qu'ont les élèves de la programmation. Il semble que, malgré les temps de verbalisation et le cadre donné à ce temps dédié, les élèves ne considèrent pas tous la programmation comme un enseignement, ou comme un enseignement aux finalités différentes des autres disciplines. Ces difficultés peuvent être mises en lien avec la place de la programmation dans les programmes de cycle 3, qui semble n'être qu'un léger point en « Espace et géométrie » mais qui nécessite pourtant de nombreuses séances dédiées pour travailler correctement tous les aspects avec les élèves.

BIBLIOGRAPHIE

Bugmann, J. et Karsenti, T. (2017). Pourquoi apprendre à coder à l'école ? *Apprendre et enseigner aujourd'hui*. Repéré à :

http://www.karsenti.ca/Apprendre_Enseigner_AUJ.pdf

Bugmann, J. et Karsenti, T. (2018). Apprendre à programmer un robot humanoïde : impacts sur des élèves de l'adaptation scolaire. *Formation et profession*, 26 (1), p. 26-42.

Bugmann, J. et Karsenti, T. (2017). Les technologies et coder à l'école : un jeu d'enfant ? *Savoir*. Repéré à :

<http://www.magazine-savoir.ca/2017/07/10/technologies-coder-a-lecole-jeu-denfant/>

Delvolve, N. (2005). *Tous les élèves peuvent apprendre. Aspects psychologiques et ergonomiques des apprentissages scolaires*. Paris : Hachette Education.

Delvolve, N. (2006). Métacognition et réussite des élèves. *Cahiers Pédagogiques*. Repéré à : <http://www.cahiers-pedagogiques.com/Metacognition-et-reussite-des-eleves>

Drechsler, M. (2017). Initier les élèves au codage et à la programmation. Repéré à : <https://www.reseau-canope.fr/savoirscdi/cdi-outil-pedagogique/conduire-des-projets/activites-pluridisciplinaires/lecriture-numerique/lapprentissage-du-codage-a-lecole.html>

Gagné, P.-P., Leblanc N. et Rousseau A. (2008). *Apprendre... une question de stratégies, Développer les habiletés liées aux fonctions exécutives*. Canada : Chenelière Education.

Komis, V. et Misirli, A. (2011). Robotique pédagogique et concepts préliminaires de la programmation à l'école maternelle : une étude de cas basée sur le jouet programmable Bee-Bot, in Baron, G.-L., Brillard, E., Komis, V., *Sciences et technologies de l'information et de la communication en milieu éducatif : analyse de pratiques et enjeux didactique*, p. 271-281, Athènes : New Technologies Editions.

Komis, V. Une analyse cognitive et didactique du langage de programmation ScratchJr. Repéré à : <http://docplayer.fr/57266243-Une-analyse-cognitive-et-didactique-du-langage-de-programmation-scratchjr.html>

Lazzarotto, F. et Cosson, M. (2014). La programmation dans les nouveaux programmes. Repéré à : <http://www.ac-grenoble.fr/tice74/spip.php?article1170>

Moreno-León, J., Robles, G. et Román-González, M. (2016). Code to learn: Where does it belong in the K-12 curriculum? *Journal of Information Technology Education: Research*, 15, 283-303. Repéré à : <http://dx.doi.org/10.28945/3521>

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

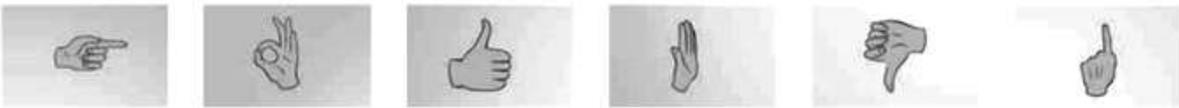
Piaget, J. (1955). *La genèse des structures logiques élémentaires : classifications et sériations*, Paris : Delachaux et Niestlé.

Touloupaki, S. et Baron G.-L. (2015). De la programmation à l'école primaire ? Une approche exploratoire en cycle 2. Présenté à Colloque ETIC2, Gennevilliers. Repéré à : <http://colloque-etic.fr/media/pdf/24.pdf>

Vergnaud, G. (2011). La conceptualisation, clef de voûte des rapports entre pratique et théorie, Eduscol. Repéré à : <http://eduscol.education.fr/pid25232-cid46598/la-conceptualisation-clef-de-voute-des-rapports-entre-pratique-et-theorie.html>

Annexe 2 : Exemples d'activités proposées aux élèves dans le cadre des séances de technologie traitant le thème « Signal et information »

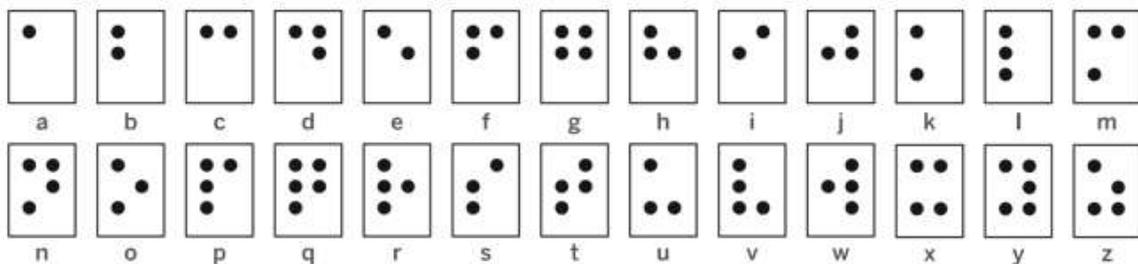
a b c d e f



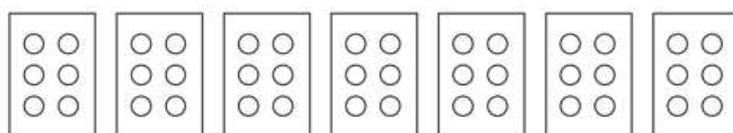
En t'aidant des signaux proposés au-dessus, quels gestes ferais-tu et dans quel ordre pour signifier « N'avance pas, tout va bien, regarde par ici. » ?

Le code Braille est utilisé par les personnes non voyantes pour lire. Chaque lettre est représentée par des points en relief. Chaque lettre est codée par 6 points qui sont en relief ou non.

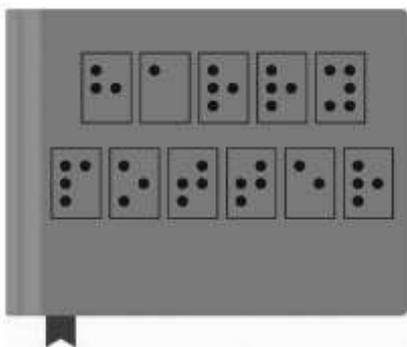
L'alphabet Braille :



Ecris le code pour « bonjour ».



Décode ce qui est écrit sur le livre.



Annexe 3 : Retranscriptions des réponses des élèves lors des entretiens

Depuis quand fais-tu de la programmation ?

	1 ^{er} entretien
Elève A	Bah depuis que je suis dans la classe avec Solène et Christine <i>L'année dernière tu n'en faisais pas ?</i> Euh non je crois pas...ah si des fois on en faisait quand on faisait des décloisonnements avec Carine ou aussi avec Stanislas
Elève B	Euh... un an ? <i>Est-ce que tu en as fait l'année dernière ?</i> Euh oui avec Stanislas, on faisait des petits exercices en fait il nous donnait des petites phrases qu'il fallait compléter mais c'était pas sur Scratch ça
Elève C	Euh... <i>Tu en faisais l'année dernière ?</i> On faisait des programmes sur scratch avec Erwan. On faisait des trucs sur Tuxbot. Aussi en CE1 on faisait des logiciels de maternelle pour apprendre à se guider. En CP aussi, et en grande section aussi, et en moyenne section aussi peut-être je ne me souviens plus.
Elève D	Depuis la grande section

Cette année, qu'avons-nous fait et que faisons-nous en programmation ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Des trucs sur Scratch, on apprend à se déplacer avec les personnages	On apprend à faire des programmes pour nos personnages
Elève B	On fait du Scratch par deux c'est tout...	Euh bah on travaille sur Scratch et ceux qui ont fini Scratch ont commencé à faire des choses sur le robot

Elève C	Bah on fait encore du Scratch et pour ceux qui vont vite on peut avoir le niveau 2 quand tu nous dis de faire des trucs plus compliqué	On fait un projet sur le barbu...le poilu et on va faire quand il est né, quand il est parti à la guerre... <i>Et avant ce projet final, qu'avons-nous fait toute l'année ?</i> On a fait un truc sur Scratch avec les arrière-plans, et avant on a fait des parcours pour que Scratch avance, et avant ça on a fait un truc un peu dur où on devait changer d'arrière-plan, parler comme un scénario...
Elève D	On fait Scratch	On a fait beaucoup de missions sur Scratch, un peu de robot et là on va faire le projet sur le poilu de Saint-Pavace avec les CM1-CM2

À quoi sert la programmation ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Bah à savoir se déplacer, avec plusieurs... par exemple avec les ordinateurs avec des feuilles... voilà	Bah ça sert à créer plein de choses comme des programmes, inventer des choses pour faire comme un mini film...
Elève B	À mieux connaître Internet... ? <i>On va sur Internet parfois avec Scratch ?</i> Euh non. C'est plutôt pour essayer...quand on sera plus grand ça nous servira	À savoir donner des indications, à être précis dans les mots qu'on utilise pour que quelqu'un d'autre nous comprenne
Elève C	Ça sert à savoir utiliser un ordinateur, à savoir utiliser des logiciels, à plein de trucs	Ca sert à apprendre à faire des programmes à savoir faire les blocs et les répétitions et ça peut être utile pour d'autres trucs ça <i>Pour quoi cela peut être utile ?</i> Je sais pas comment expliquer mais on réfléchit plus...
Elève D	Ça sert à quelque chose mais je sais pas quoi...	Ça sert à plein de choses... c'est du travail donc on apprend sinon on en ferait pas avec les maitres et les maîtresses

Qu'apprenons-nous en programmation ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Bah on apprend à faire Scratch	On a appris plein de choses parce que avant moi je savais juste bouger les personnages bah là je peux le faire parler, lui donner un nouveau costume, lui mettre un arrière-plan différent même si ça marche pas toujours
Elève B	À créer des choses, à développer son intelligence pour l'imagination, la créativité	On apprend à travailler à deux comme [nom d'élève] et moi, et on dit ce qu'on préfère et on partage pour que ça soit pas toujours la même qui décide tout
Elève C	A utiliser Scratch	On apprend à faire tout ce qui est possible presque avec Scratch et le robot sauf les catégories où il faut choisir les réponses avec des questions
Elève D	A déplacer des personnages...	On a appris à faire des bulles et les faire parler, on a appris à changer de fond, et à changer de lutins. On apprend à utiliser le robot aussi.

Qu'est-ce qu'un programme ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	C'est ce que tu fais avec des personnages pour qu'ils avancent pour qu'ils reculent pour qu'ils aillent à gauche à droite	Bah c'est tout ce que tu dis de faire à ton lutin ou à ton robot, tu les mets dans le script là et quand tu cliques sur le drapeau bah le lutin il fait tout ce que t'avais mis
Elève B	Bah un programme, on peut faire le programme de la journée plein de choses. <i>Et sur Scratch, qu'est-ce qu'un programme ?</i> Le programme là sur Scratch ça va être de savoir faire plus de choses qu'avant... <i>Et le script, qu'est-ce que c'est ?</i> Ah oui bah là on apprend à écrire, à inventer une histoire, à présenter des choses...	Un programme c'est ce qu'on prévoit de faire

Elève C	C'est quelque chose qu'on fait, c'est plein de trucs qu'on met à la suite pour que par exemple le lutin bouge	C'est plein de blocs qu'on met ensemble, comme quand on fait les scripts des personnages ou aussi des arrière-plans c'est tout ce qu'on veut qu'ils fassent à la suite
Elève D	Bah c'est faire faire des choses au lutin <i>Qu'est-ce qu'on peut lui faire faire ?</i> On peut le faire bouger, le faire parler...	Bah c'est tout ce qu'on met dans... euh... [<i>le script ?</i>] oui pour lui faire faire ce qu'on veut, c'est plein d'actions ou de choses à dire... On fait le programme des lutins et aussi des arrière-plans.

Pouvons-nous faire des programmes ailleurs que sur Scratch ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Bah oui sur Tuxbot ! Mais aussi on peut faire des programmes pour bouger comme le personnage aimant qu'on avait fait avec le parcours et toi au tableau Mais aussi l'année dernière on avait fait un programme par groupe et Stanislas il l'essayait et des fois il allait dans le mur et après on essayait d'arranger et d'aller dans les bons cerceaux	Oui tu peux faire des programmes sur d'autres sites sur l'ordinateur et sur des feuilles et comme on avait vu sur les feuilles encore mais avec la musique pour faire les gobelets
Elève B	Tu peux aussi l'utiliser sur les tablettes les téléphones portables partout...	Moi j'ai un programme le matin : je me lève, je m'habille, je me coiffe, je mange, et après je peux me poser dans le canapé parce que du coup je vais plus vite avec ce programme <i>Et si tu donnes ce programme à exécuter à quelqu'un d'autre ?</i> Bah il fera le même matin que moi
Elève C	Bah oui sur Tuxbot, et aussi sur d'autres trucs comme Minecraft mais c'est compliqué mais sur plein de trucs sur les ordinateurs et aussi bah pas sur ordinateurs comme on avait fait avec le labyrinthe	Oui sur les autres logiciels et aussi on peut faire des programmes de choses, comme des choses à faire pour faire une danse ou des choses à faire dans une journée c'est notre programme de la journée [plus tard dans l'entretien] Mais aussi des fois on fait des programmes de construction avec le compas la règle et l'équerre en géométrie et en fait quelqu'un dit ce que les autres doivent faire comme une figure mais les autres la voient pas parce qu'il la cache et on écoute ce qu'il dit, on fait tout et à la

		fin il nous montre et on compare pour voir si on a pareil que la vraie figure. Et après on explique pourquoi on a pas pareil comme quand [nom d'élève] avait dit rayon alors que c'était diamètre bah faut utiliser les bons mots pour qu'on fasse la bonne figure ou alors faut dire les choses dans le bon ordre parce que des fois on a pas assez de place sur la feuille après
Elève D	Je sais pas	Oui pour le robot par exemple ou pour un humain

Avons-nous besoin d'avoir de connaissances de d'autres matières pour faire de la programmation ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Faut savoir lire parce que pour faire un programme t'as déjà des trucs qui sont marqué sur Scratch et faut savoir les lire pour savoir les utiliser	Faut savoir lire et écrire pour lire et écrire les bulles. Et y a des programmes où faut calculer aussi parce exemple comme quand on faisait les programmes pour le robot, il y avait marqué par exemple 1+1 ou 40+40 ou alors fallait deviner la vitesse mais fallait convertir avec des 0
Elève B	C'est pas obligatoire d'être fort en maths ou en français pour faire du Scratch. Par exemple si un élève est pas très très fort en maths ou en français bah ça peut l'aider à développer son intelligence et d'autres choses, ça peut peut-être aussi aider par exemple à mieux écrire, à trouver plus vite...	Faut savoir un peu l'orthographe pour savoir bien écrire et orthographier notre bulle si on veut faire des bulles pour parler Mais j'ai pas trop d'autres exemples là tout de suite
Elève C	Oui <i>Pourquoi ?</i> Bah parce que c'est du travail quand même la programmation Il faut savoir bien lire et parler aussi	Il faut savoir se repérer, par exemple avec la gauche et la droite ou le haut et le bas pour savoir où emmener le lutin, parce que si nous on sait pas ça bah lui non plus il pourra pas savoir où aller
Elève D	Je sais pas...	Bah il faut savoir les additions pour le robot, et aussi les angles mais c'était compliqué mais il fallait écrire les chiffres avec le petit rond (<i>le degré</i>) pour dire comment le lutin doit tourner

Dans quelle autre matière pouvons-nous avoir besoin des connaissances que nous avons grâce à la programmation ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Je sais pas	Bah quand on a fait de l'anglais avec le personnage qui disait son âge et où il habitait là bah fallait faire un programme mais c'était de l'anglais
Elève B	Bah plus tard par exemple on aura peut-être besoin de faire de la programmation dans notre travail et au moins maintenant on sait tous utiliser les ordinateurs	Bah on a bien progressé quand on tape sur les touches alors en APC pour faire le blog j'écris plus vite avec l'ordinateur par exemple alors c'est grâce à la programmation
Elève C	Je ne sais pas	Bah quand on fait des choses où il y a besoin de coder comme en technologie la dernière fois les codes c'était un peu comme quand on faisait le labyrinthe Et aussi la programmation ça nous apprend à savoir où aller à se repérer dans quel sens on est et tout et ça c'était utile quand on a fait des programmes de danse un peu
Elève D	Je sais pas...	Je sais pas...

Rappel à la séance musique grâce aux traces écrites conservées.

	1 ^{er} entretien
Elève A	Ah oui on devait mettre comment on faisait avec les gobelets ou les mains, par exemple des fois on tapait avec le bord du rond pour boire là et après on tournait à gauche et fallait noter ça sur la feuille <i>A quoi cela te fait penser ?</i> A une sorte programme en fait un peu mais pas trop <i>Pourquoi ?</i> Bah tu dois noter ce que t'as fait, et si c'est bon bah c'est bon par exemple si le gobelet tu dois le mettre à gauche et que tu le notes à droite bah ça sera pas bon
Elève B	Ça ressemble un petit peu à la programmation ça <i>Pourquoi ?</i> Parce qu'on fait des schémas et on essaye d'inventer quelque chose comme n fait avec les gobelets comme taper puis faire [claque des doigts] puis trois fois... c'est un peu un programme pour que après les autres ils puissent essayer de reproduire si ils ont envie ou pas

Elève C	Ah oui ! c'est comme un programme parce qu'en fait des partitions c'est des feuilles mais c'est aussi un programme qu'on fait mais sur feuille par sur ordinateur <i>Pourquoi c'est un programme ?</i> Bah parce que c'est un dossier qu'on met mais en feuille... c'est dur à expliquer mais c'est tout ce qu'on voulait que les autres groupes fassent pour faire comme nous ce qu'on avait inventé
Elève D	Ah oui ça ce qu'on avait fait là en musique bah ça ressemble à ce qu'on avait fait avec les labyrinthes où il fallait sortir, on a fait des symboles pour dire par où aller ou quoi faire et comme là on avait fait des symboles pour dire aux autres quoi faire

Rappel à la séance de technologie grâce aux exercices faits.

2 ^{ème} entretien	
Elève A	Ben ça me fait penser à quand on avait fait les musiques avec les gobelets et qu'on devait dessiner ce qu'il fallait faire
Elève B	C'était le code ça...
Elève C	Ça c'était quand on devait comprendre des signaux pour déchiffrer des informations comme dans l'évaluation avec le braille c'était écrit Harry Potter et il fallait connaître le code parce que sinon c'est impossible de deviner Et avec les nuages de fumer aussi c'est nous qui avons inventé notre code
Elève D	Bah là on a appris qu'une information était transmise avec des codes comme le français, le morse, le langage des signes...

Est-ce qu'une recette de cuisine est un programme ?

	1 ^{er} entretien	2 ^{ème} entretien
Elève A	Bah je sais pas trop... je sais pas.	Bah oui parce qu'on nous dit de faire ça et ça et ça bah tous les aliments et en fait faut le faire comme ils disent faut faire dans le bon sens et faut tout faire faut pas en oublier
Elève B	Oui parce que ils nous demandent de faire des choses et par exemple si on te dit « mets un œuf + 3mL de lait + 300g de farine » ça c'est un programme parce qu'on assemble des choses en consistant des matières pour pouvoir créer un seul truc à la fin	Oui c'est un programme c'est ce qu'on te dit de faire pour avoir un gâteau à la fin, il faut bien tout faire

Elève C	Bah oui !	Bah oui parce que regarde si je dis « ajoute un œuf » c'est comme quand je dis « avance de 10 pas » à mon lutin il le fait bah là aussi il faut ajouter un œuf, faut faire comme c'est écrit dessus pour avoir le gâteau par exemple à la fin de la recette. C'est comme si toutes les phrases de la recette c'est des blocs.
Elève D	Oui <i>Pourquoi ?</i> Bah parce qu'on nous dit de faire des choses...	Oui parce qu'on nous dit de faire des actions et on les suit et comme ça bah on réalise le programme que la personne qui a écrit la recette voulait

