PIT

PROGRAMME FOR · INNOVATIVE · TEACHING AND · TRAINING

1#Involution

1.1 Indication didactique

Dominic Harion & Ann Kiefer

«Le big data et les algorithmes qui l'accompagnent signent un retour accru des mathématiques et des méthodes scientifiques dans l'organisation du social», constate Felix Stalder à propos des systèmes de décision automatisés dans nos cultures numériques (Stalder, 2017). Loin de l'importance qui leur est accordée en informatique, les algorithmes sont, de manière très générale, d'abord des «consignes de résolution d'un problème ou d'un type de problèmes et se composent d'un nombre fini d'unités bien définies» – même les modes d'emploi des machines et les recettes de cuisine ne sont finalement que des algorithmes. La connaissance de ce que signifie une telle algorithmicité dans notre environnement et de ce qu'elle implique, en particulier dans son imbrication avec les technologies de notre vie quotidienne, fait partie d'une exigence éducative fondée, tout comme les aptitudes à la modéliser, à l'appliquer et à la modifier si nécessaire. Tout d'abord, les algorithmes ne sont pas spécifiquement technologiques, ni même techniques ou numériques. Mais ils ne peuvent (et ne devraient) plus être supprimés de nos infrastructures. Sous ces formes spécifiques, ils prennent en charge des processus complexes.

Dans les systèmes de cartographie et d'organisation de la mobilité dans la topographie, les algorithmes font partie intégrante des processus d'organisation sociale. Les plans des transports publics et les applications de mobilité en sont des exemples concrets. «L'algorithme du plus court chemin» dans un graphe, connu dans les milieux spécialisés sous le nom d'algorithme Breadth-first search ou d'algorithme de Dijkstra (dans les cas où les arêtes du graphe ont des poids), représente ainsi une étude de cas très proche de la vie quotidienne et en même temps facile à étudier de manière autonome pour une mise en œuvre en classe. Dans le cadre du module présenté ici, une approche ludique et collaborative de cet algorithme est choisie via Involution©, un jeu ou casse-tête mathématique développé à l'Université du Luxembourg par Hugo Parlier et Bruno Teheux. L'apprentissage des modélisations algorithmiques par le biais de procédures ludiques et compétitives est déjà testé et discuté depuis longtemps à l'aide du Rubik's Cube (voir par exemple Lakkaraju et al., 2022). Elles ouvrent des voies d'apprentissage à l'interface des mathématiques et de l'informatique (voir par exemple Joyner, 2008 ; Agostinelli et al., 2019). Si ces cubes permettent d'illustrer des modèles tridimensionnels, Involution© permet de reproduire et de clarifier des modèles algorithmiques dans un espace bidimensionnel. Ce jeu est donc approprié pour ouvrir un horizon d'apprentissage transdisciplinaire en sciences numériques par le biais de la formation de modèles mathématiques liés à des problèmes du monde réel : la résolution du jeu Involution© est liée à la recherche du chemin le plus court dans un système de métro, une mission que les élèves connaissent déjà du cours de Digital Sciences (Digital Sciences, 2021). Cette recherche du plus court chemin a déjà été explorée par des enfants et des adolescents de différentes tranches d'âge et utilisée pour concrétiser des modèles formels abstraits (cf. Gibson, 2012). Elle peut donc être utilisée avec des degrés de difficulté progressifs pour un enseignement différencié.

D'un point de vue didactique, le module est axé sur les deux principes de l'apprentissage par résolution de problèmes et par coopération. Un modèle de jeu et des instructions de base sont fournis aux élèves confronté·e·s à un problème. Cela permet de stimuler un processus cognitif ciblé afin de transformer une situation donnée en une situation cible, sans donner de méthode évidente pour la résoudre, tout en faisant appel à la pensée créative et critique (cf. Mayer & Wittrock, 2006). Cela permet notamment aux élèves de changer de perspective sur les disciplines STIM (sciences, technologie, ingénierie et mathématiques). Pour de nombreux jeunes, il s'agit dans ces matières de trouver la «seule et unique» solution et ce, le plus rapidement possible, ce qui ne correspond pas à la réalité du monde scientifique. Les scientifiques des disciplines STIM n'ont généralement pas d'idée claire de la direction à prendre pour trouver une solution – cet état d'esprit est parfois négligé dans les disciplines STIM en milieu scolaire.

Les jeux et casse-têtes comme Involution© offrent également un cadre parfait pour entraîner la résilience des élèves. «La résilience est liée à la capacité affective des élèves à faire face aux obstacles et aux situations négatives du processus d'apprentissage et à les surmonter, en transformant ces situations négatives en situations qui les soutiennent.» (Hutauruk & Priatna, 2017). Les élèves qui ne sont pas habitué·e·s à gérer les situations d'apprentissage frustrantes et les échecs les considèrent comme très négatives. Mais si les élèves y sont habitué·e·s, cette expérience de résilience a des conséquences tout à fait positives pour les études et le futur emploi des jeunes (Hutauruk & Priatna, 2017).

La tâche n'est pas résolue par chaque élève individuellement, mais en binômes ou en petits groupes. Elle est construite de manière compétitive (mais non concurrentielle), comme un rallye de classe. Cette approche ludique et motivante stimule également la modélisation et la verbalisation de solutions personnelles et offre des possibilités d'apprentissage par l'enseignement, car les élèves doivent communiquer leurs processus de réflexion et les étayer par des arguments. Il est possible d'envisager une configuration dans laquelle les groupes sont constitués de manière homogène en fonction du degré de difficulté des tâches, ainsi qu'une autre configuration avec des groupes hétérogènes dans laquelle les élèves sont coaché·e·s par leurs camarades de classe lors de l'élaboration des solutions.

Références :

Agostinelli, Forest, Mavalankar, Mihir, Khandelwal, Vedant, Tang, Hengtao, Wu, Dezhi, Berry, Barnett, Srivastava, Biplav, Sheth, Amit & Irvin. Matthew. (2021). Designing Children's New Learning Partner: Collaborative Artificial Intelligence for Learning to Solve the Rubik's Cube. In Interaction Design and Children (IDC '21). Association for Computing Machinery, New York, NY, USA, 610–614. https://doi.org/10.1145/3459990.34651750

Digital Sciences. (2021). Mon monde numérique et moi. Schüler Achse 1.

Gibson, Paul J. (2012). Teaching graph algorithms to children of all ages. In Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education (ITiCSE '12). Association for Computing Machinery, New York, NY, USA, 34–39. https://doi.org/10.1145/2325296.2325308

Hutauruk, Agusmanto J.B., & Priatna, Nanang. (2017). Mathematical Resilience of Mathematics Education Students. J. Phys.: Conf. Ser. 895 012067. https://iopscience.iop.org/article/10.1088/1742-6596/895/1/012067/pdf

Joyner, David (2008). Adventures in group theory: Rubik's Cube, Merlin's machine, and other mathematical toys (2nd ed.). Baltimore : John Hopkins University Press.

Lakkaraju, Kausik, Hassan, Thahimum, Khandelwal, Vedant, Singh, Prathamjeet, Bradley, Cassidy, Shah, Ronak, Agostinelli, Forest, Srivastava, Biplav, & Wu, Dezhi. (2022). ALLURE: A Multi-Modal Guided Environment for Helping Children Learn to Solve a Rubik's Cube with Automatic Solving and Interactive Explanations. (Preliminary Preprint). https://www.aaai.org/AAAI22Papers/DEMO-00182-LakkarajuK.pdf

Mayer, Richard E. & Wittrock, Merlin C. (2006). Problem Solving. In P. A. Alexander & P. H. Winne (Eds.). Handbook of Educational Psychology (2nd Ed), 287-303. New York : Routledge.

Rogers, Hartley (1987). Theory of Recursive Funtions and Effective Computability. Massachusetts : C1957.

Stalder, Felix. (2017). Algorithmen, die wir brauchen. https://netzpolitik.org/2017/algorithmen-die-wir-brauchen/

1.2 Planification de l'unité

01 Sujet de l'unité dans la structure globale des axes

Module	Axe Thématique	Focus	Idées interdisciplinaires et lien avec d'autres matières
#Involution	Axe 1 Mon monde numérique et moi !	• Jeux et algorithmes • Algorithme du plus court chemin	• Mathématiques • Géographie
#Climate Killer Internet	Axe 2 Comprendre l'internet : World Wide Web et moi.	• Internet et le climat • Compétences de jugement	• VIESO • Géographie • Allemand • Français
#Data Viz Superpowers	Axe 3 Do you speak Informatique ?	• Différentes formes de la data visualisation • Manipulation de graphiques	• Éducation artistique • Mathématiques
#Discover Life on Mars with a Rover	Axe 5 Le robot, partenaire pour le meilleur et le pire ?	• Programmation en Scratch • Educational Robotics	• VIESO
#Pupils vs Machine	Axe 6 Une machine aussi intelligente que moi, ça existe ?	• Fonctionnement de base d'une intelligence artificielle	• Mathématiques • VIESO

Comme les modules sont indépendants les uns des autres, il n'est pas nécessaire d'avoir pris connaissance des autres modules pour traiter celui-ci.

Ce module a été conçu en collaboration avec Hugo Parlier et Bruno Teheux du département de mathématiques de l'Université du Luxembourg.

02 Modalités de l'unité

1. Public visé : 7e-5e classique et générale
2. Local : pas de local spécial à prévoir
3. Matériel nécessaire :
   - Le jeu Involution© : le jeu peut être emprunté au Centre de documentation pédagogique de l'IFEN ou au F.use (Future Space for Education) au Campus Belval.
   - Des ordinateurs ou tablettes peuvent être utilisés pour lire les instructions, Ces dernières peuvent aussi être tout simplement imprimées.
4. Durée : 2 heures d'enseignement

03 Contextualisation des Savoirs

L'objectif principal de la leçon est de donner corps aux notions d'algorithme vues en cours via *Involution©* : un jeu ou casse-tête développé par Hugo Parlier et Bruno Teheux, deux chercheurs en mathématiques de l'Université du Luxembourg. Par le biais de ce jeu, les élèves se familiarisent avec des concepts mathématiques et algorithmiques comme l'algorithme du plus court chemin, les stratégies optimales et non optimales, le tout de manière ludique. Le jeu consiste en une rangée de 10 anneaux noirs et blancs.
Une manivelle en forme de cercle est placée sur le jeu et par un mouvement de rotation, les anneaux changent de position. Le but du jeu est de changer la position des anneaux afin d'obtenir une configuration donnée à partir d'une configuration de départ.

04 Transposition didactique

a. Objectifs d'apprentissage et compétences visées

Compétences visées de l'Axe 1 : Mon Monde Numérique et Moi !

- SAVOIRS : connaissances de base du vocabulaire numérique et informatique général, de la communication entre humain et outil informatique (notion d'algorithme) et du principe d'abstraction en matière de résolution de problèmes.
- SAVOIR-FAIRE : décomposition d'un problème à des fins de résolution, conception et représentation d'un algorithme, élaboration et application d'un algorithme simple avec des cartes et d'un déroulement de programme.
- SAVOIR-ÊTRE : prise de conscience de l'implication de l'informatique dans les tâches quotidiennes en fonction des choix en matière d'outils, d'informations fournies et des résultats obtenus.

Compétences visées du Guide de référence pour l'éducation aux et par les médias[1]

- Compétences 2 – Communication et collaboration : 2.1 Interagir avec autrui
- Compétences 3 – Créer des contenus : 3.3 Modéliser, structurer et coder

[1] https://www.edumedia.lu/medienkompass/medienkompass/

b. Justification didactique

L'objectif de ce module est que les élèves comprennent la notion et le fonctionnement des algorithmes à travers un jeu. Une des particularités du module est de n'utiliser ni technologie électronique, ni ordinateur, ni tablettes. Ces activités, dites « débranchées », servent souvent à initier à la pensée informatique pour « se libérer de la complexité technologique lors de l'apprentissage et découvrir les notions fondamentales en informatique » (INRIA, 2020). L'accès est donc aisé et permet de comprendre un aspect essentiel du numérique : les cultures numériques n'ont en premier lieu rien à voir avec les ordinateurs au sens de machines électriques – la caractéristique de l'algorithmicité, par exemple, qui est au cœur de ce module, peut être décrite par la formation de modèles mathématiques et logiques formels, qui peuvent également être représentés de manière mécanique. Pour citer l'informaticien Leslie Lamport : « L'importance de réfléchir et d'écrire avant de coder doit être enseignée dans les cours d'informatique de premier cycle et elle ne l'est pas » (Han, 2022).

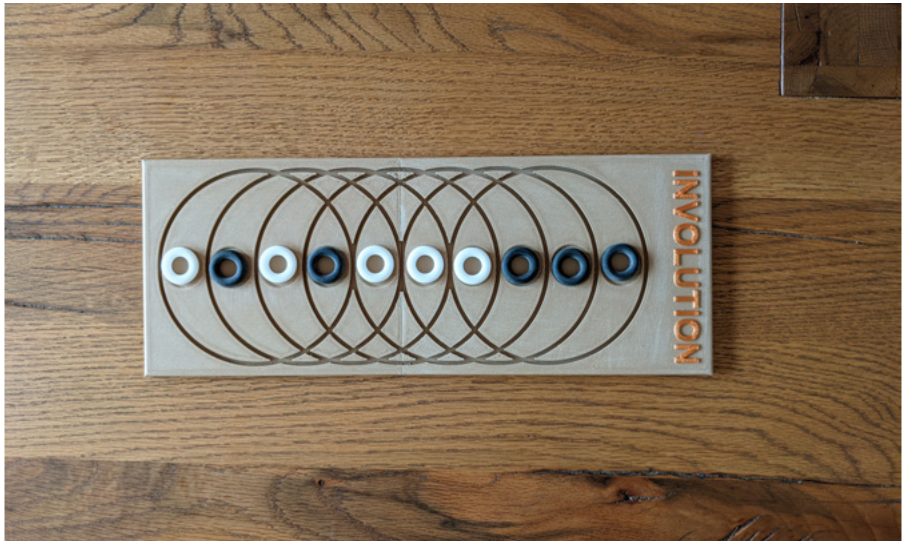
Ainsi, le module #Involution s'appuie sur les bases du codage enseignées dans le Fondamental et les développe, à la croisée des mathématiques et de l'informatique.

c. Réduction didactique

Involution© propose plusieurs défis de type « shortest path » de différents niveaux et les situe dans le cadre de l'apprentissage coopératif et de la résolution de problèmes. Le jeu stimule un processus cognitif ciblé qui peut être développé activement à partir du matériel pédagogique distribué. La planification et la mise en œuvre de l'unité d'enseignement peuvent faire totalement abstraction d'une répartition stricte du temps au profit d'une expérimentation libre avec Involution© et les instructions mises à disposition. La consolidation des connaissances et des compétences se fait par le biais de l'apprentissage par l'enseignement, les élèves présentant à la classe leurs solutions ainsi que les raisons de leur choix.

05 | Déroulement de l'unité

Dans ce module, les élèvent jouent à un jeu baptisé Involution©. Le jeu est composé d'un plateau et d'une manivelle. Le plateau contient 10 cavités qui contiennent chacune 1 anneau noir ou blanc.



Une manivelle en forme de cercle est placée sur le jeu et par un mouvement de rotation, les anneaux changent de position. Le but du jeu est de changer la position des anneaux afin d'obtenir une configuration donnée à partir d'une configuration de départ.

Involution 1

Le jeu

L'enseignant·e explique rapidement le principe du jeu. L'entrée en matière se passe directement par le jeu. Les élèves se répartissent en binômes. Chaque groupe reçoit une boîte qui contient un plateau de jeu (à assembler) et des cartes de défis. Chaque carte présente deux configurations. Le défi consiste à passer de la configuration de gauche à la configuration de droite en tournant la manivelle. Les élèves commencent à faire les défis dans l'ordre donné. Cette activité constitue le premier défi d'une longue liste de défis qui constituent ce module. Tous les défis sont donnés dans les matériels pédagogiques.

Après cette initiation, les élèves passent aux différents défis. Les explications ci-dessous concernent ces mêmes défis. Le module comprend de nombreux défis. Dans le meilleur des cas, tous les défis seront réalisés. Les défis munis d'une étoile sont plus compliqués (et de nature plus mathématique). Ils sont destinés aux élèves les plus motivé·e·s. En outre, d'autres défis sont optionnels (comme montré dans la planification détaillée de l'unité plus bas). Les laisser de côté n'empêche pas la réalisation du module.

Défi 2 : après une première phase, quand les élèves ont compris le principe du jeu, il·elle·s passent au défi 2. L'enseignant·e divise la classe en trois groupes ; un groupe travaille sur le premier des trois défis, l'autre sur le deuxième et le dernier sur le troisième. Le premier défi est légèrement plus simple que les deux autres : tenez-en compte lors de la division de la classe en groupes et de la répartition des défis.

Une fois le défi résolu, les élèves doivent se mettre d'accord au sein de leur groupe, puis communiquer leur résultat au reste de la classe.

Défi 3* : le défi 3, plus compliqué, est optionnel. À réaliser avec des élèves motivé·e·s et d'un bon niveau.

Comment obtenir toutes les configurations ?

Cette section aide les élèves à résoudre le défi 3. Elle concerne donc uniquement les élèves qui l'ont essayé. Parmi les défis présents dans cette section, le défi 9 est d'un niveau particulièrement élevé. Nous laissons le choix à l'enseignant·e de décider quel·le·s élèves peuvent/doivent résoudre les défis de cette section sans ou avec le défi 9.

Cette section guide les élèves vers une résolution du défi 3 via le principe mathématique de l'induction (sans pour autant en parler, ni même le mentionner).

Une solution optimale

Défi 10 : au défi 10, les élèves cherchent une solution et comptent le nombre de mouvements qu'il·elle·s font pour arriver à la solution. Il·elle·s comparent leurs solutions entre eux·elles.

En débat dirigé, il est conclu qu'une solution est meilleure qu'une autre si elle requiert moins de mouvements.

Une définition semblable à la suivante est établie ensemble en plénière :

Une solution est optimale si elle est réalisée avec le moins de mouvements possible.

À la recherche d'une solution optimale

Comment trouver cette solution optimale ? Faire le lien avec la recherche de l'itinéraire le plus court dans le métro de New York, vue en cours de Digital Sciences 1.

Comment représenter le jeu sous forme de graphe ?

Cette section commence par un exemple plus simple : le jeu Involution©, mais avec seulement 5 anneaux et un plateau de jeu réduit. Les 5 cavités les plus à droite ne doivent pas être utilisées. Il n'y a donc que 2 positions possibles pour poser la manivelle.

Défi 11 : l'enseignant·e divise la classe en trois groupes. Au sein de chaque groupe, les élèves restent en binômes pour jouer.

- Le premier groupe joue avec 1 anneau noir (et 4 anneaux blancs).
- Le deuxième groupe joue avec 2 anneaux noirs (et 3 anneaux blancs).
- Le troisième groupe joue avec 3 anneaux noirs (et 2 anneaux blancs).

Les élèves résolvent leur cas en binôme. Ensuite, il·elle·s discutent au sein de leur groupe, puis communiquent leur résultat au reste de la classe.

Défi 12* : ce défi est plus compliqué et optionnel.

Les élèves passent à la représentation graphique du jeu (à l'aide du même jeu réduit).

Défi 13 : pour ce défi et les suivants, le concept de graphe doit être introduit ou brièvement rappelé. Les élèves travaillent en binômes, puis comparent leurs résultats. Il·elle·s vont constater que le deuxième et troisième cas sont exactement les mêmes. En débat dirigé, les élèves sont amené·e·s au fait que ces deux cas sont symétriques.

Défi 14 : une fois le graphe construit, le défi 12 devient beaucoup plus simple. Les élèves vont voir la réponse clairement dans les graphes.

Lors de la phase suivante, les élèves repassent au jeu avec 10 anneaux (5 anneaux noirs et 5 anneaux blancs).

Défi 15* : ce défi est optionnel. L'idée du défi est de montrer aux élèves que les exercices de comptage deviennent compliqués à mesure que la complexité du problème augmente. Beaucoup d'élèves ne s'en rendent pas forcément compte, car compter est très simple quand il y a peu d'objets.

La représentation graphique du jeu Involution© est donnée, mais elle est trop grande pour chercher le plus court chemin à l'œil nu.

À la recherche d'un algorithme

Pour faciliter la recherche d'un algorithme, les élèves passent à 6 anneaux (3 noirs et 3 blancs). Le graphe d'Involution© à 6 anneaux est assez petit pour chercher le plus court chemin à la main (contrairement au graphe d'Involution© à 10 anneaux), mais assez grand pour que la solution ne soit pas ridicule (contrairement à Involution© à 5 anneaux).

Défis 16 et 17 : ces défis servent à mettre les élèves sur la voie d'un algorithme. Ces défis sont résolus en binômes. Les résultats sont ensuite discutés en classe sous forme de débat dirigé.

Défi 18 : dans ce défi, l'algorithme est établi à la main. Les élèves doivent rédiger des instructions informelles. Le point b) est plus complexe. Il est destiné seulement aux élèves motivé·e·s.

Pour dessiner l'organigramme, les élèves ont le choix de le dessiner de zéro, ou d'utiliser l'aide qui leur fournit déjà les champs. Les élèves doivent ensuite mettre les champs dans le bon ordre et ajouter les flèches pour former un organigramme.

Défi 19 : ce défi ne sert qu'à montrer aux élèves que l'algorithme général est exactement le même que celui qu'il·elle·s viennent d'établir. Ce défi est à faire directement en débat dirigé.

Défi 20 : en débat dirigé, les élèves réalisent que l'algorithme qu'il·elle·s viennent d'établir est un algorithme de recherche du plus court chemin. Les mêmes types d'algorithme sont utilisés par les navigateurs GPS (voir 1.6.02. L'importance de la théorie des graphes dans les sciences numériques).

06 Possibilités de différenciation

Pour les débutants

Le module est tout à fait faisable en laissant tomber les défis et la section munis d'une étoile. Encouragez les élèves de faible niveau à faire le module sans ces défis.

Pour les plus motivé·e·s

Incluez les défis munis d'une étoile (et la section munie d'une étoile) pour les élèves plus motivé·e·s. Ces défis sont de nature un peu plus mathématique et incluent des raisonnements logiques. Il existe également un défi muni de 2 étoiles. Il est uniquement destiné aux élèves de très bon niveau qui accrochent bien au sujet.

Dans les défis 2 et 11, l'enseignant·e peut constituer les trois groupes de manière à ce que le groupe traitant le cas le plus simple soit constitué d'élèves d'un niveau plus faible.

Le défi 18, c'est-à-dire le défi principal où l'algorithme est établi, possède trois niveaux de difficulté :

1. L'établissement d'un algorithme qui ne compte que le nombre minimal de mouvements et où les champs de l'organigramme sont déjà donnés.
2. Idem mais sans les champs de l'organigramme.
3. L'établissement d'un algorithme qui compte le nombre minimal de mouvements et la suite des mouvements.

07 Autres critères à remplir dans le cadre de la série des unités

a. **Contexte luxembourgeois :** le jeu Involution© a été inventé par Hugo Parlier et Bruno Teheux, deux chercheurs en mathématiques de l'Université du Luxembourg. Dans sa forme actuelle, il peut être rattaché aux contenus du Guide de référence pour l'éducation aux et par les médias luxembourgeois et correspond à l'axe thématique I de *Digital Sciences*.
b. **Différenciation :** plusieurs défis du module sont optionnels. Les défis marqués par une étoile sont plus compliqués que les autres et peuvent être utilisés, ou pas, selon le niveau de la classe et/ou des élèves.
c. **Guide de référence pour l'éducation aux et par les médias :** voir les objectifs d'apprentissage visés par le guide de référence dans la section de l'analyse didactique du présent document.
d. **Modèle des 4 C : communication, collaboration, créativité, pensée critique :** Le modèle 4C est appliqué de diverses manières à travers les différentes formes sociales et activités d'enseignement.
e. **Lien avec la recherche actuelle :** les défis d'Involution© font partie de la famille des problèmes de reconfiguration (Combinatorial reconfiguration), un domaine de recherche actuel qui est étudié en mathématiques et en informatique. Les algorithmes de plus court chemin ont plus de 50 ans, mais font toujours l'objet de recherches afin d'optimiser les applications de recherche de plus court chemin.
f. **Lien avec la recherche au Luxembourg :** le jeu Involution© a été développé par deux chercheurs de l'Université du Luxembourg. Dans le podcast de la section 1.7 Parole aux scientifiques, Hugo Parlier explique comment il développe des jeux en utilisant les résultats de ses recherches.

Références :
Institut national de recherche en sciences et technologies du numérique (INRIA). (2020). *Éducation et Numérique : enjeux et défis.* Livre Blanc N 04. https://hal.inria.fr/hal-03051329v2/document
Han, Sheon. (2022). How to Write Software With Mathematical Perfection. Quantamagazine. https://www.quantamagazine.org/computing-expert-says-programmers-need-more-math-20220517

Sujet de l'unité : Découverte des algorithmes via un jeu

Objectifs d'apprentissage et compétences à développer au cours de l'unité : Les élèves connaissent les bases de la résolution de problèmes algorithmiques et les ont testées sur un jeu réel. Il·elle·s ont ainsi développé des stratégies de résolution de manière coopérative et y ont réfléchi de manière métacognitive à la base d'une documentation autonome.

Défis OPTIONNELS ET FACULTATIFS : Dans l'aperçu suivant, une distinction est faite entre les tâches optionnelles et facultatives. Selon la vitesse à laquelle les élèves travaillent sur les différentes phases, il est donc possible de décider spontanément si certaines tâches doivent être omises ou ajoutées.

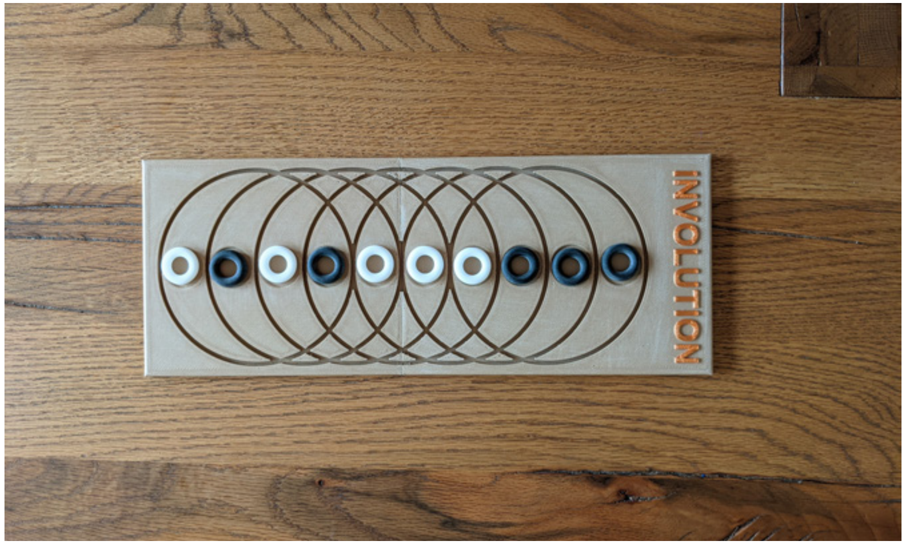
Évaluation (possible) : Plusieurs possibilités d'évaluation sont proposées en 1.5 Idées d'évaluation avec des niveaux de difficulté différents

Durée	Phases	Focus	Formes Sociales/Méthodes	Matériels et Supports	Processus d'apprentissage
10min	Entrée en matière	**Apprendre le fonctionnement du jeu Involution©** **Défi 1**	• Groupes de deux • Cours magistral • Travail en binômes	• Involution© • Fiches de travail avec instructions et défis (en version imprimée ou sur tablette)	Les élèves ... **comprennent** le concept et le mode de fonctionnement du jeu Involution.
20min	Involution©	**[OPTIONNEL] Élaboration** **Défi 2 & 3**	• Travail en binômes	• Involution© • Fiches de travail avec instructions et défis (en version imprimée ou sur tablette)	Les élèves ... **élaborent** ensemble une solution aux défis proposés. ... **réfléchissent** à leurs stratégies et leurs solutions.
20min	Involution© Une solution optimale	**Élaboration** **Défi 10**	• Travail en binômes	• Involution© • Fiches de travail avec instructions et défis (en version imprimée ou sur tablette)	Les élèves ... **élaborent** ensemble une solution aux défis proposés. ... **réfléchissent** à leurs stratégies et leurs solutions.
25min	Involution© À la recherche d'une solution optimale	**Élaboration** **Défis 11-15** **[OPTIONEL: Défis 12, 14, 15]**	• Travail en binômes	• Involution© • Fiches de travail avec instructions et défis (en version imprimée ou sur tablette)	Les élèves ... **élaborent** ensemble une solution aux défis proposés. ... **réfléchissent** à leurs stratégies et leurs solutions.
25min	Involution© À la recherche d'un algorithme	**Élaboration** **Défis 16-20**	• Travail en binômes	• Involution© • Fiches de travail avec instructions et défis (en version imprimée ou sur tablette)	Les élèves ... **élaborent** ensemble une solution aux défis proposés. ... **réfléchissent** à leurs stratégies et leurs solutions.

1.3 Matériels pédagogiques

01 | Le jeu

Dans ce module, nous allons jouer à un jeu appelé Involution©. Le jeu est composé d'un plateau et d'une manivelle. Le plateau contient 10 cavités qui contiennent chacune 1 anneau (d'une couleur).



Les anneaux peuvent être déplacés comme le montre la vidéo suivante :

Involution 1

Défi 1

Mettez-vous en binômes et jouez à Involution© : essayez de résoudre les défis proposés sur les cartes.

Défi 2

En partant de la configuration de gauche, est-il possible d'arriver à la configuration de droite ?

○○○○○●●●●●	●○●○●○●○●○
○○○○○●●●●●	●●●●○●○○○○
○○○○○●●●●●	●●●○●●○○○○

Si oui, expliquez vos mouvements. Sinon, pourquoi ?

Défi 3*

Peut-on atteindre toutes les configurations d'anneaux ? Si oui, pourquoi ? Si non, lesquelles sont impossibles ? Justifiez votre réponse.

02 | Comment obtenir toutes les configurations ?

Pour y voir plus clair, essayons maintenant de jouer au même jeu, mais avec 9 anneaux blancs et un anneau noir.

La question essentielle est la suivante.

Défi 4*

Est-il possible d'atteindre la configuration suivante à partir de n'importe quelle autre configuration ? *Justifiez* votre réponse.

●○○○○○○○○○

Jouons maintenant avec deux anneaux noirs.

Défi 5*

Est-il possible d'atteindre la configuration suivante à partir de n'importe quelle autre configuration ? Utilisez votre solution du défi 4 et *justifiez* votre réponse.

●●○○○○○○○○

Jouons maintenant avec trois anneaux noirs.

Défi 6*

Utilisez la solution du défi 5 pour déterminer si on peut atteindre la configuration suivante

●●●○○○○○○○

à partir de n'importe quelle autre configuration.

Pour finir, jouons avec quatre anneaux noirs.

Défi 7*

Utilisez la solution du défi 6 pour déterminer si on peut atteindre la configuration suivante

●●●●○○○○○○

à partir de n'importe quelle autre configuration.

Défi 8*

Reconsidérez le défi 3*.

Défi 9**

Si nous jouons au même jeu, mais avec 12 anneaux (6 noirs et 6 blancs), est-il possible d'obtenir toutes les configurations ? Et avec 100 anneaux (50 noirs et 50 blancs) ?

03 | Une solution optimale

Revenons au jeu avec 10 anneaux.

Défi 10

En partant de la configuration suivante,

○○○○○●●●●●

trouvez la configuration suivante

●●●●●○○○○○

et comptez le nombre de mouvements.

Comparez vos solutions entre vous. Laquelle est la meilleure ? *Définissez un critère* pour évaluer quelle solution est meilleure qu'une autre.

Après discussion, essayez de compléter la définition suivante :

Une solution est optimale si

04 | A la recherche d'une solution optimale

Nous allons maintenant construire une représentation visuelle de notre jeu afin de chercher une solution optimale. Rappelons-nous qu'au cours de *Digital Sciences 1*, un des premiers algorithmes que nous avions rencontrés était un algorithme pour retrouver un itinéraire avec le métro de New York. Tout comme le plan du métro qui est une représentation graphique du réseau réel, nous allons réaliser une représentation graphique du jeu : les nœuds du réseau sont les différentes configurations du jeu (dans le cas du métro, les nœuds étaient les stations) et deux configurations sont reliées par une ligne si on peut passer d'une configuration à l'autre par un mouvement (dans le cas du métro, deux stations sont reliées par une ligne si l'on peut aller d'une station directement à l'autre via le métro).

Prenons un exemple. Jouons à Involution©, mais avec seulement 5 anneaux et un plateau de jeu réduit (prenez le même plateau, mais sans utiliser les 5 cavités les plus à droite). Vous n'avez donc droit qu'à deux mouvements distincts de manivelle.

Défi 11

La classe se divise en 3 groupes.

- Le premier groupe joue avec 1 anneau noir (et 4 anneaux blancs).
- Le deuxième groupe joue avec 2 anneaux noirs (et 3 anneaux blancs).
- Le troisième groupe joue avec 3 anneaux noirs (et 2 anneaux blancs).

Par groupe de deux, dessinez toutes les configurations possibles de votre version d'Involution©.

Discutez et *expliquez* ensemble combien de configurations de votre version d'Involution© sont possibles. *Présentez* ensuite votre solution et votre explication à l'ensemble de la classe.

Défi 12*

Décidez si, dans votre cas, il est possible d'obtenir n'importe quelle configuration et *expliquez* votre réponse.

Nous allons maintenant construire la représentation graphique.

Défi 13

Donnez un nom à chaque configuration (par exemple C1, C2, C3, etc.). Pour chaque configuration trouvée dans le défi 11, dessinez un nœud et notez son nom à côté. Puis reliez deux nœuds par une ligne s'il est possible de passer d'une configuration à l'autre par un seul mouvement de manivelle. Comparez vos graphes.

Défi 14

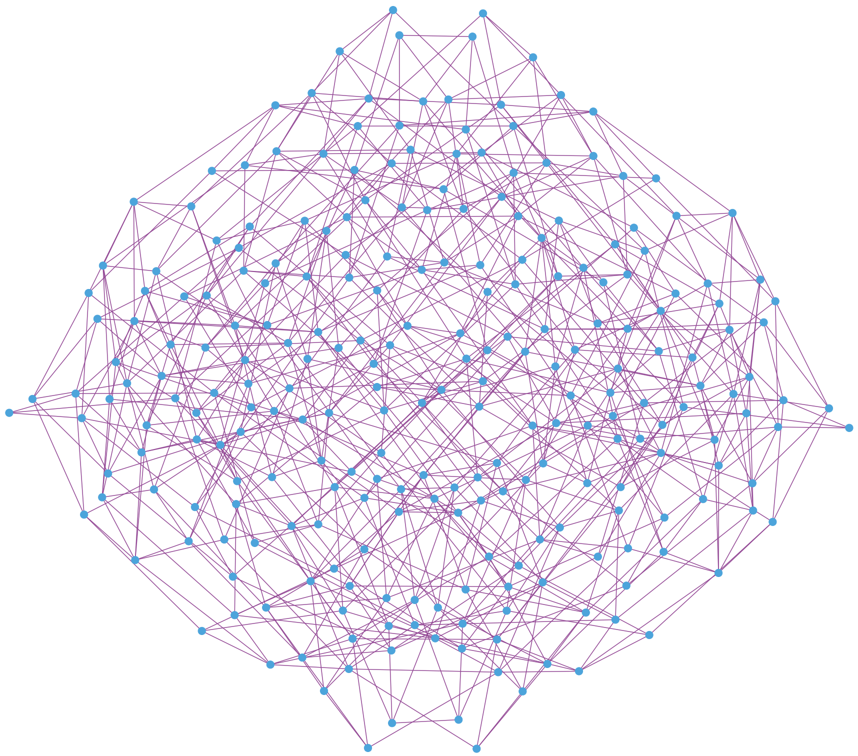
Reprenez le défi 12 (en utilisant le graphe que vous venez de construire).

Revenons maintenant au jeu initial avec 10 anneaux.

Défi 15*

Savez-vous calculer le nombre de configurations différentes ou donner au moins une borne supérieure ?

En tout, il y a 252 configurations différentes. Ceci donne donc un graphe à 252 sommets, qui ressemble à ceci :



Le graphe d'Involution© avec 10 anneaux est beaucoup trop grand pour voir le chemin le plus court à l'œil nu. C'est pourquoi nous avons besoin d'un algorithme.

05 | A la recherche d'un algorithme

Défi 16

Reprenez votre jeu et jouez maintenant avec 6 anneaux (3 noirs et 3 blancs) et un plateau de jeu réduit (prenez le même plateau, mais sans utiliser les 4 cavités les plus à droite). Trouvez toutes les configurations possibles et dessinez le graphe.

Défi 17

Utilisez le graphe trouvé au défi 16 pour déterminer une solution optimale pour arriver à la configuration ci-dessous

○●○●○●

en partant de la configuration suivante :

○○○●●●

The Big Bang Theory Sheldon's Friendship Alogrithm

Défi 18

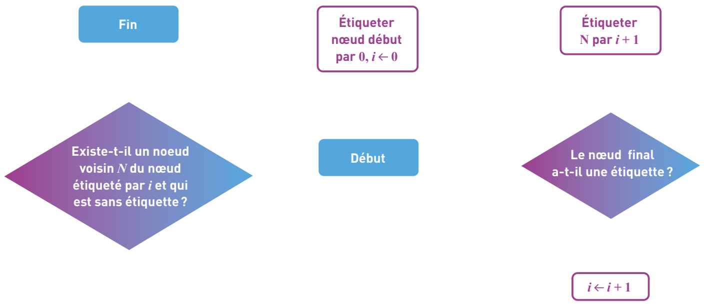
En vous inspirant du défi précédent, cherchez un algorithme qui donne

1. le nombre minimal de mouvements pour aller d'une configuration à une autre (une fois que le graphe du jeu est donné).

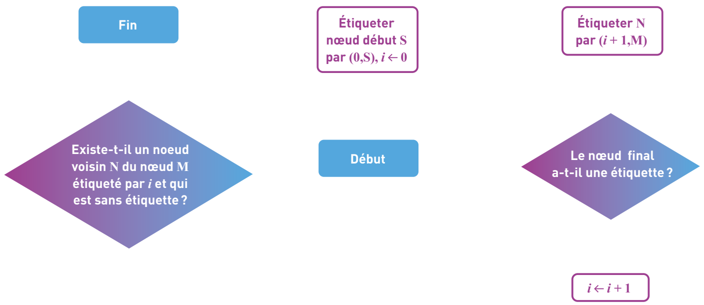
2. une solution optimale pour aller d'une configuration à une autre (une fois que le graphe du jeu est donné).

Inscrivez les instructions informelles de votre algorithme et dessinez son organigramme de programmation.

Aide 1 :



Aide 2:

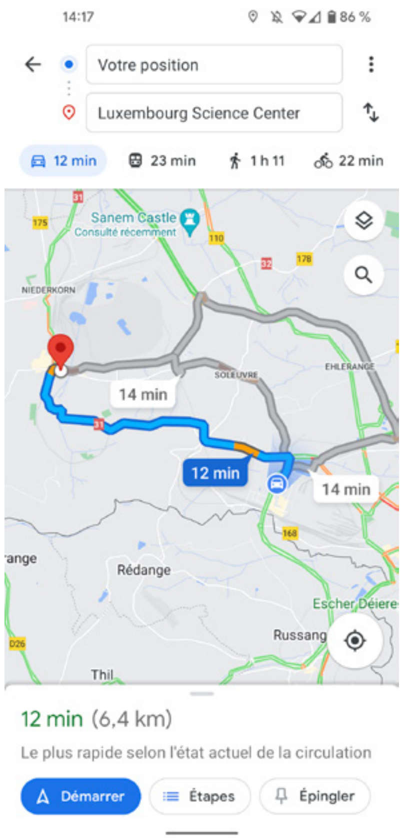


Défi 19

Généralisez l'algorithme au jeu Involution© avec 10 anneaux.

Défi 20

Regardez l'image suivante :



De quelle application s'agit-il? Cherchez un lien entre cette application et la recherche d'une 15

1.4 Idées interdisciplinaires

Mathématiques

Ce module vise exactement les quatre compétences relatives aux processus mathématiques énumérées dans les documents *Compétences disciplinaires attendues à la fin de la classe de 6e et à la fin de la classe de 4e* et *Compétences disciplinaires attendues à la fin de la classe de 7G-6G-5G* émis par le ministère de l'Éducation nationale. Ces quatre compétences sont les suivantes :

1. Résolution de problèmes : comme expliqué dans le document des compétences disciplinaires, la résolution des problèmes mathématiques « est caractérisée d'un côté par la mise en œuvre de stratégies générales et de l'autre par la mise en œuvre de stratégies spécifiques ». C'est exactement ce que les élèves sont amené·e·s à faire dans ce module : il·elle·s commencent par découvrir le jeu en testant quelques exemples, puis il·elle·s suivent la stratégie donnée par le module. Ce module constitue un bel exemple pour démontrer que l'on s'entraîne à la résolution de problèmes en « travaillant activement sur des problèmes et en réfléchissant sur les méthodes et les stratégies de résolution ».
2. Modéliser : dans ce module, les élèves sont d'abord amené·e·s à simplifier le problème, puis à le modéliser sous forme de graphe. Ce processus correspond exactement à la description de la compétence modéliser : « Il s'agit de simplifier d'abord la situation réelle et de la mathématiser ensuite, c.-à-d. de la décrire avec des outils mathématiques ».
3. Argumenter : cette compétence est décrite comme suit : « L'argumentation mathématique commence par l'exploration de situations, par la recherche de structures et de relations et par la formulation de conjectures sur des relations mathématiques ». C'est exactement ce qui se passe dans les défis 3 à 8 dans 1.3 Matériels pédagogiques. Dans le premier exercice, il est demandé aux élèves de formuler une conjecture, puis dans les exercices suivants, il·elle·s sont amené·e·s à la preuve de leur conjecture.
4. Communiquer : les exercices proposés dans ce module ne sont pas des exercices typiques d'un cours de mathématiques, mais nécessitent un raisonnement et une argumentation afin d'expliquer le raisonnement. Les élèves doivent donc « exposer des contenus mathématiques de manière adaptée à l'aide du langage courant et du langage mathématique ».

Pour être plus précis, ce module permet de travailler sur une compétence du chapitre calcul littéral du nouveau programme provisoire de mathématiques de la classe de 6eC. Cette compétence se lit comme suit : décoder une expression littérale en une série ordonnée d'instructions de calcul et savoir formaliser cet algorithme.

Le jeu Involution© peut également être utilisé pour illustrer les concepts de symétries centrales et de rotations. Le mouvement de la manivelle du jeu n'est rien d'autre qu'une symétrie centrale ou une rotation de 180 degrés. Ces transformations géométriques sont traitées en 5e et 4e et posent souvent problème aux élèves : la symétrie axiale leur paraît beaucoup plus naturelle et ils·elle·s ont du mal à visualiser ces nouvelles transformations. Involution© constitue un autre moyen de visualisation.

Dans le secondaire technique inférieur, les élèves commencent déjà à se familiariser avec les probabilités. Involution permet de faire des comptages et des dénombrements (combien de configurations différentes existe-t-il ?) et de calculer des probabilités de multiples manières.

Enfin, ce module prépare le terrain à la preuve par induction. Dans les défis 4 à 8 de 1.3 Matériels pédagogiques les élèves sont amené·e·s, étape par étape, à faire une preuve par induction (sans qu'on leur parle de ce terme mathématique).

Géographie

Via les algorithmes de navigation (comme Googlemaps), on peut faire le lien avec les cartes utilisées par les applications de navigation. L'orientation et les cartes figurent au programme de géographie de 7eC et 7eG.

1.5 Idées d'évaluation

Il est possible de compléter la discussion en groupe sur les méthodes de résolution des différents défis à l'aide d'Involution© par une évaluation. Elle contribue à la consolidation des connaissances et compétences acquises.

Créer un livre de solutions pour Involution© à 3 couleurs

Jouons à Involution© avec 3 couleurs : 2 anneaux noirs, 2 anneaux blancs et 1 cavité sans anneau (la cavité sans anneau joue le rôle de troisième couleur). Les élèves sont chargé·e·s de rédiger un livret de solutions optimales pour un nombre de défis déterminé (par l'enseignant·e). Le choix de la présentation est à la disposition des élèves – il peut s'agir de simples descriptions de processus, de dessins, de représentations purement visuelles ou d'une vidéo produite par les élèves.

Pour trouver les solutions optimales, les élèves doivent construire le graphe correspondant à Involution© avec 3 couleurs et chercher le chemin le plus court dans ce graphe (attention : ce chemin n'existe peut-être pas toujours).

Les livrets de solutions terminés peuvent ensuite être échangés entre les élèves et leur exactitude vérifiée à l'aide du jeu. Les élèves peuvent ainsi s'évaluer mutuellement et donner un feed-back.

Il est aussi possible de réaliser des projets plus larges et plus complexes. Nous en proposons ici trois différents : le premier est de nature plus informatique (et demande de la programmation), le deuxième s'inscrit plutôt dans le domaine des sciences sociales et de la créativité et demande des facultés de rédaction, le troisième est destiné aux élèves motivé·e·s par les mathématiques.

Programmer Involution©

Le but est d'écrire un programme (sur Scratch ou Python), en utilisant l'algorithme établi aux défis 15 et 16 pour résoudre le jeu Involution©.

Plusieurs niveaux de difficulté se présentent :

Écrire un programme qui prend comme input deux configurations du jeu Involution© et donne comme output le nombre minimal de mouvements pour aller d'une configuration à l'autre.

Écrire un programme qui prend comme input une configuration du jeu Involution© et donne comme output une suite de mouvements (pas nécessairement la suite correspondant à la solution optimale) permettant d'aller de cette configuration à la configuration où tous les anneaux blancs sont d'un côté et tous les anneaux noirs de l'autre.

Écrire un programme qui prend comme input une configuration du jeu Involution© et donne comme output la suite optimale de mouvements permettant d'aller de cette configuration à la configuration où tous les anneaux blancs sont d'un côté et tous les anneaux noirs de l'autre.

Inventer un jeu pour deux joueurs

Les élèves doivent réfléchir à une version pour jouer à Involution© à deux joueurs. Il·elle·s doivent inventer un jeu collaboratif et un jeu compétitif. Il·elle·s sont ensuite invité·e·s à rédiger des règles du jeu claires. Le choix de la présentation est laissé aux élèves (petit livret, dessins, support numérique, etc.).

Et avec 3 couleurs ?

Jouons à Involution© avec 3 couleurs : n anneaux noirs, n anneaux blancs et 1 cavité sans anneau (la cavité sans anneau joue le rôle de troisième couleur). La valeur n peut prendre n'importe quelle valeur supérieure ou égale à 2. La question que les élèves doivent résoudre est la suivante : à partir de quelle valeur de n peut-on obtenir toutes les configurations à partir de n'importe quelle autre configuration ?

1.6 Pour aller plus loin

01 | Les algorithmes BFS et Dijkstra

L'algorithme que les élèves viennent de découvrir dans ce module est un algorithme de recherche du plus court chemin. Le nom technique de cet algorithme est *Breadth-First Search algorithm* ou simplement *BFS algorithm*. En français, on parle aussi d'algorithme de parcours en largeur. Inventé en 1945, il permet de parcourir un graphe en commençant par explorer un nœud source, puis ses successeurs, puis les successeurs non explorés des successeurs, etc. L'algorithme de parcours en largeur sert à calculer les distances de tous les nœuds depuis un nœud source dans un graphe. Le mode de fonctionnement du BFS utilise une file dans laquelle il prend le premier sommet et place en dernier ses voisins non encore explorés. Les nœuds déjà visités sont marqués afin d'éviter qu'un même nœud ne soit exploré plusieurs fois. Cet algorithme fonctionne seulement sur les graphes non pondérés, c.-à-d. les graphes dans lesquels les nœuds sont connectés par des arêtes, toutes égales non pondérées. C'est bel et bien le cas des graphes de ce module : deux nœuds (ici des configurations) sont reliés par une arête si l'on peut passer d'une configuration à l'autre par un seul mouvement de manivelle. Par contre, le BFS fonctionne pour les graphes orientés et non orientés. Un graphe orienté est un graphe dans lequel les arêtes entre deux nœuds sont orientées, c.-à-d. qu'elles vont d'un nœud vers un autre (Diestel, 2018), ou ont un sens de parcours. Comme le jeu Involution© est symétrique (si l'on peut passer d'une configuration A à une configuration B avec un seul mouvement, on peut passer de B à A avec le mouvement inverse, qui est en effet exactement le même mouvement), les graphes explorés dans ce module sont non orientés. Le département d'informatique de l'Université de Cornell a publié une introduction à l'algorithme BFS via des vidéos qui durent en tout 15 minutes (Gries, n.d.).

En dehors des jeux, les graphes doivent être pondérés. Le graphe du réseau routier que nous avons vu dans ce module est un graphe pondéré : la distance entre deux carrefours varie d'une situation à une autre. Le graphe doit donc en tenir compte en attribuant un poids aux arêtes. Le plus court chemin n'est donc plus forcément le chemin qui parcourt le moins de nœuds possible et le BFS ne fonctionne donc pas sur les graphes pondérés.
L'algorithme de Dijkstra est un algorithme connu qui donne le plus court chemin pour les graphes pondérés et non pondérés, qu'ils soient orientés ou non orientés. Il porte le nom de son inventeur, Edsger Dijkstra, et a été publié pour la première fois en 1959 (Dijkstra, 1959). Le mode de fonctionnement de l'algorithme de Dijkstra est le suivant : l'algorithme commence à un nœud de départ et calcule les distances entre ce nœud et tous les autres nœuds du graphe. Il choisit ensuite le nœud le plus proche du nœud de départ. Ce nœud est ajouté avec le nœud de départ dans une file, puis toutes les distances sont réévaluées : si la distance entre le nœud de départ et un autre nœud est plus courte en passant par le nouveau nœud, elle est remplacée par la distance la plus courte. L'algorithme continue ainsi jusqu'à ce qu'il ait parcouru tous les nœuds, ce qui permet d'établir les distances minimales entre le nœud de départ et tous les autres nœuds. Le département d'informatique de l'Université de Cornell a aussi publié une introduction à l'algorithme de Dijkstra (Gries, n.d.).

02 | L'importance de la théorie des graphes dans les sciences numériques

À plusieurs reprises, le mathématicien J. H. C. Whitehead a déclaré que « la combinatoire est le bidonville de la topologie » (Cameron, 2011). La combinatoire est une discipline mathématique dont fait partie la théorie des graphes. Au début du XXe siècle, cette vision de la théorie des graphes est répandue parmi beaucoup d'autres mathématiciens, mais elle a fortement, voire complètement, changé aujourd'hui. La théorie des graphes est un outil important pour étudier les liens dans les systèmes complexes. Dès qu'un système peut être représenté schématiquement sous forme de nœuds et d'arêtes, la théorie des graphes s'applique. Ces systèmes peuvent être très divers : réseaux routiers, réseaux urbains des données informatiques... Les applications dans le monde numérique d'aujourd'hui sont énormes. Voici quelques exemples très importants (Flovik, 2020) :

- La propagation du virus COVID 19 dans les communautés
- Le classement des pages web dans les moteurs de recherche (comme Google)
- La sécurité du réseau
- Les navigateurs GPS
- etc

Revenons par exemple aux navigateurs GPS (comme Google maps, Waze, etc.), qui interviennent aussi dans ce module. Les systèmes de navigation GPS font partie des applications de la théorie des graphes qui ont un impact direct sur notre vie quotidienne : nous nous fions bien souvent à ces systèmes pour rejoindre un lieu en empruntant le chemin le plus court (ou le plus rapide). Un réseau routier peut être représenté par un graphe dont les nœuds sont les carrefours et les immeubles, et dont les arêtes sont les portions de route qui les relient. Comme certaines routes sont en sens unique, le graphe est orienté (contrairement au graphe d'Involution©, qui est non orienté). Le système de navigation calcule un chemin optimal entre deux points du réseau, en fonction de critères définis par l'utilisateur (il optimise soit la longueur du trajet, soit le temps de parcours ou favorise les autoroutes, etc.). Pour fixer les idées, imaginons que le système cherche à minimiser la longueur des trajets.
Pour modéliser le problème, nous devons ajouter de l'information au graphe orienté qui modélise le réseau routier. On associe en effet à chaque portion de route un poids qui représente exactement la longueur de cette portion. Les navigateurs utilisent donc des graphes orientés et pondérés. L'algorithme de recherche du plus court chemin que les navigateurs utilisent est basé sur l'algorithme de Dijkstra (Teheux, 2019).

Il est intéressant de noter que les systèmes de navigation GPS implémentent l'algorithme de Dijkstra à l'envers. Plus précisément, si le but de l'utilisateur est de se rendre du nœud A au nœud B, le système de navigation applique l'algorithme de Dijkstra au graphe obtenu en retournant les arêtes du graphe du réseau routier avec le nœud B comme nœud source. Le résultat est alors le chemin le plus court de B vers A. Mais il suffit de retourner ce chemin pour obtenir le chemin souhaité. L'idée qui sous-tend cette inversion est une économie de calculs. Il n'est pas rare que nous, en tant que conducteurs, quittions l'itinéraire recommandé par le navigateur (nous nous trompons, une route est fermée à la circulation, nous changeons de plan à la dernière minute, etc.). Si l'algorithme avait effectué la recherche du plus court chemin dans le bon sens, il devrait recommencer (car nous nous trouvons à un endroit qui n'était pas prévu dans les calculs de l'algorithme) et tous les calculs faits avant auraient été perdus. Par contre, en commençant par la fin, l'algorithme a déjà calculé le plus court chemin de n'importe quel point (dans un rayon raisonnable) au point final. En conséquence, même en changeant d'itinéraire, l'algorithme est capable de nous proposer rapidement une alternative (Teheux, 2019).

Bien que la théorie des graphes pure soit une discipline abstraite et mathématique, ceci montre qu'elle a beaucoup d'applications concrètes et intéressantes en informatique. De toute façon, comme le confirme l'informaticien mondialement reconnu Leslie Lamport : « les mathématiques font partie de la pensée informatique » (Han, 2022).

Références :
Cameron, Peter J. (2011). Aftermath. Preprint. https://arxiv.org/pdf/1111.4050.pdf
Diestel, Reinhard. (2018). *Graph theory (5th Ed.).* Graduate Texts in Mathematics, 173. Berlin : Springer.
Dijkstra, Edsger W. (1959). A note on two problems in connexion with graphs. *Numer. Math.*, 1, 269 – 271. https://doi.org/10.1007/BF01386390
Flovik, Vegard. (2020). What is Graph Theory, and why should you care. From graph theory to path optimization. *Towards Data Science.* https://towardsdatascience.com/what-is-graph-theory-and-why-should-you-care-28d6a715a5c2
Gries, David. (n.d.). *Depth-first search and breadth-first search.* https://www.cs.cornell.edu/courses/JavaAndDS/dfs/dfs01.html
Gries, David. (n.d.). *The shortest-path algorithm.* https://www.cs.cornell.edu/courses/JavaAndDS/shortestPath/shortestPath.html
Han, Sheon. (2022). How to Write Software With Mathematical Perfection. *Quantamagazine.* https://www.quantamagazine.org/computing-expert-says-programmers-need-more-math-20220517
Teheux, Bruno. (2019). À la recherche des chemins les plus courts. *Losange*, N 46, 45-54.

1.7 La parole aux scientifiques : Podcast avec Dr Hugo Parlier et conversation avec Dr Bruno Teheux

Hugo Parlier est professeur au département de mathématiques de l'Université du Luxembourg. Ses recherches portent sur différents thèmes issus de la géométrie, de la topologie et de la combinatoire. Il a étudié les mathématiques à l'École Polytechnique Fédérale de Lausanne où il a obtenu son doctorat en 2004. Après des séjours postdoctor aux à Madrid, Toronto et Genève, il est devenu professeur associé à Fribourg, avec le soutien du Fonds National Suisse. En 2017, il a rejoint le département de mathématiques de l'Université du Luxembourg.

Bruno Teheux est chercheur permanent au département de l'Université du Luxembourg et fait des recherches en logique mathématique. Il a obtenu son doctorat en 2009 à l'Université de Liège, en Belgique. Après avoir travaillé pendant deux ans à Animath à Paris, il a rejoint l'Université du Luxembourg en 2012.

Les deux mathématiciens adorent partager leur passion pour les mathématiques au travers d'activités destinées au grand public. Ce sont les instigateurs des projets The Simplicity of Complexity et Recreate: shapes from the collective Imagination, qui ont été présentés à l'exposition universelle de Dubaï en 2020. Ils participent également au projet The Sound of Data dans le cadre de Esch2022, capitale européenne de la culture.

Outre ces projets communs, Bruno Teheux a édité le volume pilote de BD scientifiques scénarisées par des doctorants·e·s. Quant à Hugo Parlier, il a co-écrit un livre interactif de mathématiques pour iPad appelé Mathema. Ce livre a donné naissance à des casse-têtes pour iPad ou iPhone appelés Quadratis.

Théorie des graphes avec Bruno Teheux et Ann Kiefer

Malheureusement, Hugo Parlier n'a pas pu participer à notre interview. C'est pourquoi nous vous renvoyons vers l'excellent podcast « Mäin Element » avec Hugo Parlier. Ce podcast est fait par le Lëtzebuerger Journal en collaboration avec le Fonds National de la Recherche (FNR).

Podcast Mäin Element avec Hugo Parlier

ISSN 2738-9383