



MINISTÈRE
DE L'ÉDUCATION NATIONALE,
DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE



Algorithmique et programmation

Inspection pédagogique régionale



ALGORITHMIQUE ET PROGRAMMATION AU CYCLE 2

Espace et géométrie : (se) repérer et (se) déplacer en utilisant des repères et des représentations

- Connaissances et compétences associées
 - Coder et décoder pour prévoir, représenter et réaliser des déplacements [...] sur un écran
- Exemples
 - Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran
- Repères de progressivité
 - Dès le CE1, les élèves peuvent coder des déplacements à l'aide d'un logiciel de programmation adapté, ce qui les amènera au CE2 à la compréhension, et la production d'algorithmes simples



ALGORITHMIQUE ET PROGRAMMATION AU CYCLE 3

Contribution au socle commun

- Domaine 2
les méthodes et outils pour apprendre
« En mathématiques, ils apprennent à utiliser des logiciels de calculs et d'initiation à la programmation »

Programme de mathématiques

Préambule

«[...] des activités géométriques peuvent être l'occasion d'amener les élèves à utiliser différents supports de travail : papier et crayon, mais aussi logiciels de géométrie dynamique, d'initiation à la programmation ou logiciels de visualisation de cartes, de plans. »

Thème espace et géométrie

« Les activités spatiales et géométriques [...] constituent des moments privilégiés pour une première initiation à la programmation notamment à travers la programmation de déplacements ou de construction de figures. »

Thème espace et géométrie

Connaissances et compétences, situations

- (Se) repérer et (se) déplacer dans l'espace en utilisant ou en élaborant des représentations
 - Se repérer, décrire ou exécuter des déplacements, sur un plan ou sur une carte
 - Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran
- Exemples
 - Travailler avec de nouvelles ressources comme des logiciels d'initiation à la programmation

Thème espace et géométrie

Connaissances et compétences, situations

- Reconnaître, nommer, décrire, reproduire, représenter, construire quelques solides et figures géométriques
 - Réaliser, compléter et rédiger un programme de construction
 - Réaliser une figure simple ou une figure composée de figures simples à l'aide d'un logiciel



ALGORITHMIQUE ET PROGRAMMATION AU CYCLE 4

Thème E : algorithmique et programmation

« Au cycle 4, 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. »

Repères de progressivité

« En cinquième, les élèves s'initient à la programmation événementielle.

Progressivement, ils développent de nouvelles compétences, en programmant des actions en parallèle, en utilisant la notion de variable informatique, en découvrant les boucles et les instructions conditionnelles qui complètent les structures de contrôle liées aux événements. »

Connaissances et compétences associées

- Décomposer un problème en sous-problèmes afin de structurer un programme ; reconnaître des schémas
- Écrire, mettre au point (tester, corriger) et exécuter un programme en réponse à un problème donné
- Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs
- Programmer des scripts se déroulant en parallèle
 - Notions d'algorithme et de programme
 - Notion de variable informatique
 - Déclenchement d'une action par un événement, séquences d'instructions, boucles, instructions conditionnelles

Algorithmique et programmation au cycle 4

- Attendu de fin de cycle : « *Écrire, mettre au point et exécuter un programme simple* »
- Accent mis sur :
 - l'analyse du problème en vue de concevoir un algorithme
 - La phase de programmation effective, indissociable d'une étape de mise au point

Écrire, mettre au point exécuter un programme simple

- Pas de chronologie précise, avancée en parallèle en raison de :
 - phase d'essais-erreurs
 - questions algorithmiques non prises en compte
- Pas de distinction algorithme/programme
- Pas de langage de description d'algorithme (pseudo-langage ou organigramme)
- Phase de mise au point essentielle :
 - Jeu de tests
 - Comparaison de solutions algorithmiques et programmes, conduisant à une approche informelle de la documentation d'un programme (commentaires et spécification)



OBJECTIFS DE FORMATION

Objectifs de formation

- Pas d'expertise visée
- Apport de clés de décryptage d'un monde numérique en évolution constante
- Acquisition de méthodes qui construisent la pensée algorithmique
- Développement de compétences dans :
 - La représentation de l'information et son traitement
 - La résolution de problèmes
 - Le contrôle des résultats

Objectifs de formation

- Mise en place de modalités d'enseignement fondées sur la pédagogie de projet
- Les langages informatiques :
 - pas une finalité
 - pratique permettant d'acquérir :
 - d'autres démarches d'investigation
 - d'autres modes de résolution de problèmes
 - d'autres modes de simulation ou de modélisation



COMPÉTENCES DÉVELOPPÉES

Connaissances et compétences

Rôles communs aux mathématiques

- En algorithmique, développer des compétences :
 - de décomposition de tâches complexes
 - de généralisation et d'abstraction
- En programmation :
 - Mettre en action des concepts de mathématiques (par exemple en géométrie)

Compétences développées

- Décomposition :
 - analyser un problème compliqué
 - le découper en sous-problèmes, en sous-tâches
- Reconnaissance de schémas :
 - reconnaître des schémas, des configurations, des invariants, des répétitions
 - mettre en évidence des interactions

Compétences développées

- Généralisation et abstraction :
 - repérer les enchaînements logiques
 - les traduire en instructions conditionnelles
 - traduire les schémas récurrents en boucles (programmation impérative)
 - concevoir des méthodes liées à des objets qui traduisent le comportement attendu (programmation orientée objet)

Compétences développées

- Conception d'algorithme :
 - écrire des solutions modulaires à un problème donné
 - réutiliser des algorithmes déjà programmés
 - programmer des instructions déclenchées par des événements
 - concevoir des algorithmes se déroulant en parallèle



DÉMARCHE PÉDAGOGIQUE

Modalités d'apprentissage

- Travail en mode débranché
- Individuel ou en groupe
- En salle informatique ou en salle banale
- Sur tablette ou ordinateur

Une pédagogie spécifique

« L'informatique a ceci de particulier de faire très rapidement appel à des démarches de passage à l'abstraction afin d'élaborer des solutions génériques à de grandes classes de problèmes [...].

Les élèves, qui ont aujourd'hui une connaissance pratique voire intuitive de ces contextes (tirée de la manipulation quotidienne des objets numériques), maîtrisent assez peu les concepts sous-jacents et ne sont généralement pas prêts à aborder directement l'abstraction. »

Réponses pédagogiques

- Mise en activité des élèves aussi fréquente que possible
- Alternance entre les temps consacrés :
 - aux notions fondamentales
 - à ceux dédiés aux applications et projets

Démarches pédagogiques

- Démarche de projet active et collaborative :
 - établissement d'objectifs partagés et répartition des tâches
 - communication entre élèves contributeurs d'un même projet (avec intervention possible de plusieurs disciplines)

Démarches pédagogiques

- Démarche de création :
 - réalisation de productions collectives :
 - programmes
 - applications
 - animations, ...
 - Développement de :
 - l'autonomie
 - la créativité et l'imagination
 - du sens du travail collaboratif
- Démarche interdisciplinaire :
 - mise en œuvre d'activités de création numérique, en particulier dans le cadre des enseignements complémentaires

Pédagogie de projet

- Objectif : construire des connaissances et des compétences en développant imagination et créativité dans le cadre d'un projet suivi sur quelques séances
- Approche possible :
 - Séance 1 : activité introductive et exploitation de nouveaux concepts
 - Séance 2 : développement de programmes dans les directions choisies par les élèves, aidés par leur professeur
 - Séance 3 : finalisation des projets et mise en commun des concepts et techniques

Mise en activité des élèves sur une séance

- Viser des objectifs de formation clairs et explicites
- L'essentiel consacré à une activité autonome
- Organisation possible :
 - Introduction d'une notion par une situation problème
 - Réinvestissement d'une activité avec exploitation de cette notion
 - Bilan récapitulatif

Boucle essai-erreur

- La programmation informatique permet de :
 - procéder par essais et erreurs
 - d'analyser
 - de vérifier et de corriger
- Le constat de l'erreur n'est plus un obstacle, mais une étape à interpréter dans un processus d'apprentissage où la médiation de la machine garantit la brièveté de la boucle essai-erreur et évite tout jugement porté sur la personne de l'élève

Boucle essai-erreur

- L'obtention d'une solution peut être poursuivie pour en améliorer la qualité, la performance...
- Aucun jugement de valeur n'est porté sur l'élève ou ses performances : il constate s'il s'est trompé ou non, et essaie une autre solution en cas d'échec
- Le professeur accepte la nécessité de phases de tâtonnement et d'essais-erreurs, durant lesquelles il intervient le moins possible