



d'

??,
de,
des



φ



d'

??,
de,
des



l'



z

Table des matières

- [Enseigner la programmation ? Pourquoi ? ce que disent les programmes](#)
- [Des défis : Pourquoi ? pour qui ? quand ? comment ? où ?](#)
- [La mise en œuvre du défi cycle 3](#)
- [Les pré requis](#)
- [Des ressources : Sitographie, bibliographie, ludographie](#)

➔ Enseigner la programmation ? Pourquoi ? Ce que disent les programmes

◆ Les programmes

Introduction

L'initiation à la programmation constitue une nouveauté importante pour les cycles 2 et 3. Elle s'inscrit dans les objectifs du socle commun de connaissances, de compétences et de culture, où il est précisé, dans le **domaine 1** (Les langages pour penser et communiquer) : « [L'élève] *sait que des langages informatiques sont utilisés pour programmer des outils numériques et réaliser des traitements automatiques de données. Il connaît les principes de base de l'algorithmique et de la conception des programmes informatiques. Il les met en œuvre pour créer des applications simples.* ». Il s'agit aux cycles 2 et 3 d'amorcer un travail qui sera poursuivi au cycle 4.

L'initiation à la programmation apparaît dans les programmes au sein du thème **Espace et géométrie** en lien avec l'objectif « (Se) repérer et (se) déplacer en utilisant des repères » au cycle 2 et « (Se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations » au cycle 3.

La diversité des équipements sur le territoire nécessite de s'appuyer sur des activités faisant appel des supports variés :

- sans matériel spécifique, « en débranché » ;
- des robots programmables ;
- des applications en ligne utilisables sur ordinateurs ou tablettes ;
- des logiciels pouvant être installés sur des ordinateurs ou des tablettes.

L'initiation à la programmation pourra être une opportunité pour des travaux interdisciplinaires : avec le champ **questionner le monde au cycle 2**, par exemple, autour de la question du repérage ; ou avec le français, dans le développement des usages du langage oral ou écrit, notamment en créant des histoires illustrées par de courtes animations créées par les élèves ; ou encore en langues vivantes en créant des animations où les personnages dialoguent en langue étrangère ; etc.

Progressivité des apprentissages

Les apprentissages se construisent progressivement tout au long des quatre cycles de l'école et du collège.

Au cycle 1, les élèves apprennent à « utiliser des marqueurs spatiaux adaptés (devant, derrière, droite, gauche, dessus, dessous...) dans des récits, descriptions ou explications ». Ils apprennent également à « situer des objets par rapport à soi, entre eux, par rapport à des objets repères », « se situer par rapport à d'autres, par rapport à des objets repères » et « dans un environnement bien connu, réaliser un trajet, un parcours à partir de sa représentation (dessin ou codage) ». Ce travail leur permet de développer l'aptitude à émettre des instructions élémentaires de déplacement, instructions qu'ils apprendront à associer dans les cycles suivants pour construire des programmes de déplacement.

Au cycle 2, les élèves apprennent à « coder et décoder pour prévoir, représenter et réaliser des déplacements dans des espaces familiers, sur un quadrillage, sur un écran ». Ces déplacements ont lieu dans des espaces réduits en début de cycle (classe ou école) pour s'étendre progressivement tout au long du cycle jusqu'au quartier ou village pour lesquels ils pourront utiliser des plans. À partir du CE1, les élèves sont invités à coder des déplacements à l'aide d'un logiciel de programmation adapté.

Au cycle 3, les élèves apprennent à « programmer les déplacements d'un robot ou ceux d'un personnage sur un écran ». Les élèves travaillent « dans des espaces de travail de tailles différentes (la feuille de papier, la cour de récréation, le quartier, la ville, etc.) », ils utilisent pour cela des plans en travaillant « avec de nouvelles ressources comme les systèmes d'information géographique ». Le cadre est aussi celui « d'activités géométriques (construction de figures simples ou de figures composées de figures simples) », en utilisant des logiciels de géométrie dynamique.

[Table des matières](#)

• **Au cycle 4**, Algorithmique et programmation devient un thème d'étude à part entière au même titre que Nombres et calculs ou Espace et géométrie. L'objectif est d'amener les élèves à « écrire, mettre au point et exécuter un programme simple ». « Les élèves s'initient à la programmation, en développant dans une démarche de projet quelques programmes simples, sans viser une connaissance experte et exhaustive d'un langage ou d'un logiciel particulier. En créant un programme, ils développent des méthodes de programmation, revisitent les notions de variables et de fonctions sous une forme différente, et s'entraînent au raisonnement. »

*Extrait du document Eduscol Maths C2-C3 Initiation Programmation –
Pour lire l'intégralité, : téléchargement du document [ici](#)*

◆ Quelques définitions

Les mots « algorithme » et « programmation » étant polysémiques, il est nécessaire de clarifier le sens. Les mots « algorithme » et « programmation » s'entendent ici comme création d'un programme d'actions, à ne pas confondre avec « algorithme » qui signifierait « rythme de la douleur », ce qui serait une conception navrante des mathématiques !

Travailler les algorithmes en maternelle n'est pas seulement construire « des colliers de perles » de couleurs répétitives...

Extrait de « algorithme et pédagogie » (Dominique Pernoux)

L'algorithme est essentiellement connu à l'école maternelle sous la forme de suites algorithmiques (exemples ci-dessous), mais moins sous sa forme de suites de règles permettant de résoudre un problème tel que cité par Bouvier (voir p 5) :

On peut distinguer deux types de suites algorithmiques :

• les plus « simples » sont les **suites algorithmiques répétitives** comme par exemple :

△ ○ □ △ ○ □ △ ○ □ △ ○ □ △ ○ □ △ ○ □ ... (remarque : il s'agit ici d'une suite algorithmique répétitive ternaire)

• les plus « complexes » sont les **suites algorithmiques récursives** comme par exemple :

○ △ ○ ○ △ ○ ○ ○ △ ○ ○ ○ ○ △

(remarque : la récursivité, qui permet de définir successivement chacun des objets de la suite à partir des objets précédents joue un rôle important en mathématiques et en informatique...)

• On peut signaler, par ailleurs, les deux notions suivantes :

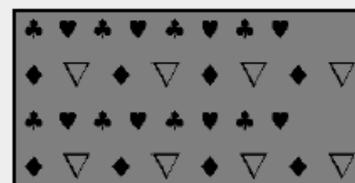
Les suites isomorphes : les suites suivantes ♣ △ △ ♣ △ △ ♣ △ △ ♣ △ △et

tambour cloche cloche tambour cloche cloche tamboursont isomorphes (elles ont la même structure).

(la notion d'isomorphisme est très importante en mathématiques car elle permet d'aller vers plus d'abstraction en dégagant des structures communes à des objets a priori de natures différentes...).

A cela peut s'ajouter le **pavage périodique** se compose en fait de plusieurs suites algorithmiques dans différentes directions.

Exemple : (Il y a ici des suites verticales, horizontales et obliques)



Il s'agit surtout d'initiation à la pensée algorithmique, plus que de programmation : savoir décomposer un problème en tâches simples, savoir reconnaître les tâches qu'on a déjà effectuées ou qui se répètent...

Pour comprendre et aborder la notion d'algorithme, voir cette vidéo (3min)

<http://www.universcience.tv/video-les-sepas-et-les-algorithmes-5829.html>

Pour travailler avec les élèves :

- ◆ D'où viennent les algorithmes ? <https://www.lumni.fr/video/d-ou-viennent-les-algorithmes>
- ◆ C'est quoi le code informatique ? <https://www.lumni.fr/video/c-est-quoi-le-code-informatique-1-jour-1-question>

« Un algorithme est un enchaînement ordonné d'actions, qui chacune a un effet, et dont l'exécution complète permet de résoudre une classe de problèmes. »

Il existe une sous-catégorie de l'algorithme qui est l'algorithme « instancié ». Celui-ci ne résout qu'un problème particulier.

- Exemples d'algorithmes au sens de la définition encadrée qui résout toute une classe de problèmes : le tri, le calcul posé...
- Exemples d'algorithmes « instanciés » qui ne résout qu'un seul problème : le codage d'un parcours de Bee-bot, la recette de cuisine.

Source : texte de Pierre Tchounikine (...)

La notion d'algorithme n'est pas propre à l'informatique : en maths les programmes indiquent notamment la compétence cible « Mettre en œuvre un algorithme de calcul posé pour l'addition, la soustraction, la multiplication ». Une recette de cuisine est un algorithme, dont les actions sont de « casser les œufs », de « mettre de la farine » ou de « si nécessaire, rajouter un peu de sel ». En informatique, les actions dont il est question sont, par exemple, d'afficher une information à l'écran, de demander une donnée à l'utilisateur ou de faire un calcul. Elles sont souvent structurées par des boucles (qui permettent de répéter une séquence d'actions) et/ou des structures conditionnelles (structures Si-Alors-Sinon, qui permettent de faire une séquence d'actions ou une autre en fonction de l'évaluation d'une condition). Un programme est un algorithme traduit dans un langage de programmation. Un ordinateur est une machine qui (entre autres) est capable d'exécuter un programme... »

Pour exprimer un algorithme plusieurs éléments peuvent être utilisés :

1. des instructions ordonnées en séquences
2. des boucles (répétition d'une séquence organisée)
3. des tests (des instructions conditionnelles)
4. des variables

L'algorithme peut être construit sans ordinateur, en langage naturel oral et/ou écrit (ce que l'on veut faire). Il est ensuite codé dans le langage correspondant à l'outil utilisé (Bee-bot, Scratch...). On obtient ainsi un programme qui permet de valider l'algorithme et le codage par l'outil.

Pour aller plus loin :

→ le rapport IGEN du (décembre 2015) : algorithmique au cycle 3

http://cache.media.eduscol.education.fr/file/Formation_continue_enseignants/58/1/151204-algorithmique-liensmodifies_511581.pdf

→ le texte de Pierre Tchounikine (professeur à l'université Grenoble-Alpes) : Initier les élèves à la pensée informatique et à la programmation avec Scratch

<http://lig-membres.imag.fr/tchounikine/PenseeInformatiqueEcole.html>

→ l'article de vulgarisation d'Aurélien Alvarez et de Thierry Viéville dans « Images des maths » du CNRS : « Dis maman (ou papa), c'est quoi un algorithme dans ce monde numérique ?

<http://images.math.cnrs.fr/Dis-maman-ou-papa-c-est-quoi-un-algorithme-dans-ce-monde-numerique>

→ Bouvier (2005, p.27) la définition de l'algorithme : « Un algorithme est une suite finie de règles à appliquer dans un ordre déterminé à un nombre fini de données pour arriver avec certitude (c'est-à-dire sans indétermination ou ambiguïté), en un nombre fini d'étapes, à un certain résultat et cela indépendamment des données. Un algorithme ne résout donc pas un problème unique mais toute une classe de problèmes ne différant que par les données mais gouvernés par les mêmes prescriptions. »

Des défis : C'est quoi ? Pourquoi ? pour qui ? quand ? comment ? où ? avec quel matériel ?

◆ Un défi, c'est quoi ?

C'est une situation-problème qui doit :

- avoir du **sens** (interpeller, concerner l'apprenant qui ne se contente pas d'obéir, d'exécuter),
- être liée à un **obstacle** repéré, défini, considéré comme dépassable et dont les apprenants doivent prendre conscience à travers l'émergence de leurs conceptions (représentations mentales),
- faire naître un **questionnement** chez les élèves (qui ne répondent plus aux seules questions du maître),
- créer une ou des **ruptures** amenant à déconstruire le ou les modèles explicatifs initiaux s'ils sont inadapés ou erronés,
- correspondre à une **situation complexe**, si possible liée au réel, pouvant ouvrir sur différentes réponses acceptables et différentes stratégies utilisables,
- déboucher sur un **savoir d'ordre général** (notion, concept, loi, règle, compétence, savoir-être, savoir-devenir...),
- faire l'objet d'un ou plusieurs moments de **métacognition** (analyse a posteriori de la manière dont les activités ont été vécues et du savoir qui a pu être intégré).

(Gérard De Vecchi, dans *Enseigner l'expérimental en classe*, Hachette Éducation)

[Table des matières](#)

◆ Pourquoi ?

Les défis-code proposés ont pour objectifs de créer une dynamique de classe, de fédérer des groupes classes, de créer du lien et donner du sens à l'enseignement de la programmation. Ce défi-code est un dispositif clé en main permettant aux PE la mise en œuvre d'enseignement de la programmation en classe.

Les compétences travaillées :

PROGRAMMES DU CYCLE 3 :

FRANÇAIS

Langage oral

Participer de façon constructive aux échanges avec d'autres élèves dans un groupe pour confronter des réactions ou des points de vue.

Participer à des échanges dans des situations diverses

EMC

Identifier et exprimer en les régulant ses émotions et ses sentiments

Exprimer son opinion et respecter l'opinion des autres

Être capable de coopérer

Savoir s'intégrer dans une démarche collaborative et enrichir son travail ou sa réflexion grâce à cette

Démarche

HISTOIRE ET GEOGRAPHIE

Se repérer dans le temps e : construire des repères historiques

Ordonner des faits les uns par rapport aux autres et les situer dans une époque ou une période donnée.

S'informer dans le monde du numérique

Connaitre différents systèmes d'information, les utiliser

Trouver, sélectionner et exploiter des informations dans une ressource numérique

MATHEMATIQUES

Espace et géométrie

Se repérer, décrire ou exécuter des déplacements, sur un plan ou sur une carte

SOCLE COMMUN DE CONNAISSANCE ET DE CULTURE

Domaine 1

Les langages pour penser et communiquer

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

Domaine 2

Les méthodes et outils pour apprendre

Tous les enseignements doivent apprendre aux élèves à organiser leur travail pour améliorer l'efficacité des apprentissages. Elles doivent également contribuer à faire acquérir la capacité de coopérer en développant le travail en groupe et le travail collaboratif à l'aide des outils numériques, ainsi que la capacité de réaliser des projets.

En mathématiques, ils apprennent à utiliser des logiciels de calculs et d'initiation à la programmation.

◆ Pour qui ?

Ces défis s'adressent aux élèves du cycle 3.

◆ Quand ?

Les défis du groupe départemental vous sont proposés sur 2 périodes de l'année (période 2 et 3 de l'année scolaire) et sur une durée de 15 jours pour les mettre en œuvre dans la classe.

◆ Comment ?

Les classes s'inscrivent aux défis pour l'année scolaire en cours. Avant la date du début du défi, les épreuves sont mises en ligne. Une période de 15 jours pour réaliser ces défis et faire parvenir au groupe départemental ses productions.

En retour des indices pour résoudre une énigme en fin de défi seront adressés aux classes participantes. Les classes pourront ensuite proposer des défis sur les thématiques des défis en cours.

◆ Où ?

Ces défis sont à réaliser en classe avec une modalité organisationnelle propre à chaque enseignant.

◆ Avec quel matériel ?

- Des Bee-bot ou Blue-bot (Robot disponible en circonscription ou dans des écoles proches de la vôtre)
- Un plateau d'évolution pour les robots (le plateau pour les défis est proposé à chaque fois en téléchargement) Vous pouvez prévoir un tapis transparent quadrillé 15x15cm sous lequel vous glissez les feuilles du plateau proposé. Ce tapis peut être réalisé avec une nappe cirée transparente (épaisseur moyenne) que vous quadrillez en 15x15cm
- Si vous avez un TNI c'est une plus-value pour projeter les défis, faire travailler un groupe d'élèves dessus, effectuer la synthèse collective... et vous pouvez prolonger les activités
- Du papier et des crayons pour les activités débranchées

[Table des matières](#)



La mise en œuvre du défi

- Dès que le défi est mis en ligne (calendrier communiqué aux classes inscrites), les classes peuvent travailler et répondre aux questions du défi en téléchargeant la fiche activité et la fiche réponse.
- Avant la clôture du défi en cours, les classes inscrites doivent renvoyer leurs documents réponses en respectant les consignes pour l'envoi
- A la fin de la période du défi, des indices seront envoyés aux classes. Leur nombre varie en fonction des bonnes réponses.
- A réception des indices et avant la date indiquée, les classes doivent résoudre l'énigme.

[Table des matières](#)

Les pré requis - Activités d'entraînement

Avant de relever les défis proposés, il peut être nécessaire d'effectuer un module d'apprentissage sur l'introduction des robots en classe.

Ce module est accessible [ici](#)

Des ressources

◆ Sitographie :



1,2,3... CODEZ !

Le guide présente des activités branchées et débranchées, scénarisées de façon à offrir aux élèves de chaque cycle une entrée en matière accessible et attirante. À part au cycle 1, les modules d'activités ne sont pas réellement conçus pour être mis en place in extenso au sein d'une classe, de la première à la dernière séance : en effet, nous proposons plusieurs variantes d'un même parcours selon le matériel à disposition de la classe, et selon le temps que l'enseignant peut y consacrer. Le guide est ainsi conçu de façon à permettre à l'enseignant de « piocher » les séances qui lui permettront de réaliser son propre parcours.

<https://www.fondation-lamap.org/fr/page/34459/module-pedagogique-1-2-3-codez>

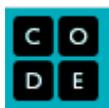


\$_CODÉFI

ROBOTIQUE ET OBJETS CONNECTÉS



<http://codefi.dane.ac-versailles.fr/>



<https://code.org/learn> (Blockly Games) et

<https://studio.code.org/courses>

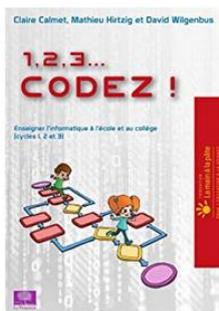


Stamp it : <http://cic-lavaladjoind-ia53.ac-nantes.fr/codblocs/index.php>

L'application en ligne STAMP IT ! est une simulation de machine à tamponner consistant en un automate orienté se déplaçant sur une grille de forme carrée. L'automate peut se déplacer en marche avant ou en marche arrière, le pas de déplacement étant fixe (1/8, 1/10 ou 1/16 de la longueur du côté de la grille). Il peut également pivoter de 90° à droite ou à gauche. La fonction principale de la machine consiste à tamponner la grille avec une couleur préalablement sélectionnée parmi une palette prédéfinie. La zone tamponnée correspond à la case dans laquelle est situé l'automate. Le contrôle de l'automate se fait par le biais d'une interface de programmation visuelle permettant de concevoir des programmes à partir d'une bibliothèque de blocs organisée en plusieurs catégories.

◆ Bibliographie :

📖 **1,2,3... codez !** - Auteur : Claire Calmet (pour les enseignants) :



Le guide présente des activités branchées et débranchées, scénarisées de façon à offrir aux élèves de chaque cycle une entrée en matière accessible et attirante. À part au cycle 1, les modules d'activités ne sont pas réellement conçus pour être mis en place in extenso au sein d'une classe, de la première à la dernière séance : en effet, nous proposons plusieurs variantes d'un même parcours selon le matériel à disposition de la classe, et selon le temps que l'enseignant peut y consacrer. Le guide est ainsi conçu de façon à permettre à l'enseignant de « piocher » les séances qui lui permettront de réaliser son propre parcours.

📖 **Lucie le robot** - Auteur : Leroy, Jean



Résumé : Le papa de Lucie est vraiment un super-papa : il lui a offert un robot ! Kop joue au foot, fait du vélo et prépare même les crêpes ! Mais Kop sait-il vraiment tout faire ?

📖 **Robot, mais pas trop** - Auteur Éric Simard



Résumé : Adam vit dans une maison très sophistiquée. Quand il crie « Tarzan », son lit se transforme en toboggan. Pour actionner la chasse d'eau, il faut chanter : « Il pleut, il pleut, bergère... » Mais certains appareils sont un peu détraqués... Alors, quand le directeur de l'école s'invite à prendre le thé, c'est la panique !

📖 **Le môme en conserve** - Auteur Christine Nöstlinger



Résumé : Mme Bartolotti adore commander par correspondance les objets les plus divers. Elle reçoit un jour une immense boîte de conserve, avec un mode d'emploi... Ayant suivi les instructions, elle obtient un enfant de sept ans en parfait état de marche - un garçon modèle garanti par le fabricant ! Mais Frédéric est trop parfait, et devient le sujet de moquerie des écoliers. Il faut qu'il apprenne quelques mauvaises manières

📖 **Au secours, le maître est un robot** - Auteur Anne-Gaëlle Balpe



Résumé : La rentrée des classes de Barnabé, Lili et Martin ne se passe pas du tout comme prévu, car leur maître est... un robot ! Avec monsieur ZR38, adieu les récréés, bonjour les dictées... Rapidement, les trois amis mettent au point un plan pour que le maître-robot pète un boulon !

◆ Ludographie :



Marbology est un jeu de logique. Marbology, ce sont des tuiles coulissantes et des billes, pour s'entraîner à la logique en solo : un jeu de taquin et de déplacement, qui s'appuie sur 52 défis de 4 niveaux différents, pour des heures à se creuser la tête ! A partir d'une position de départ, le but du jeu est de parvenir à amener les boules colorées à leur point d'arrivée, en faisant astucieusement coulisser les tuiles blanches pour créer un chemin.



Jeu de logique alliant déplacement et programmation pour aider les écureuils à cacher leurs noisettes. On place les écureuils, la fleur et les noisettes comme indiqué sur le défi sélectionné. L'objectif est de mettre les noisettes dans les trous du plateau de jeu. Les noisettes sont placées entre les pattes des écureuils qui vont les déposer dans les trous, ce qui est à la fois concret et très mignon... Il faut organiser et programmer leurs déplacements pour réaliser les défis. Plus on avance dans les défis, plus il y a du monde sur le plateau et plus il faut réfléchir!



Jeu de logique: après avoir positionné les véhicules suivant le plan d'une des cartes, il faut faire sortir le camion du marchand de glace en faisant glisser les autres véhicules, sans jamais les soulever du plateau!



Plus qu'un jeu : apprendre à coder aux enfants est aujourd'hui une nécessité. Ce kit d'activités permet aux enfants de s'initier à la programmation informatique.

Dans un ensemble de chemins, d'abord simples puis plus complexes, construits à partir de dalles de mousse de différentes couleurs, les enfants vont apprendre à se diriger à l'aide d'instructions logiques (les « cartes de codage ») : tout droit, tourner, stop, pour accomplir leur mission : aller chercher un petit robot. Des variantes sont possibles avec des objets spéciaux (pince, jet pack..), pour compliquer le tout. Un enfant peut en guider un autre, à l'aide des flèches, et s'ils rencontrent une difficulté, ils réfléchissent ensemble à la manière de la résoudre.