

# LES SCIENCES INFORMATIQUES ET LES AUTRES DISCIPLINES

## TABLE DES MATIERES

Introduction .....	1
Les sciences informatiques pour modéliser, pour clarifier les règles : application dans un cours de langue.....	2
Les sciences informatiques pour mobiliser les savoirs : application dans un cours de mathématiques .....	5
Les sciences informatiques pour aider à la « décision » : application dans un cours de biologie .....	8
Les sciences informatiques au cœur des nouvelles technologies : application dans un cours d'éducation par technologie .....	11
Conclusion.....	13

## INTRODUCTION

Les sciences informatiques en particulier l'algorithmique et la programmation, constituent des voies intéressantes pour structurer le raisonnement. De plus, la pensée algorithmique, basée sur des représentations discrètes, la séquentialité et la rigueur imposée par la machine, offre un mode de résolution de problèmes parfois alternatif aux autres disciplines. Les sciences informatiques ont donc un intérêt en soi dans l'acquisition de compétences de base par l'ensemble de la population (voir document « informatique et enseignement obligatoire »).

L'informatique pousse à modéliser. Les domaines d'application sont multiples car l'informatique manipule des données. Si on peut représenter un phénomène, un concept via des données, on peut les manipuler avec l'informatique. On peut les « simuler » dans un monde virtuel. Ainsi, puisque l'on peut modéliser le mouvement par la position, la vitesse, l'accélération, on peut manipuler ces données en machine et simuler le mouvement. Puisque l'on peut modéliser les réactions chimiques par des atomes qui ont des propriétés définies (valence, masse, ...), on peut manipuler ces données et simuler les réactions, modéliser les liaisons chimiques, la forme des molécules, ... Puisqu'en langue, un mot est une succession de lettres, que chaque mot a une nature et une fonction, on peut manipuler ces lettres pour conjuguer, ... Puisque les mathématiques manipulent des nombres, on peut les représenter en machine, les additionner, les combiner en appliquant diverses règles. Un peu comme un « couteau suisse », les sciences informatiques peuvent servir à « tout ». Avoir des bases dans ce domaine sera utile dans de nombreux métiers et contextes. On peut donc faire appel aux sciences informatiques dans des matières aussi diverses que l'apprentissage des langues, les mathématiques, la biologie, ...

Facile à dire mais ce serait plus convainquant avec quelques exemples à la portée de jeunes au début du secondaire.

Nous allons donc nous attarder sur l'apport des sciences informatiques à l'apprentissage dans d'autres disciplines. Quatre exemples vont être explicités. Nous les avons choisis pour la diversité des thématiques abordées et parce qu'ils illustrent des apports différents à l'apprentissage.

## LES SCIENCES INFORMATIQUES POUR MODÉLISER, POUR CLARIFIER LES RÈGLES : APPLICATION DANS UN COURS DE LANGUE

### CONTEXTE

Prenons tout d'abord un exemple relatif à l'apprentissage des langues. Une des premières règles que tout apprenant doit intégrer est la formation du pluriel des noms. Ces règles comme beaucoup d'autres ont été formalisées par des grammairiens. On peut bien évidemment se contenter de les énoncer et proposer des exercices d'application pour que les élèves puissent apprendre à les appliquer dans divers contextes. On peut également donner quelques exemples de pluriels et demander aux élèves d'inférer la règle comme l'on fait les grammairiens en leur temps. Cela permet aux élèves d'ancrer plus durablement la règle dans la mesure où ils la construisent eux-mêmes.

Collectionner les exemples et en formaliser une règle n'est pas toujours une chose simple pour les élèves. Ce travail parfois fastidieux ne les motive pas toujours.

### BRÈVE DESCRIPTION DE L'ACTIVITÉ

Quel pourrait être l'apport de l'informatique ? Les nouvelles technologies sont un support généralement motivant pour les élèves. L'application de règles, d'instructions, est la base de la programmation. Alors pourquoi ne pas proposer aux élèves de créer directement un programme qui détermine le pluriel d'un nom ?

Avec un langage visuel par blocs comme Blockly ou Scratch, c'est assez simple. Il suffit de partir de la règle de base. Ici en Néerlandais, ajouter « en » à la fin des mots. Ensuite, on se souvient qu'il faut faire attention aux syllabes ouvertes ou fermées. Si la dernière lettre du mot est une consonne précédée de deux voyelles identiques, on retire une voyelle, on doit mettre la consonne puis ajouter « en ». Comme pour « boot » qui devient « boten ». Si la dernière lettre est une consonne précédée d'une seule voyelle, on doit doubler la consonne, puis ajouter « en ». Comme pour « kat » qui devient « katten ». Que faire des nombreuses exceptions ? Il suffit de les lister et d'indiquer la règle à appliquer. Par exemple, si les 2 dernières lettres sont « el », le pluriel est formé en ajoutant « s » comme dans « vogels ».

On a alors explicité ces règles de manière beaucoup plus précise que ce qui est généralement perçu par les élèves. Ils peuvent de plus tester leur programme, vérifier son fonctionnement et ils obtiennent alors une méthode systématique qu'ils pourront eux-mêmes appliquer par après dans des exercices.

La figure 1 montre à quoi pourrait ressembler un programme élémentaire écrit par l'élève dans un langage de programmation visuel. Un autre exemple d'implémentation élémentaire est disponible en Scratch<sup>1</sup> sur <https://scratch.mit.edu/projects/155336096/#editor>

### QUEL APPOINT POUR LA DISCIPLINE CONCERNÉE ?

Pour l'apprentissage des langues, les règles grammaticales sont ainsi construites pas à pas par les élèves. Ils peuvent tester leur programme au fur et à mesure et vérifier s'ils ont bien pensé à tous les cas. Le professeur peut intégrer une série des tests qui attire l'attention des élèves sur des cas particuliers, des exceptions de plus en plus poussées.

---

<sup>1</sup> Ces langages de programmation visuels sont pour la plupart traduits en français, les mots-clés comme « print », « if...do...else... » peuvent être facilement disponibles dans différentes langues. Dans l'exemple en ligne, il suffit de cliquer sur la petite sphère en haut à gauche où on peut choisir « français » ou « nederlands ».

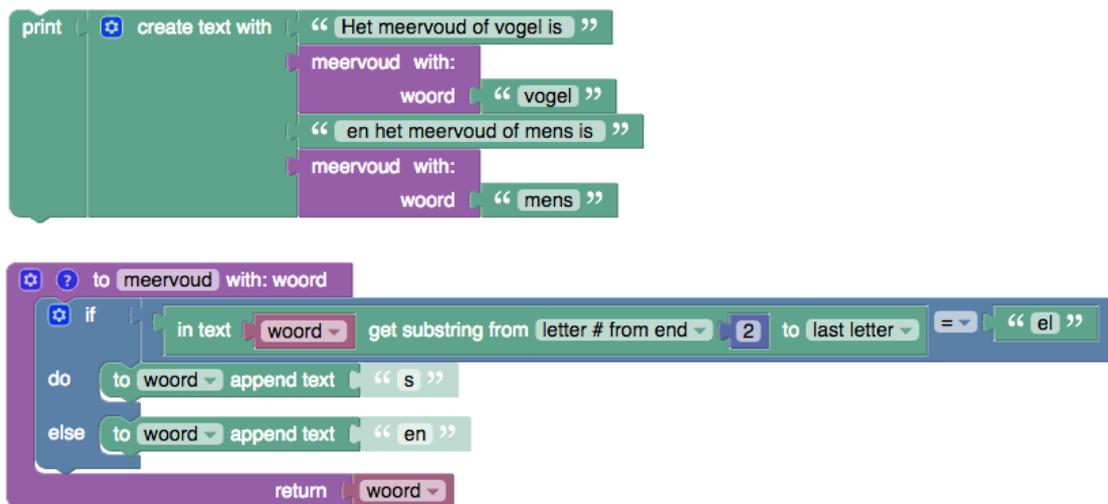


FIGURE 1 : EXEMPLE DE PROGRAMME ÉCRIT EN BLOCKLY PERMETTANT DE CONSTRUIRE LE PLURIEL D'UN NOM EN NÉERLANDAIS. CE PROGRAMME AFFICHE À L'ÉCRAN « HET MEERVOUD OF VOGEL IS VOGELS EN HET MEERVOUD OF MENS IS MENSEN »

#### QUEL APPORT POUR LES COMPÉTENCES TRANSVERSALES ?

Dans de nombreux contextes, l'élève apprend à construire une règle, qui par nature doit être rigoureuse, systématique et prendre en compte un maximum d'exceptions.

La programmation pousse dans cette direction car s'il ne peut pas coder un à un le pluriel de tous les noms, il devra « modéliser » et tenter de trouver des règles génériques. Les cas-tests proposés par l'enseignant peuvent l'aider à trouver des généralités au milieu des exceptions.

De plus, la « bêtise » de la machine oblige l'élève à faire preuve de rigueur dans les instructions qu'il donne.

Au terme de l'activité, l'élève a une réalisation concrète: un programme qu'il a écrit et qui est capable de générer le pluriel des noms. En termes de motivation, c'est un réel plus.

Par ailleurs, l'élève passe du stage de consommateur d'outils technologiques à celui de producteur. Il découvre l'envers du décor de toutes les applications qu'il utilise tous les jours.

#### QUEL APPORT POUR LES SCIENCES INFORMATIQUES ?

Au travers de cette activité, l'élève perçoit qu'un programme est constitué d'instructions qui doivent être précises. Ces instructions s'exécutent de manière séquentielle.

Il a également utilisé divers éléments de base de la programmation

- structure de contrôle conditionnelle (SI ... ALORS ... SINON ...);
- des variables;
- des fonctions;
- des listes.

## QUELS SONT LES PRÉREQUIS EN SCIENCES INFORMATIQUES ?

Pour réaliser rapidement (de l'ordre de 2 périodes) cette activité dans un cours de langue, il faut que les élèves connaissent les bases de la programmation que sont

- les instructions ;
- les structures de contrôle ;
- les variables ;
- les listes.

Les instructions qu'ils manipulent les chaînes de caractères peuvent quant à elles être présentées au début de l'activité.

## LES SCIENCES INFORMATIQUES POUR MOBILISER LES SAVOIRS : APPLICATION DANS UN COURS DE MATHÉMATIQUES

### CONTEXTE

On a tous appris ce qu'était une PGCD (plus grand commun diviseur). On en apprend la définition en primaire. On en découvre les propriétés au secondaire.

$$PGCD(a, b + a) = PGCD(a, b)$$

$$PGCD(a + b, b) = PGCD(a, b)$$

En mathématiques, c'est important d'apprendre au 1<sup>e</sup> degré du secondaire à ne plus simplement manipuler des nombres mais également des expressions littérales, d'exprimer mathématiquement une propriété. Cela mène progressivement à l'abstraction qui sera bien nécessaire quand on devra découvrir les dérivées et les intégrales en fin de secondaire.

Mais toutes ces propriétés même si on peut être persuadé qu'elles sont justes ne paraissent pas bien utiles. On les apprend parce qu'il le faut et pas parce qu'on y voit un intérêt.

Les sciences informatiques peuvent aider à en montrer l'intérêt.

### BRÈVE DESCRIPTION DE L'ACTIVITÉ

De nombreux algorithmes permettent de déterminer le pgcd.

Le plus immédiat pour les élèves consistera probablement à déterminer tous les diviseurs des deux nombres et d'identifier le plus grand diviseur commun : une application directe de la définition.

Mais ils réaliseront vite qu'il n'est déjà pas évident de déterminer si un nombre est diviseur d'un autre. On a bien des règles de divisibilités pour 2, 3, 4, 5, 10 et quelques autres nombres mais pas pour tous. On peut alors les mettre sur la voie en leur rappelant que si un nombre est diviseur d'un autre, le reste de la division (modulo) est nul.

Cependant, l'application directe de la définition n'est pas une méthode très efficace. Euclide<sup>2</sup> s'était intéressé en son temps à cette question et avait mis au point un algorithme.

si un des nombres est nul, l'autre est le PGCD

sinon il faut soustraire le plus petit du plus grand et laisser le plus petit inchangé.

Puis, recommencer ainsi avec la nouvelle paire jusqu'à ce qu'un des deux nombres soit nul. Dans ce cas, l'autre nombre est le PGCD.

Cet algorithme exploite les propriétés du PGCD

$$PGCD(a, b + a) = PGCD(a, b)$$

$$PGCD(a, b) = PGCD(b, a)$$

$$PGCD(a, 0) = a$$

---

<sup>2</sup> Eh oui, les sciences informatiques ce ne sont finalement pas si récentes. Le mot algorithme vient du nom du mathématicien perse du 9<sup>ième</sup> siècle Abu Abdullah Muhammad ibn Musa al-Khwarizmi.

On peut aller encore plus loin et rendre compte qu'en soustrayant  $a$  à  $b$  plusieurs fois, on calcule en fait le reste de la division de  $b$  par  $a$ . On peut donc utiliser directement le modulo.

Ces algorithmes « définition du PGCD », « soustraction du plus petit au plus grand », « modulo » peuvent être implémentés facilement avec un langage comme Blockly ou Scratch et permettre aux élèves d'avoir une réalisation concrète et de percevoir la différence d'efficacité pour des grands nombres.

La figure 2 montre un programme reprenant plusieurs algorithmes de calcul du PGCD.

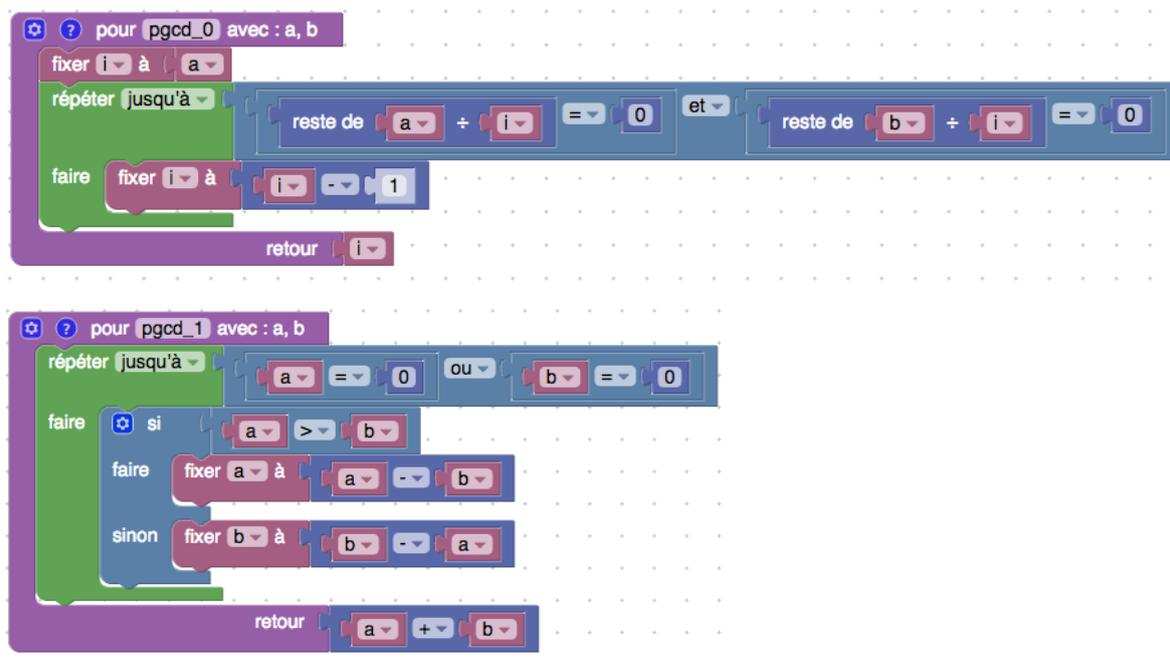


FIGURE 2 : EXEMPLE DE PROGRAMME ÉCRIT EN BLOCKLY IMPLÉMENTANT DEUX ALGORITHMES DE CALCUL DU PGCD, EN HAUT LA « DÉFINITION DU PGCD » ET EN BAS LA « SOUSTRATION DU PLUS PETIT AU PLUS GRAND » - ALGORITHME D'EUCLIDE

#### QUEL APPOINT POUR LA DISCIPLINE CONCERNÉE ?

Cette activité permet d'utiliser les propriétés d'une fonction comme le PGCD dans un but précis : être le plus efficace possible. Cela donne du sens à l'apprentissage de toutes ces propriétés mathématiques.

C'est également l'occasion de réinvestir des notions comme diviseur, reste d'une division qui sont moins utilisées à partir du moment où les élèves connaissent les nombres réels et effectuent les opérations avec une calculatrice.

Par ailleurs, manipuler des variables dans un programme est très proche de la manipulation d'inconnues dans des expressions littérales. Cela peut aider les élèves à franchir le pas vers l'abstraction qui débute au 1<sup>e</sup> degré en mathématiques par les expressions littérales et qui débouche en fin de secondaire sur les dérivées et intégrales.

#### QUEL APPOINT POUR LES COMPÉTENCES TRANSVERSALES ?

L'activité illustre également un élément central en algorithmique mais également dans la vie quotidienne : la complexité (l'efficacité) des algorithmes. Il y a souvent plusieurs solutions pour résoudre un problème mais toutes ces méthodes ne sont pas équivalentes en termes d'efficacité.

La population est parfois impressionnée par la performance des machines (smartphone, calculatrices, ordinateurs) et les trouvent pour cette raison très intelligentes. Cette activité peut permettre de réaliser qu'en fait, la machine ne fait qu'exécuter un algorithme (une recette) et qu'elle est juste très rapide pour l'exécuter. Il n'y a pas la moindre intelligence dans tout cela. La machine peut éviter à l'humain de répéter des tâches (avec risque d'erreur) et l'activité met en lumière l'importance de la compréhension de l'algorithme en lui-même. Ceci permet à l'apprenant de se rendre compte que l'intelligence est bien celle du mathématicien combinée à celle de l'informaticien et non pas la « magie » d'une machine.

#### QUEL APPOINT POUR LES SCIENCES INFORMATIQUES ?

Une partie importante de cette activité peut se faire en mode « déconnecté », c'est-à-dire sans ordinateur. L'accent est mis sur les algorithmes et non sur la programmation, sur le raisonnement et non sur la réalisation. Cela permet de montrer une autre facette des sciences informatiques. L'informatique, ce n'est pas que programmer !

Mais évidemment, la programmation n'est jamais très loin et les élèves peuvent aller au terme de l'activité jusqu'à la réalisation un programme qui calcule de manière plus ou moins efficace le PGCD de 2 nombres.

#### QUELS SONT LES PRÉREQUIS EN SCIENCES INFORMATIQUES ?

Aucun prérequis en sciences informatiques n'est nécessaire pour la partie algorithmique. Evidemment, des bases de programmation sont nécessaires pour la réalisation concrète du programme :

- les instructions ;
- les structures de contrôle (boucles et conditions) ;
- les variables.

## LES SCIENCES INFORMATIQUES POUR AIDER À LA « DÉCISION » : APPLICATION DANS UN COURS DE BIOLOGIE

### CONTEXTE

Nous vivons dans un monde où l'information est omniprésente. Mais trop d'information tue l'information. Nous n'arrivons plus à en extraire les éléments pertinents. Face à une série de quelques nombres, l'être est capable d'identifier les liens, les propriétés communes mais face à des milliards, on est noyé et on ne voit plus rien.

C'est un peu le problème de biologistes actuellement. Ils ont de plus en plus d'informations sur les réactions chimiques qui ont lieu dans nos cellules. Ils ont de plus en plus d'informations sur les animaux « préhistoriques ».

En biologie dès le premier degré, on présente ainsi aux élèves la « classification des animaux », qui en fait ne cesse d'évoluer. Anciennement par exemple, on scindait les animaux en « vertébrés » et « invertébrés ». Le groupe des « invertébrés » a été remplacé par plusieurs sous-groupes. Le groupe des poissons a disparu : il est remplacé par 2 groupes : les poissons à squelette cartilagineux et les poissons à squelette osseux. Les reptiles ne constituent plus un groupe de vertébrés. Les crocodiles sont plus proches des oiseaux que des autres reptiles. Beaucoup de reptiles fossiles sont plus proches des mammifères que des lézards.

Pourquoi cette évolution ? En fait, la nouvelle classification se fonde sur des arbres phylogénétiques qui permettent de visualiser les relations de parenté plus ou moins fortes entre les espèces. Ils sont le reflet des connaissances à un moment donné, et donc ils évoluent en fonction des caractéristiques de fossiles retrouvés.

Souvent, les séquences d'apprentissage sur cette matière inscrite au programme du 1<sup>e</sup> degré proposent aux élèves de classer les animaux en fonction de caractères spécifiques à certains êtres vivants (avoir des poils, des sabots, ...), ils en déduisent des regroupements avec des groupes emboîtés. On leur montre généralement qu'on peut représenter ces groupes emboîtés sous forme d'arbres de parenté.

Ensuite, on indique que les paléo-biologistes ont fait cela avec tous les êtres vivants actuels et retrouvés sous forme fossile et en ont déduit une classification. On montre alors aux élèves au moins une partie de l'arbre et ils s'exercent à l'utiliser.

Les sciences informatiques pourraient permettre d'explicitier un peu plus les caractéristiques de ce type d'algorithmes de classification.

### BRÈVE DESCRIPTION DE L'ACTIVITÉ

Comme proposé par CSUnplugged, l'activité débute en choisissant 4 mots avec une prononciation très proche comme « patte, pâte, plate, blatte ». On choisit 9 enfants. Le premier écrit un message composé des 4 mots. La longueur de ce message doit être de 6 ou 7 mots, tel que « patte, pâte, plate, blatte, plate, patte ».

Comme indiqué à la figure 3, cet élève 1 chuchote son message aux élèves 2 et 3 successivement. Ceux-ci notent le message qu'ils ont compris. A leur tour, ils chuchotent le message en fonction de l'arbre ci-dessous.

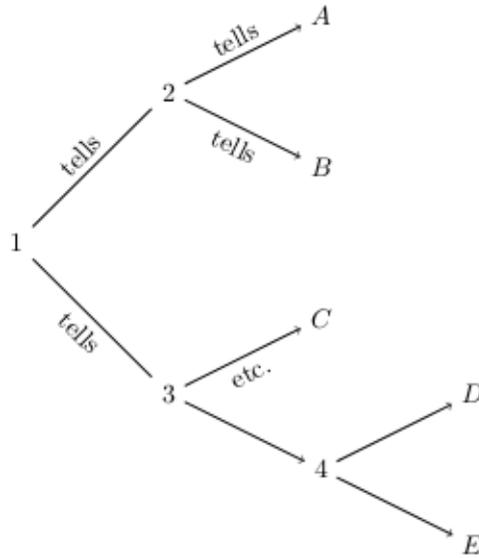


FIGURE 3 : ARBRE SELON LEQUEL LES ÉLÈVES 1, 2, 3, A, B, C, D ET E SE TRANSMETTENT UN MESSAGE.  
SOURCE : COMPUTER SCIENCES UNPLUGGED

L'enseignant récupère les messages écrits par les différents élèves 1, 2, 3, 4, A, B, C, D. Evidemment, il y a eu des modifications du message au fur et à mesure de sa transmission. Celles-ci se sont propagées sur la branche concernée de l'arbre.

Pour la transmission des caractères morphologiques des êtres vivants, de leur patrimoine génétique, il en est de même au cours de l'évolution.

Ensuite, le but du jeu est de pouvoir retrouver à quel élève correspond quel message. Les élèves peuvent y réfléchir en groupes et proposer leur solution. On voit alors si c'est bien conforme à ce qui s'est produit.

L'enseignant poursuit en expliquant l'algorithme décrit dans l'activité de CSUnplugged <http://csunplugged.org/wp-content/uploads/2014/12/PhylogeneticsUnplugged.pdf>. Celui-ci se base sur le calcul de la distance entre deux messages (la distance étant simplement définie comme le nombre de différences entre deux messages. Par exemple, la distance entre « patte, pâte, plate, blatte, plate, patte » et « patte, patte, plate, blatte, blatte, patte » est donc de 2. On crée alors un premier regroupement avec les 2 messages les plus proches. Ensuite, on recalcule les distances. La distance par rapport à un regroupement est la moyenne des distances par rapport à ces membres. Ainsi de suite, on peut créer les regroupements et construire un arbre de parenté entre les messages.

Il est alors plus facile de comprendre que l'on peut faire cela grâce à un ordinateur en utilisant de très nombreux critères (plumes, écailles mais aussi existence de certains os, leur forme, ...).

#### QUEL APPOINT POUR LA DISCIPLINE CONCERNÉE ?

L'impression au terme des activités classiques sur le sujet est souvent que l'on peut obtenir n'importe quelle classification en fonction des critères que l'on prend en premier. Pourquoi les paléo-biologistes mettent-ils plus de poids sur le fait d'avoir 4 membres que sur le fait d'avoir des plumes, des écailles ou des ailes ? Cela peut en effet paraître étrange de séparer les crocodiles des autres « reptiles » ou de rapprocher les poissons des mammifères en les séparant radicalement d'autres poissons cartilagineux. L'élève est naturellement plus enclin à utiliser des critères concernant l'apparence des animaux que des critères parfois internes comme la présence ou l'absence d'un os.

Cette activité met en pratique un algorithme qui explicite la démarche des paléo-biologistes. Expliciter l'algorithme rend les choses plus « acceptables » pour les élèves.

Au-delà de percevoir qu'il peut y avoir plusieurs classifications, cette activité montre que la classification actuelle (phylogénétique) est en lien avec le modèle de l'évolution des espèces (entre autres parce que les paléo-biologistes y incluent les caractéristiques d'espèces éteintes mais retrouvées comme fossiles). On met également en lumière que l'état de nos connaissances est amené à la faire évoluer.

Il est également intéressant en fin d'activité de rappeler qu'un bon algorithme ne doit pas être suivi aveuglément, les paléo-biologistes peuvent influencer la classification par les informations qu'ils fournissent à l'ordinateur mais aussi par les détails de l'algorithme qu'ils implémentent. C'est à eux d'avoir un regard critique sur les sorties fournies par cet algorithme.

#### QUEL APPOINT POUR LES COMPÉTENCES TRANSVERSALES ?

Au-delà de cet exemple, les classifications sont présentes dans de nombreuses matières (classification des forces en physique, classification des fonctions des mots en langue, ...) et pouvoir élaborer et utiliser un arbre de classification est une compétence transversale fondamentale.

#### QUEL APPOINT POUR LES SCIENCES INFORMATIQUES ?

Cette activité explicite les fondements de la classification actuelle des êtres vivants mais met aussi en lumière l'apport de l'informatique au traitement des informations recueillies par les paléo-biologistes. Cette activité se réalise uniquement en « déconnecté », sans ordinateur. On se focalise sur l'algorithme et son apport pour une discipline particulière. Cela montre l'importance de l'interdisciplinarité qui peut naître de la collaboration entre un informaticien et un spécialiste d'une autre discipline. De plus, cette collaboration ne se limite pas à l'utilisation par le spécialiste d'un outil informatique préexistant mais il y a une réelle interaction entre les deux disciplines.

#### QUELS SONT LES PRÉREQUIS EN SCIENCES INFORMATIQUES ?

Aucun prérequis n'est nécessaire en sciences informatiques. Un peu de calcul, la représentation d'information sous forme de tableaux et le tour est joué.

## CONTEXTE

Les technologies sont partout autour de nous. De plus en plus, l'informatique s'immisce dans tous les produits technologiques. Les bâtiments sont truffés de capteurs qui allument ou éteignent l'éclairage, la qualité de l'air est vérifiée pour réguler la ventilation, des capteurs de température peuvent être commandés à distance pour planifier notre retour de vacances et être accueillis dans une maison qui a eu le temps de se réchauffer. Les voitures sont truffées de capteurs également pour éviter les collisions, assurer la qualité du freinage, elles utilisent le GPS pour nous diriger et éviter les embouteillages. Demain nous ne devons même plus les conduire, elles le feront toutes seules. Ce ne sont que quelques exemples. Imaginer initier les jeunes à la technologie sans les ouvrir vers les applications de l'informatique serait une aberration au 21<sup>e</sup> siècle.

Manipuler des senseurs simples (tels que MakeyMakey) permet de diminuer les erreurs de mesure. L'apprenant doit concevoir, maîtriser le processus expérimental pour l'automatiser et le programmer.

## BRÈVE DESCRIPTION DE L'ACTIVITÉ

Voici une activité où la connaissance de base des principes de l'informatique permettrait de faire des activités en automatisant les mesures, en utilisant des données existantes ou en tirant profit d'objets connectés. Ici, il s'agit de mesurer le niveau d'eau dans une bouteille avec des senseurs à différents paliers à l'aide d'un MakeyMakey pour confectionner un pluviomètre

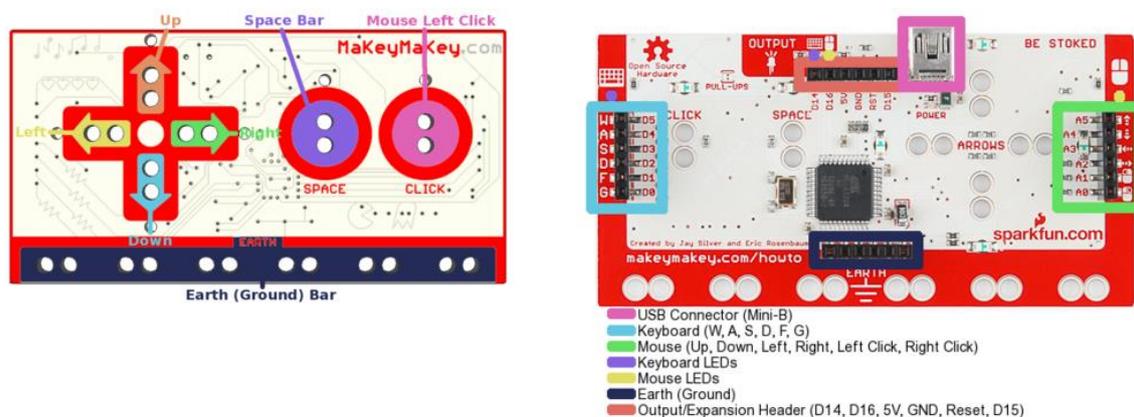


FIGURE 4 : RECTO-VERSO DE LA CARTE ÉLECTRONIQUE MAKEYMAKEY

MakeyMakey<sup>3</sup> est une carte électronique à laquelle on peut relier n'importe quel objet du quotidien pourvu qu'il soit conducteur. Lorsqu'un évènement survenant dans la réalité est détecté par un capteur connecté à la carte MakeyMakey (on touche l'un des objets), un signal est transmis à l'ordinateur, via USB. Ce signal correspond à la pression sur une touche (« key », en anglais) du clavier ou de la souris prédéterminée. Le lien entre l'objet touché et le signal envoyé dépend du câblage effectué. Si une pomme est reliée par un câble à l'entrée correspond à la flèche de gauche, le signal « flèche gauche » sera envoyé à l'ordinateur comme si on avait touché cette touche. On peut ainsi « simuler » le clavier en utilisant plusieurs fruits et par exemple déplacer un personnage sur l'écran. Tout se passe comme si, lorsqu'il se produit quelque chose, une main invisible pressait une touche<sup>4</sup>.

<sup>3</sup> [http://labenbib.fr/index.php?title=Makey\\_Makey](http://labenbib.fr/index.php?title=Makey_Makey) en français ou le site officiel <http://www.makeymakey.com/> en anglais.

<sup>4</sup> <https://vimeo.com/60307041>

Amusant mais on peut faire plus utile. Sur l'ordinateur, un programme écrit par l'élève peut prendre le relai pour dire ce qu'il faut faire avec ce signal. On peut ainsi permettre à l'ordinateur de facilement interagir avec son environnement.

Pour créer un pluviomètre par exemple, c'est très simple, il suffit d'un récipient pour récolter l'eau. On le gradue et on y place des objets (de simples morceaux de papier aluminium par exemple) tous les 5 mm. Ces objets sont connectés au MakeyMakey. Il pleut, l'eau monte. Quand le premier papier aluminium est en contact avec l'eau. L'évènement est transmis au MakeyMakey (il a y conduction). Celui-ci envoie par exemple un signal (comme si on venait de presser la touche W). Le programme prévoit que lorsque la touche W est enfoncée, un message d'affiche à l'écran pour indiquer qu'il a plu 5 mm. S'il continue à pleuvoir, le 2<sup>e</sup> papier aluminium sera en contact avec l'eau, il indiquera via le MakeyMakey à l'ordinateur qu'un autre évènement s'est produit (traduit pour l'ordinateur par la touche A a été enfoncée) et le programme indiquera à l'écran qu'il a plu 10 mm, ...

Cette version est très simple mais on peut la complexifier : afficher une gauge à l'écran qui se remplit au fur et à mesure qu'il pleut, afficher une représentation graphique, utiliser un dispositif de récupération de l'eau de pluie plus évolué qu'un simple réservoir<sup>5</sup>. On peut aussi simplement placer deux connecteurs sur une vitre (dans de la pâte à modeler par exemple) et laisser la pluie qui ruisselle sur la fenêtre servir de conducteur entre les deux. Eh oui, la pâte à modeler et l'eau de pluie sont deux conducteurs ! L'ordinateur reçoit le signal via MakeyMakey et mémorise ainsi la fréquence et la durée des averses.

#### QUEL APPORT POUR LA DISCIPLINE CONCERNÉE ?

Un tel projet permet d'utiliser très facilement des capteurs, pas besoin d'électronique compliquée. L'ordinateur traite le signal reçu à l'aide d'un programme écrit dans un langage simple (langage visuel comme Blockly ou Scratch). Les élèves peuvent ainsi réaliser un projet avec un/des objets interactifs combinant électronique et informatique.

Aucun danger dans ce type de projet, ils peuvent tester, expérimenter. Ils découvrent la conductivité, les circuits électriques de base. Et surtout ils peuvent interagir avec l'environnement en récoltant de l'information via la MakeyMakey, en la traitant et en programmant les actions qui en découlent (affichage à l'écran, stockage de l'information, allumer une lampe, ...).

Dans un monde connecté où interagir avec des objets devient la « norme », ce type de projet devrait susciter la motivation des élèves.

#### QUEL APPORT POUR LES COMPÉTENCES TRANSVERSALES ?

Les élèves peuvent expérimenter tant au niveau des connections dans les circuits électriques qu'ils développent que dans le développement d'un programme. Ils apprennent de leurs erreurs et peuvent ainsi progresser. Cette démarche est au cœur du processus d'apprentissage.

La démarche d'un informaticien mais qui ne lui est pas propre consiste à « imaginer, modéliser, faire des hypothèses », « réaliser un prototype », « observer le comportement du prototype » et en « tirer des conclusions pour améliorer le modèle » et arriver ainsi à un nouveau prototype, etc. En informatique, on parle d'un cycle entre « analyser », « concevoir », « développer », « tester ». C'est donc une manière d'illustrer la démarche expérimentale commune à de nombreuses autres disciplines.

---

<sup>5</sup> Exemple d'un pluviomètre électronique plus complexe utilisant MakeyMakey  
<https://www.youmagine.com/designs/3dkanjers-rain-gauge-regenmeter>

Ce type de dispositif est une porte ouverte vers la créativité. Le matériel est simple et a un coût raisonnable. Les applications sont immenses alors après un premier projet « guidé » pourquoi ne pas laisser les élèves inventer.

#### QUEL APPOINT POUR LES SCIENCES INFORMATIQUES ?

Un tel projet permet une application directe des sciences informatiques avec à la clé une réalisation concrète. Les élèves utilisent ainsi les bases de la programmation. Ils perçoivent comment on peut interfacier un ordinateur avec des périphériques extérieurs ici le MakeyMakey. Ils ne voient alors plus l'ordinateur comme un objet impossible à programmer et constituant une entité monolithique qui n'interagit pas avec d'autres éléments.

Le fait de comprendre que les contacts avec les connecteurs envoient des signaux électriques peut également aider les élèves à comprendre que l'ordinateur n'est en fait qu'un MakeyMakey en plus grand avec elle aussi une carte-mère, un processeur et des périphériques comme le clavier, l'écran, ...

#### QUELS SONT LES PRÉREQUIS EN SCIENCES INFORMATIQUES ?

Cette activité est typiquement le lieu pour apprendre la programmation. Elle devrait idéalement être précédées de petites activités plus courtes où les élèves apprennent à maîtriser d'environnement de programmation, découvrent les instructions séquentielles, les boucles, les conditions, les fonctions. Ils ont alors toutes les cartes en main pour pouvoir affronter un tel projet.

#### CONCLUSION

Ces 4 applications des sciences informatiques à d'autres disciplines dans le 1<sup>e</sup> degré du secondaire ne sont que des exemples. Bien d'autres applications pourraient être imaginées. Nous avons tenté de montrer qu'au-delà de l'intérêt pour l'acculturation informatique de la population à cette nouvelle science en plein essor au 21<sup>e</sup> siècle (voir document « informatique et enseignement obligatoire »), l'enseignement des sciences informatiques a de nombreux intérêts à la fois pour l'acquisition de compétences dans les autres disciplines mais également pour l'acquisition de compétences transversales.

Au cœur des sciences informatiques, il y a

- le **développement de réalisations concrètes** au travers de la programmation, ce qui est une source de motivation pour les élèves qui obtiennent un résultat tangible au terme de leurs efforts.
- la **démarche expérimentale itérative** : 1. « modéliser une problématique », « faire des hypothèses » et « imaginer une solution », 2. « réaliser un prototype », 3. « observer le comportement du prototype » et en « tirer des conclusions pour améliorer le modèle » et arriver ainsi à un nouveau prototype, etc. Au sein de cette démarche, l'algorithmique se trouve plutôt dans premier temps de ce cycle, la programmation au deuxième et les tests au troisième.
- la **rigueur** imposée par l'ordinateur. Faire fonctionner cette machine nécessite d'explicitier précisément les ordres qu'on lui donne. Avec les langages de haut niveau et les langages visuels actuellement disponibles, cela est désormais accessible pour les jeunes (pour certains langages dès la maternelle). La rigueur est évidemment requise dans de nombreuses autres disciplines mais être confronté à la machine rend cette nécessité plus impérieuse, plus visible.
- **l'interdisciplinarité** car on réalise toujours un système informatique pour répondre à un besoin et la plupart des champs d'applications des sciences informatiques se trouvent en dehors de cette science.

Nous pensons qu'une fois les compétences de bases acquises en sciences informatiques, les possibilités d'interdisciplinarité sont infinies et limitées uniquement par l'imagination des enseignants. C'est pour cela qu'il faudrait qu'un cours portant sur les sciences informatiques soit instauré dans le secondaire (voir document « informatique et enseignement obligatoire »). De cette façon, l'enseignant d'une autre discipline pourra compter sur cette base commune de connaissances et compétences pour mettre en place ses activités.

TABLEAU SYNTHÉTIQUE

Activité / discipline concernée / thématique	Apport pour la discipline	Apport pour les compétences transversales	Apport pour les sciences informatiques
<ul style="list-style-type: none"> <li>- modéliser, pour clarifier les règles</li> <li>- cours de langue</li> <li>- le pluriel des noms</li> </ul>	<ul style="list-style-type: none"> <li>- Construire les règles grammaticales et les tester</li> </ul>	<ul style="list-style-type: none"> <li>- « modéliser » et tenter de trouver des règles génériques</li> <li>- être rigoureux dans la rédaction de ces règles</li> </ul>	<ul style="list-style-type: none"> <li>- Utiliser les concepts de base de la programmation : instructions séquentielles, conditions, variables, listes</li> </ul>
<ul style="list-style-type: none"> <li>- mobiliser les savoirs</li> <li>- cours de mathématiques</li> <li>- calculer le PGCD</li> </ul>	<ul style="list-style-type: none"> <li>- utiliser les propriétés d'une fonction comme le PGCD dans un but précis</li> <li>- réinvestir des notions</li> <li>- appuyer l'apprentissage du calcul littéral par l'utilisation de variables en programmation</li> </ul>	<ul style="list-style-type: none"> <li>- Illustrer le fait qu'il y a souvent plusieurs solutions pour résoudre un problème mais toutes ces méthodes ne sont pas équivalentes en termes d'efficacité (ou d'un autre critère)</li> <li>- Montrer que l'intelligence est bien celle de l'être humain et non pas la 'magie' d'une machine qui elle est simplement rapide</li> </ul>	<ul style="list-style-type: none"> <li>- Mettre l'accent sur la notion d'algorithme</li> <li>- Comparer l'efficacité de plusieurs algorithmes</li> </ul>
<ul style="list-style-type: none"> <li>- aide à la décision</li> <li>- cours de sciences-biologie</li> <li>- la classification phylogénétique</li> </ul>	<ul style="list-style-type: none"> <li>- Dévoiler le type d'algorithmes utilisés pour créer la classification phylogénétique des êtres vivants</li> <li>- montrer que la classification actuelle est en lien avec la théorie de l'évolution des espèces, et que l'état de nos connaissances est amené à la faire évoluer</li> </ul>	<ul style="list-style-type: none"> <li>- classer, faire des regroupements, de catégories est une compétence présente dans de nombreuses matières</li> <li>- pouvoir élaborer et utiliser un arbre de classification est une compétence transversale fondamentale</li> </ul>	<ul style="list-style-type: none"> <li>- montrer l'importance de l'interdisciplinarité qui peut naître de la collaboration entre un informaticien et un spécialiste d'une autre discipline</li> <li>- mettre l'accent sur l'algorithmique par une activité « déconnectée »</li> </ul>
<ul style="list-style-type: none"> <li>- au cœur des nouvelles technologies</li> <li>- cours d'initiation à la technologie</li> <li>- la réalisation d'un pluviomètre électronique</li> </ul>	<ul style="list-style-type: none"> <li>- automatiser des mesures par l'utilisation de capteurs</li> <li>- susciter la motivation des élèves dans un monde connecté où interagir avec des objets devient la « norme »</li> </ul>	<ul style="list-style-type: none"> <li>- Mettre en pratique la démarche expérimentale</li> <li>- Offrir une porte ouverte vers la créativité</li> </ul>	<ul style="list-style-type: none"> <li>- Proposer une application directe des sciences informatiques avec à la clé une réalisation concrète</li> <li>- Mettre en avant la démarche expérimentale et itérative des informaticiens</li> </ul>