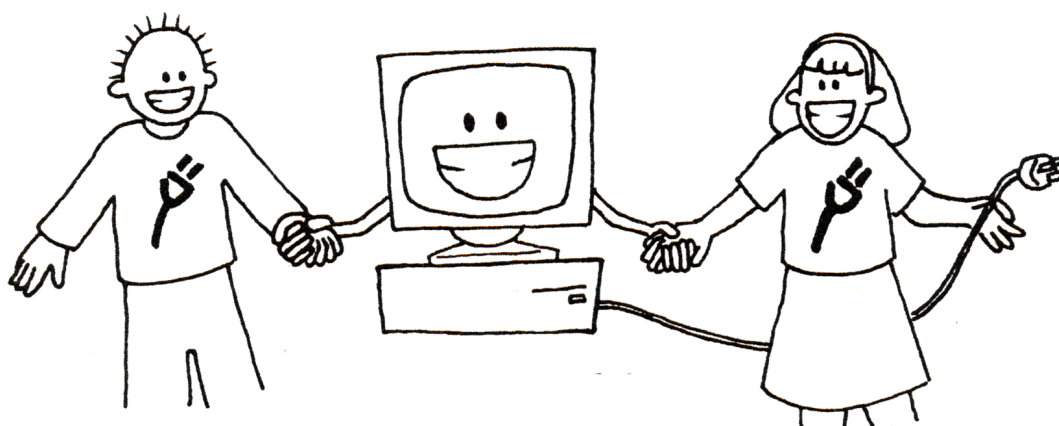
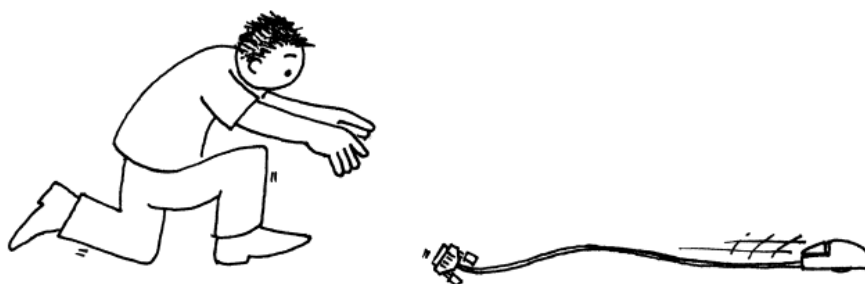


COMPUTER SCIENCE *Unplugged*

L'informatique sans ordinateur Parties IV, V et VI



**Programme d'enrichissement et
d'approfondissement
pour les élèves du primaire et du collège**



Créé par

Tim Bell, Ian H. Witten et Mike Fellows

**Adapté à l'utilisation en classe par Isaac Freeman à partir du travail
de Robyn Adams et Jane McKenzie**

**Illustrations de Malcolm Robinson, Gail Williams, Matt Powell et
Isaac Freeman**

Introduction

Cet ouvrage présente huit activités qui viennent compléter le livre de l'enseignant de *Computer Science Unplugged*. Elles figuraient déjà dans l'édition originale de 1999 mais sont ici adaptées à une utilisation en classe dans le même esprit que l'édition 2002 du livre de l'enseignant.

Cet ouvrage destiné à un usage personnel et éducatif est disponible en téléchargement libre grâce à la participation généreuse de Google Inc. Il est distribué sous une licence Creative Commons Attribution – Pas d'utilisation commerciale – Partage dans les mêmes conditions, ce qui signifie que vous êtes libres de reproduire, distribuer et communiquer ce manuel, ainsi que de le modifier ou de l'adapter, dans les conditions suivantes : vous devez indiquer le nom des auteurs de la version originale, ne pas l'utiliser à des fins commerciales, et si vous le modifiez ou l'adaptez, partager le résultat sous la même licence. Pour plus d'informations sur cette licence, recherchez sur le Web CC BY-NC-SA 3.0.

Nous encourageons son utilisation dans un contexte pédagogique, nous vous invitons à l'imprimer pour vos propres besoins et à distribuer les exercices à vos élèves. Toutes les remarques ou suggestions sont les bienvenues. Veuillez les adresser directement aux auteurs (cf. csunplugged.org).

Ce manuel est traduit de l'anglais en plusieurs langues. Merci de consulter le site Web pour toutes informations relatives à la disponibilité des traductions.

La version française a été coordonnée par l'équipe d'Interstices <https://interstices.info> et la mission de médiation scientifique d'Inria. Merci à Laurent Théry et Samuel Chalifour pour leur relecture attentive.

Sommaire

Les problèmes vraiment difficiles—L'intraitabilité.....	109
Le cartographe sans le sou—Coloration de graphes.....	112
Touristeville—Ensembles dominants.....	125
Les routes de glace—Arbres de Steiner.....	133
Partager des secrets et lutter contre la criminalité—La cryptographie	144
Partager des secrets—Protocoles de protection des données	148
Le pile ou face péruvien—Protocoles cryptographiques.....	152
Les cryptographes en herbe—Chiffrement à clé publique	162
Le visage humain de l'informatique—Interagir avec les ordinateurs	172
La chocolaterie—Conception d'interfaces utilisateurs	175
Dialoguer avec les ordinateurs—Test de Turing	187

Partie IV

**Les problèmes vraiment difficiles –
*L'intraitabilité***

L'intraitabilité

Certains problèmes sont-ils trop difficiles, même pour les ordinateurs ? La réponse est oui. Nous verrons dans l'Activité 20 que le simple fait de bavarder, de converser, est une chose impossible à faire pour les ordinateurs, non pas parce qu'ils sont incapables de parler, mais parce qu'ils ne peuvent pas comprendre ni dire des choses sensées. Les problèmes sur lesquels nous allons nous pencher ne concernent pas tant l'incapacité des ordinateurs à bavarder que notre propre incapacité à comprendre comment nous le faisons, et donc à expliquer à l'ordinateur comment le faire. Dans cette partie, toutefois, nous allons nous pencher sur des problèmes où il est facile de donner des ordres à l'ordinateur, en le programmant, mais où ce dernier ne peut pas accomplir la tâche en question parce que cela prendrait beaucoup trop de temps, peut-être des millions de siècles. Se procurer un ordinateur plus performant ne servirait pas à grand-chose : même cent fois plus rapide, il lui faudrait encore des millions d'années pour accomplir la tâche ; et même un million de fois plus rapide, il lui faudrait des centaines d'années. Un problème que même l'ordinateur le plus rapide imaginable mettrait plus d'une vie humaine à résoudre, voilà ce qu'on appelle un problème *difficile* !

Les activités de la Partie II sur les algorithmes vous ont montré comment améliorer l'efficacité des programmes informatiques. Nous allons aborder ici des problèmes pour lesquels *aucune* solution efficace n'est connue à ce jour, des problèmes qu'un ordinateur mettrait des millions de siècles à résoudre. Nous découvrirons ce qui constitue sans doute le plus grand mystère de la science informatique actuelle : *nul ne sait* s'il existe une manière plus efficace de résoudre ces problèmes ! Peut-être simplement que personne n'a encore trouvé la bonne méthode, ou peut-être que la bonne méthode n'existe pas... Il est impossible de le savoir. Et ce n'est pas tout. Il existe des milliers de problèmes qui, bien que totalement différents en apparence, sont équivalents au sens où si une méthode efficace était trouvée pour résoudre l'un d'eux, on pourrait l'adapter pour résoudre tous les autres. Ce sont ces problèmes que vous découvrirez dans les activités suivantes.

Pour les enseignants

Cette partie regroupe trois activités. Dans la première, il s'agit de colorier des cartes géographiques en déterminant le nombre de couleurs nécessaires pour que tous les pays limitrophes soient différents. Dans la deuxième, il faut savoir utiliser un plan de ville simple pour placer les camions des marchands de glaces à certains coins de rue, le but étant que personne n'ait trop de chemin à parcourir pour acheter une glace. La troisième est une activité de plein-air où l'on apprend à construire, à l'aide de ficelles et de piquets, des réseaux courts reliant un ensemble de points.

Ces activités permettent d'appréhender concrètement l'idée de complexité, c'est-à-dire le fait que des problèmes très simples à énoncer puissent s'avérer incroyablement difficiles à résoudre. Ces problèmes ne sont pas incompréhensibles : il s'agit de questions pratiques que l'on rencontre dans des activités aussi courantes que la création de cartes géographiques, d'itinéraires ou d'emplois du temps. La notion informatique qui les sous-tend, appelée « NP-complétude », est expliquée dans la section « Ce qu'il faut retenir » qui suit chaque activité. Bien que les activités elles-mêmes puissent être réalisées dans n'importe quel ordre, ces dernières sections doivent être lues dans leur ordre d'apparition. Une fois arrivé à la fin, vous aurez compris ce qui constitue la plus grande question ouverte de l'informatique contemporaine.

Le terme technique définissant les problèmes insolubles en pratique est « intraitabilité ». Il provient du latin *tractare*, « tirer », « traîner ». Ce concept définit donc des problèmes difficiles à appréhender parce que leur résolution demanderait trop de temps.

Sous ses dehors abscons, le concept d'« intraitabilité » revêt un grand intérêt pratique. Une découverte dans ce domaine aurait en effet des répercussions majeures dans de nombreux axes de recherche. La plupart des systèmes cryptographiques, par exemple, sont fondés sur l'« intraitabilité » de certains problèmes. Un criminel qui trouverait une solution efficace pourrait donc faire fortune en décodant des informations secrètes et en les revendant, ou plus simplement, en réalisant de fausses transactions bancaires. Nous aborderons ce sujet dans la Partie V, intitulée « Cryptographie ».

Activité 13

Le cartographe sans le sou – *Coloration de graphes*

Résumé

Souvent, les problèmes d'optimisation sont liés à des situations dans lesquelles certains événements ne peuvent être simultanés, ou dans lesquelles certains éléments d'un ensemble d'objets ne peuvent être adjacents. Quiconque a déjà essayé d'établir un emploi du temps scolaire ou de fixer une date de réunion, par exemple, sait combien il est difficile de répondre aux contraintes de toutes les personnes impliquées. Bon nombre de ces difficultés s'apparentent au problème du coloriage des cartes, où deux pays limitrophes ne doivent jamais être de la même couleur. Cette activité porte précisément sur ce problème.

Liens pédagogiques

- ✓ Mathématiques : nombres. Étudier les nombres dans d'autres bases. Représenter les nombres en base 2
- ✓ Mathématiques : algèbre. Continuer une séquence, décrire la règle qui la définit. Séquences et relations en puissance de 2

Compétences

- ✓ Résolution de problèmes
- ✓ Raisonnement logique
- ✓ Procédures algorithmiques et complexité
- ✓ Communication des idées

Âge

- ✓ 7 ans et plus

Matériel

- ✓ Un tableau

Pour chaque élève :

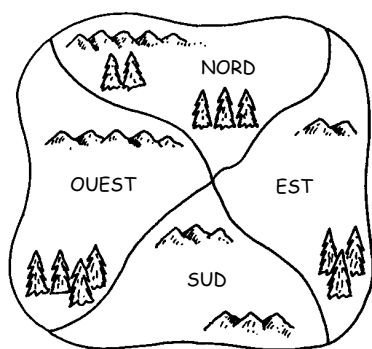
- ✓ Une copie d'une ou plusieurs fiches d'activité
- ✓ De petits éléments mobiles de couleur (jetons ou autres)
- ✓ Quatre crayons ou feutres de couleurs différentes

Coloration de graphes



Introduction

Dans cette activité, les élèves sont chargés d'aider un cartographe (quelqu'un qui fabrique des cartes géographiques) à colorier les pays d'une carte. La couleur attribuée à chaque pays importe peu, l'essentiel étant qu'elle soit différente de celle des pays limitrophes.



Prenons pour exemple cette carte qui représente quatre pays. Si on colorie le Nord en rouge, alors l'Ouest et l'Est ne peuvent pas être en rouge parce qu'il serait difficile de distinguer leur frontière avec le Nord. On pourrait donc colorier l'Ouest en vert, et il serait alors aussi possible de colorier l'Est en vert puisqu'il ne partage pas de frontière avec l'Ouest. (Si deux pays ne se touchent que sur un point, on considère qu'ils n'ont pas de frontière commune et on peut donc leur attribuer la même couleur). Enfin, le Sud peut être colorié en rouge. Deux couleurs suffisent donc à

colorier cette carte.

Dans notre histoire, le cartographe est pauvre et ne peut pas s'acheter beaucoup de crayons. L'idée est donc d'utiliser le moins de couleurs possible.

Déroulement de la séance

Décrivez le problème sur lequel les élèves vont travailler et montrez au tableau comment réaliser le coloriage.

Distribuez la première fiche d'activité. Deux couleurs suffisent pour la colorier correctement. La limitation à deux couleurs pourra sembler particulièrement contraignante mais la tâche est très simple comparée aux cartes qui nécessitent davantage de couleurs, car les possibilités pour chaque pays sont alors très restreintes.

Demandez aux enfants d'essayer de colorier la carte en n'utilisant que deux couleurs. Peut-être découvriront-ils la règle à suivre : une fois qu'un pays est colorié, on sait qu'« il faut que » tous les pays limitrophes soient de l'autre couleur. Le principe se répète jusqu'à ce que tous les pays soient coloriés. Il est préférable que les élèves parviennent à trouver la règle eux-mêmes, pour mieux intégrer le processus.

Au fur et à mesure que les élèves terminent une carte, vous pouvez leur donner la suivante pour qu'ils essaient de la remplir.

Les élèves découvriront peut-être aussi qu'il vaut mieux ne pas commencer directement par le coloriage, mais utiliser d'abord des éléments mobiles tels que des jetons afin de pouvoir changer d'avis plus facilement.

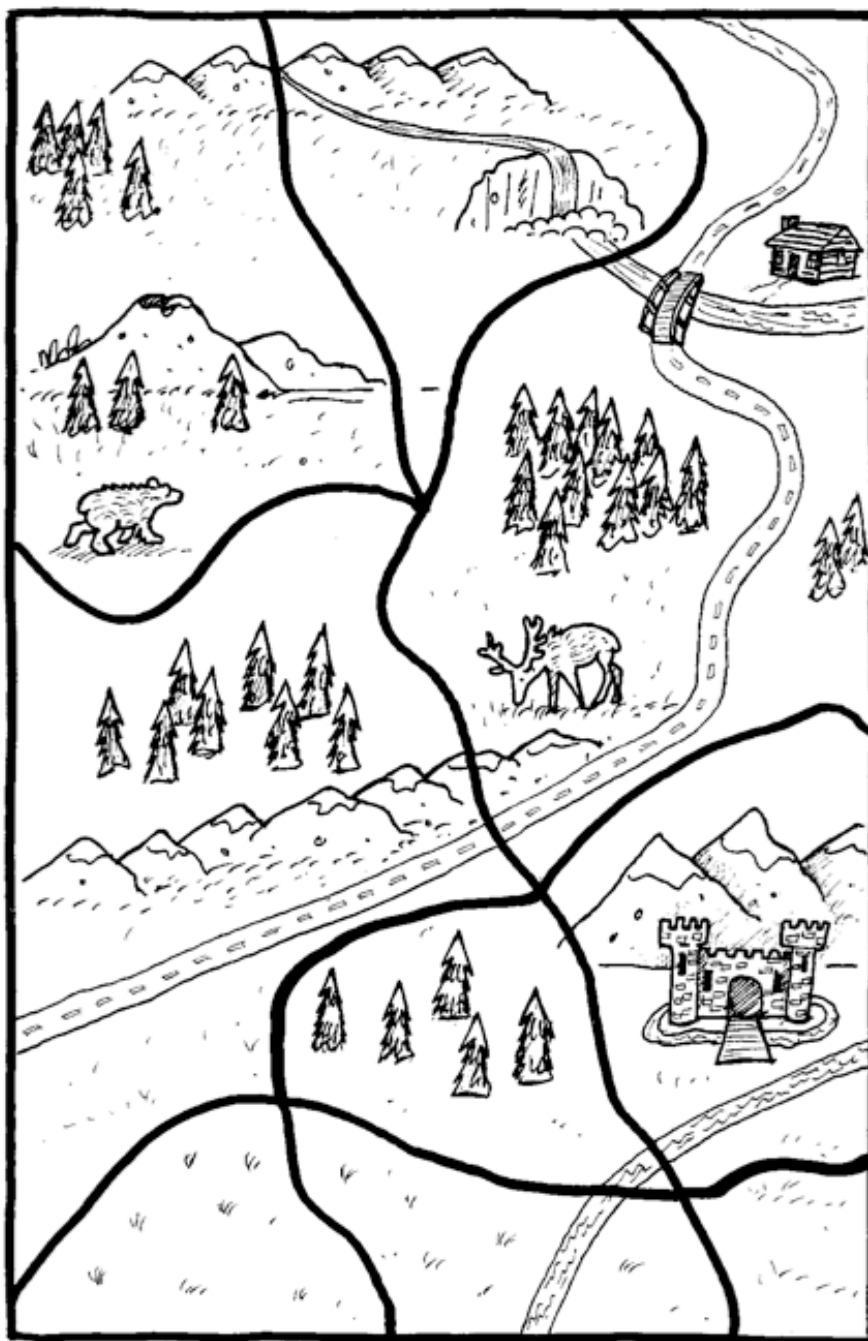
Demandez aux élèves plus âgés d'expliquer comment ils savent qu'ils ont trouvé le nombre minimum de couleurs. Pour la carte de la fiche d'activité 3, par exemple, il faut au moins trois couleurs parce qu'elle comprend un groupe de trois pays (les trois plus grands) qui partagent chacun une frontière avec les deux autres.

Demandez aux élèves qui ont terminé toutes les fiches d'activité avant les autres de dessiner une carte qui nécessite cinq couleurs. Il a été prouvé que quatre couleurs suffisaient à colorier n'importe quelle carte, cela devrait donc les tenir occupés un bon moment ! L'expérience nous a montré que les élèves imaginent rapidement des cartes qui, selon eux, nécessitent cinq couleurs, mais il est bien sûr toujours possible de trouver une solution à quatre couleurs.

Fiche d'activité : coloration de graphes 1

Colorie tous les pays de la carte en utilisant le moins de couleurs possible.

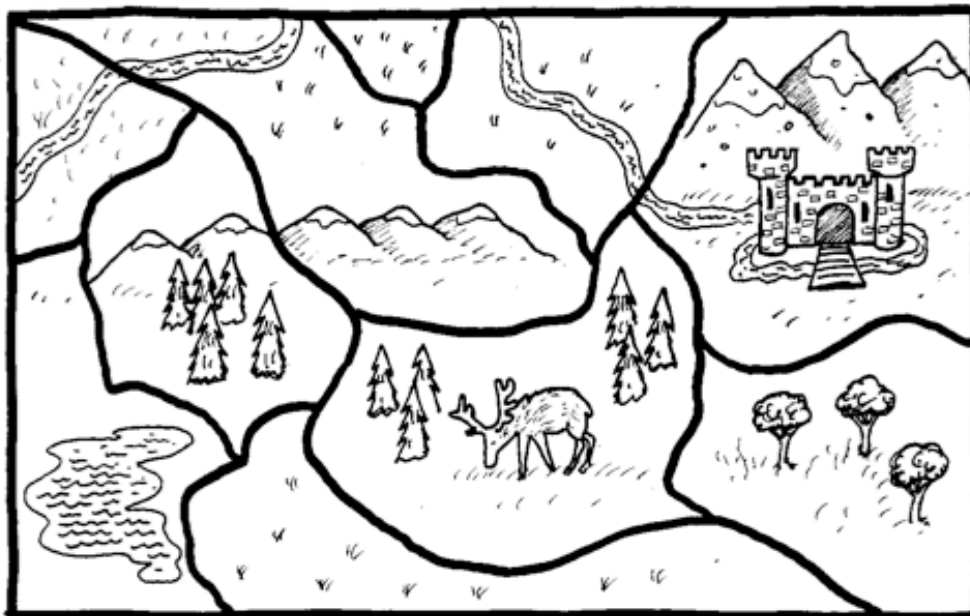
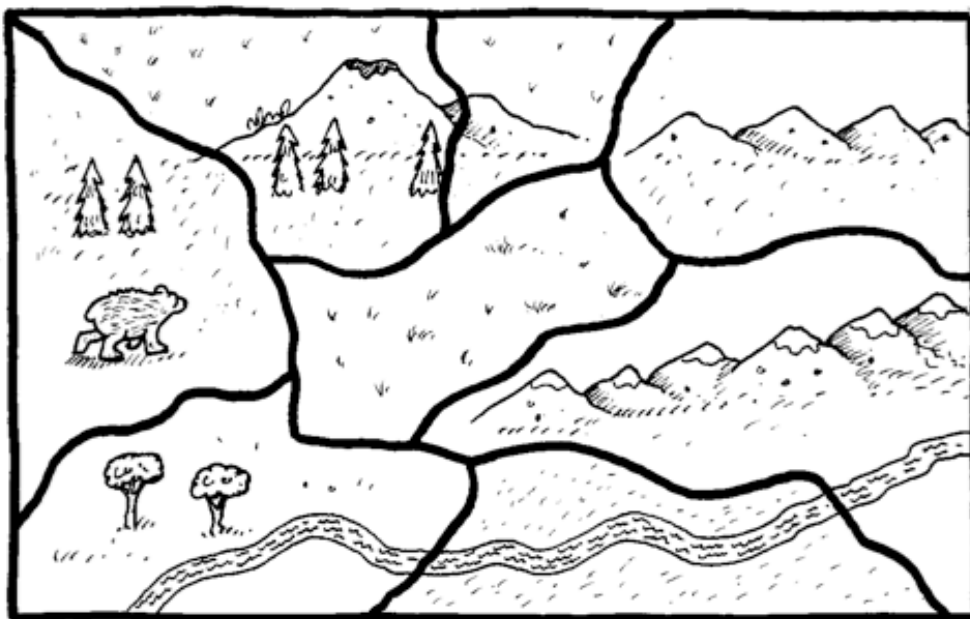
Attention : deux pays qui partagent une même frontière ne doivent pas être de la même couleur.



Fiche d'activité : coloration de graphes 2

Colorie tous les pays de la carte en utilisant le moins de couleurs possible.

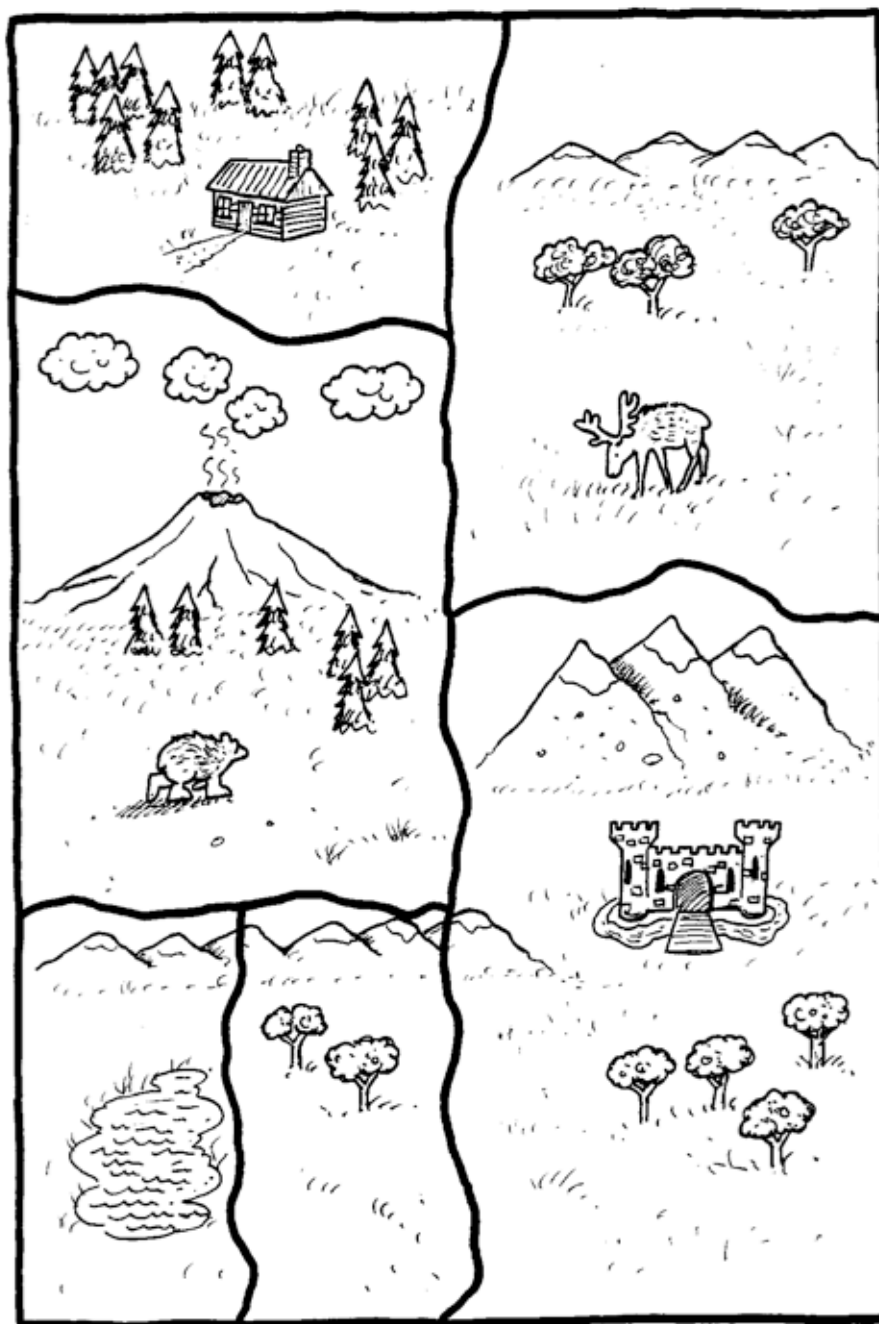
Attention : deux pays qui partagent une même frontière ne doivent pas être de la même couleur.



Fiche d'activité : coloration de graphes 3

Colorie tous les pays de la carte en utilisant le moins de couleurs possible.

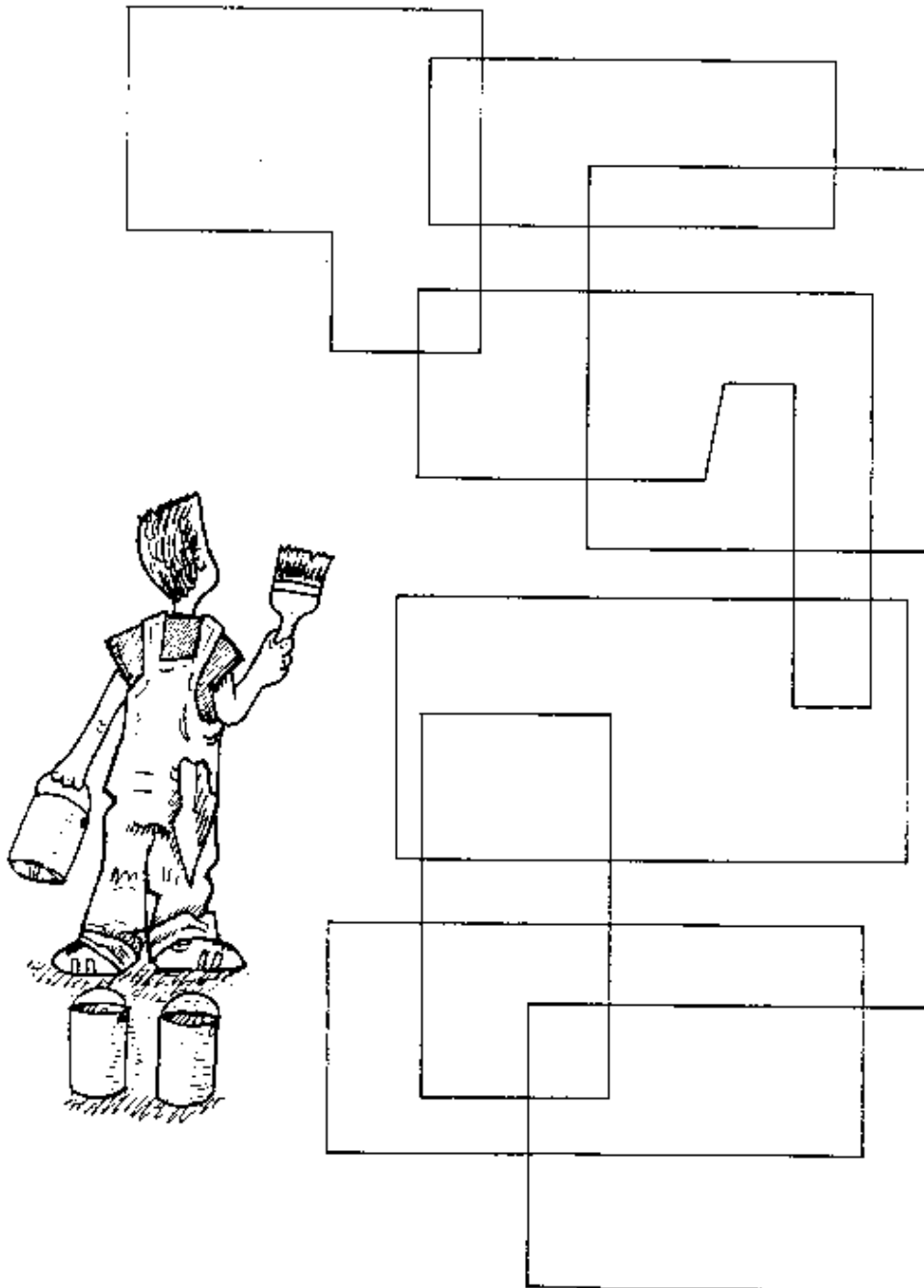
Attention : deux pays qui partagent une même frontière ne doivent pas être de la même couleur.



Fiche d'activité : coloration de graphes 4

Colorie tous les pays de la carte en utilisant le moins de couleurs possible.

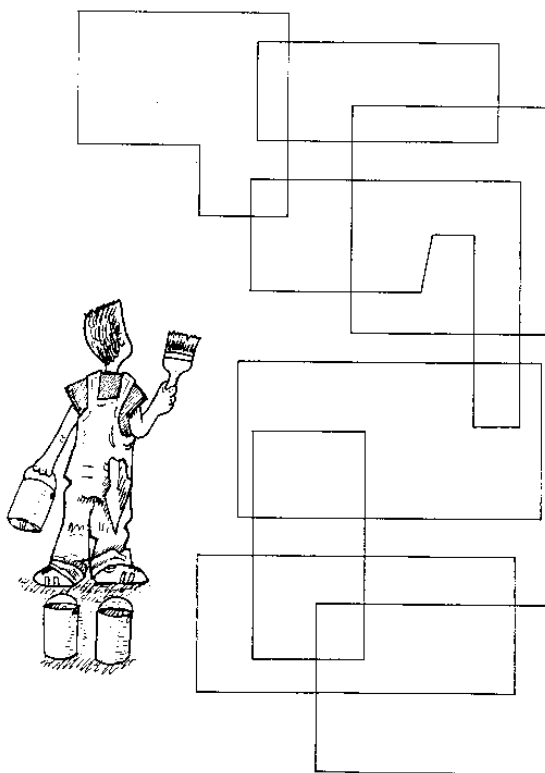
Attention : deux pays qui partagent une même frontière ne doivent pas être de la même couleur.



Variantes et activités complémentaires

Il existe un moyen simple de concevoir des cartes pour lesquelles deux couleurs suffisent, comme celle présentée ci-contre. Cette carte a été dessinée en superposant des courbes fermées (lignes dont le début rejoint la fin). Quels que soient leur nombre et leur forme, de telles courbes produisent toujours une carte à deux couleurs. Les élèves pourront essayer de créer ce type de carte.

Quatre couleurs suffisent toujours pour une carte dessinée sur une feuille de papier ou une sphère (un globe terrestre). On pourrait se demander (à l'instar des scientifiques qui sont payés pour le faire) combien de couleurs il faudra pour des cartes dessinées sur des surfaces moins ordinaires comme le tore (en forme de bouée). Dans ce cas, on peut avoir besoin de cinq couleurs, mais jamais davantage. Les élèves auront peut-être envie de tenter l'expérience.



Il y a de nombreuses variantes du problème du coloriage des cartes qui mènent dans des directions encore largement inconnues. Par exemple, si je colorie tout seul une carte sur une feuille de papier, je sais qu'un travail intelligent me permettra de trouver une solution à quatre couleurs. Mais supposons qu'au lieu de travailler seul je travaille avec un partenaire incompétent (voire hostile), et que nous choisissons à tour de rôle les couleurs des pays. Supposons que je travaille intelligemment alors que mon partenaire se contente de travailler en suivant la règle. Combien de crayons doivent se trouver sur la table pour que je parvienne à réparer les choix pas toujours éclairés (voire mal intentionnés) de mon partenaire ? Le nombre maximum n'est pas connu ! En 1992, il a été prouvé que 33 crayons suffisaient toujours, puis, en 2008, que 17 suffisaient. Mais on ne sait pas encore s'il en faut toujours autant (les experts pensent que moins de dix couleurs suffisent). Les élèves auront peut-être envie de mettre en scène cette situation à travers un jeu à deux, où l'un d'eux essaie de maximiser le nombre de couleurs dont son adversaire aura besoin.

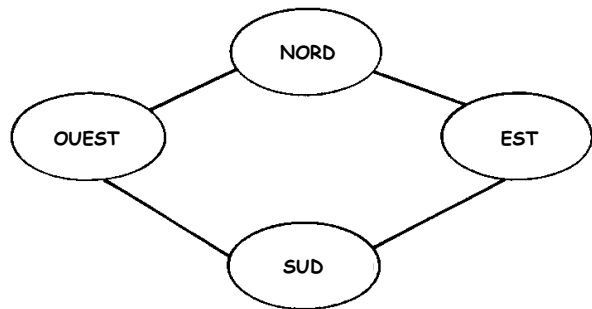
Dans une autre variante, appelée *coloriage de l'empire*, on commence avec deux cartes sur deux feuilles de papier. Le même nombre de pays figure sur chaque carte. Chaque pays de l'une des cartes (disons qu'elle représente la Terre) est couplé avec un pays de l'autre carte (disons qu'elle représente les colonies de la Lune). Aux impératifs habituels (différentes couleurs pour les pays limitrophes) s'ajoute l'obligation de colorier chaque pays de la Terre de la même couleur que sa colonie sur la Lune. Combien de couleurs faudra-t-il dans ce cas ? La réponse n'est pas connue à ce jour.

Ce qu'il faut retenir

Le problème du coloriage des cartes consiste pour l'essentiel à trouver le nombre minimum de couleurs (deux, trois ou quatre) nécessaires pour colorier une carte. L'hypothèse selon laquelle quatre couleurs suffisent à colorier n'importe quelle carte a été formulée en 1852 mais n'a pas été prouvée avant 1976. L'informatique regorge de problèmes irrésolus, et le fait que les scientifiques aient mis plus de 120 ans à prouver le théorème des quatre couleurs devrait encourager ceux qui travaillent sur d'autres problèmes dont la solution leur échappe depuis plusieurs décennies.

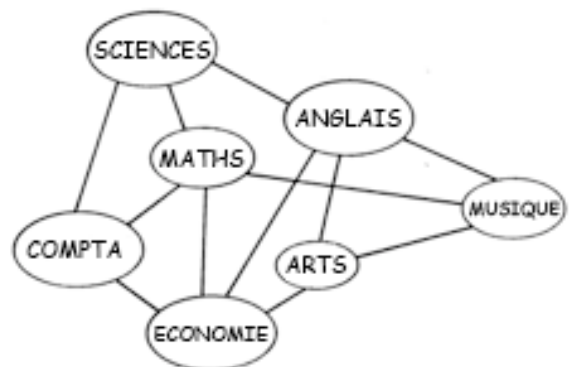
Le coloriage des cartes appartient à un ensemble de problèmes appelé « coloration de graphes ». En informatique, un graphe est une représentation abstraite de relations comme celle présentée ci-dessous.

Comme nous l'avons vu dans l'Activité 9 (La ville embourbée), le terme « graphe » ne doit pas être confondu avec le terme « graphique ». Les graphiques sont utilisés en statistiques, pour représenter des données numériques, par exemple, un graphique en barres créé à partir d'un tableau de données. Ces



graphiques n'ont aucun rapport avec les graphes. En informatique, les graphes sont dessinés à l'aide de cercles ou de gros points (le terme technique est « nœud ») qui représentent des objets et sont reliés par des lignes indiquant leurs relations. Le graphe ci-dessus représente la carte du début de l'activité. Les nœuds représentent les pays, et une ligne entre deux nœuds signifie qu'ils partagent une frontière commune. Ici, la règle de coloriage consiste donc à ne pas attribuer la même couleur à deux nœuds reliés. Il n'y a pas de limitation concernant le nombre de couleurs nécessaires à la coloration des graphes en général, contrairement aux cartes. Les lignes peuvent en effet représenter de nombreuses contraintes, alors que la nature bidimensionnelle des cartes réduit les possibilités d'arrangement. Le « problème de la coloration des graphes » consiste donc à trouver le nombre minimum de couleurs nécessaires pour un graphe donné.

Sur le graphe ci-contre, les nœuds correspondent aux matières à l'école. Une ligne entre deux nœuds signifie qu'au moins un élève a choisi les deux matières et qu'on ne doit donc pas les programmer aux mêmes horaires. En utilisant cette représentation, le problème qui consiste à créer un emploi du temps en utilisant le moins de plages horaires possible est le même que celui de la coloration, les différentes couleurs correspondant alors aux différents horaires. Les algorithmes de coloration de graphes sont d'un grand intérêt en informatique. On les applique à de nombreux problèmes qui se posent dans la vie réelle, mais ils ne sont probablement jamais utilisés pour colorier des cartes. En fait, l'histoire de notre cartographe sans le sou n'est qu'une fiction !



Il existe des milliers d'autres problèmes fondés sur les graphes. Certains sont traités dans d'autres parties du livre, notamment l'arbre couvrant minimum de l'Activité 9 et l'ensemble dominant de l'Activité 14. Les graphes constituent un moyen très général de représenter des données. On peut les utiliser pour représenter toutes sortes de situations, que ce soit une carte faite de routes et d'intersections, les liaisons entre les atomes d'une molécule, les trajectoires que les messages peuvent emprunter dans un réseau informatique, les connexions entre les différents composants d'un circuit imprimé, ou encore les liens entre les tâches requises pour mener à bien un projet d'envergure. Voilà pourquoi les problèmes liés aux graphes fascinent depuis longtemps les informaticiens.

Bon nombre de ces problèmes sont très difficiles, non pas d'un point de vue conceptuel mais parce qu'il faut beaucoup de temps pour les résoudre. Ainsi, trouver la meilleure solution pour un problème de coloration de graphe de taille modérée – déterminer l'emploi du temps pour une école de 30 enseignants et 800 élèves, par exemple – pourrait demander des années, voire des siècles, à l'ordinateur qui appliquerait le meilleur algorithme connu. Le problème n'aurait donc plus d'importance au moment où la solution serait trouvée – et encore faut-il que l'ordinateur ne tombe pas en panne avant de terminer ! C'est seulement en pratique que de tels problèmes se résolvent : on se contente alors de solutions qui, à défaut d'être optimales, sont très bonnes. Si nous devions garantir que la solution trouvée est la meilleure, le problème serait totalement impossible à traiter en pratique.

Le temps de calcul dont l'ordinateur a besoin pour résoudre un problème de coloration augmente de façon exponentielle en fonction de la taille du graphe. Prenons l'exemple du problème du coloriage des cartes, qu'on peut résoudre en testant tous les coloriages possibles. On sait qu'un maximum de quatre couleurs sera nécessaire. Il faudra donc évaluer toutes les combinaisons possibles pour l'attribution des quatre couleurs aux pays. S'il y a n pays, il y a 4^n combinaisons. Ce nombre augmente très rapidement : chaque pays ajouté multiplie par quatre le nombre de combinaisons possibles, et quadruple donc le temps de résolution. Ainsi, même si l'on mettait au point un ordinateur capable de résoudre le problème en une heure pour cinquante pays, le fait d'ajouter ne serait-ce qu'un pays lui demanderait quatre heures, et il suffirait d'ajouter dix pays pour que l'ordinateur mette plus d'un an à trouver la solution. Même si l'on continue d'inventer des ordinateurs toujours plus rapides, ce genre de problème n'est donc pas près de disparaître !

La coloration de graphe offre un bon exemple de problèmes dont le temps de résolution s'accroît de façon exponentielle. On trouve aisément la solution optimale dans des cas très simples comme ceux des fiches d'activité mais, dès que le nombre de pays dépasse la dizaine, résoudre soi-même le problème devient très compliqué et, avec plus d'une centaine de pays, même un ordinateur a besoin de plusieurs années pour tester toutes les façons possibles de colorier la carte afin de trouver la meilleure.

Dans la vie réelle, il existe toutefois de nombreux problèmes similaires que nous sommes bien obligés de résoudre. Les informaticiens emploient donc des méthodes qui donnent des résultats sinon parfaits, du moins satisfaisants. Ces techniques *heuristiques* sont souvent proches de l'optimum et permettent d'obtenir rapidement des résultats satisfaisants pour une application pratique. Les écoles peuvent ainsi tolérer l'utilisation d'une salle de classe supplémentaire par rapport au nombre requis si les emplois du temps étaient parfaits, tout comme le cartographe sans le sou peut sans doute s'acheter une couleur de plus que le strict nécessaire...

On n'a pas prouvé qu'il n'y avait aucune manière efficace de résoudre ce genre de problèmes sur des ordinateurs classiques, mais on n'a pas non plus prouvé qu'il y en avait une, et les informaticiens doutent qu'une méthode efficace soit jamais trouvée. Les deux activités suivantes nous feront découvrir plus avant ce type de problèmes.

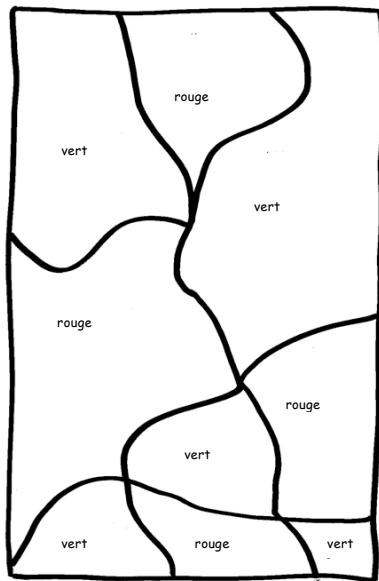
Suggestions de lecture (en anglais)

Dans son ouvrage intitulé *Algorithmics*, Harel présente le théorème des quatre couleurs et son histoire. Le problème du coloriage des cartes est traité plus en détails dans *This is MEGA-Mathematics !*, de Casey et Fellows. Enfin, le livre publié en 2004 par Kubale, *Graph Colorings*, présente l'historique du problème de la coloration de graphes. De nombreux sites Web traitent par ailleurs de ce sujet.

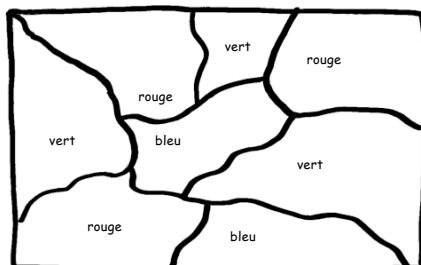
En français, nous vous recommandons les ouvrages suivants :

- *En cheminant avec Kakeya. Voyage au cœur des mathématiques*, Vincent Borrelli, Jean-Luc Rullière, ENS, 2014
- *Les Mathématiques*, Ian Stewart, Pour la Science, 1989
- *Le Théorème des quatre couleurs, coloriages des cartes graphiques*, Jean-Claude Fournier, Palais de la Découverte, 1978

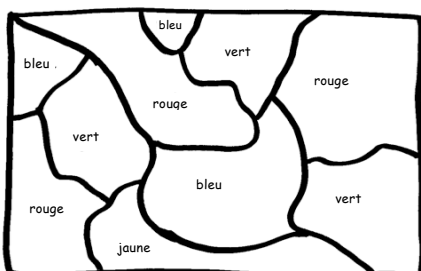
Astuces et solutions

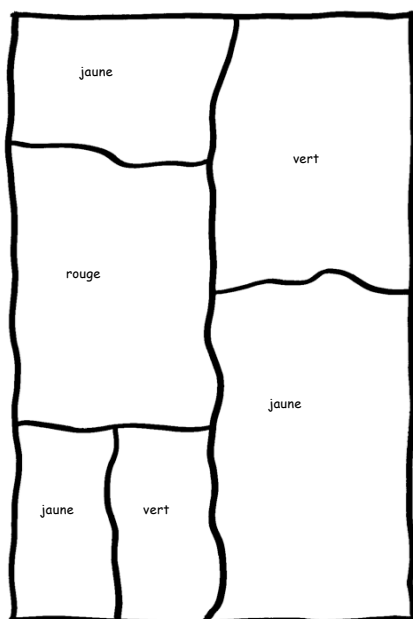


Voici la seule solution possible pour la fiche d'activité 1 (le choix des couleurs revient bien sûr aux élèves, mais deux couleurs suffisent).

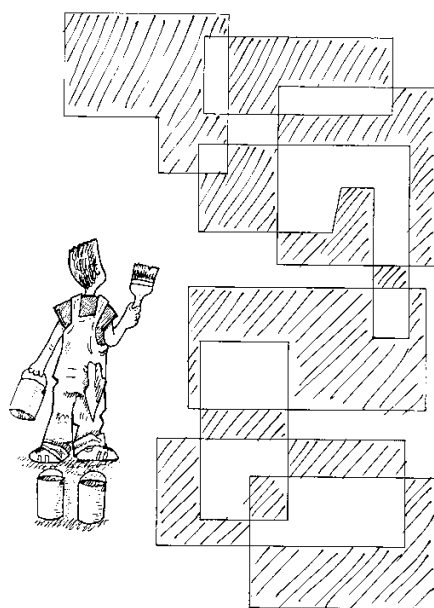


Sur la fiche d'activité 2, la carte du haut peut être correctement coloriée à l'aide de trois couleurs, celle du bas à l'aide de quatre couleurs. Voici deux solutions possibles.





La carte de la fiche d'activité 3 est une version plus simple de carte à trois couleurs, dont voici une solution possible.



Solution pour la fiche d'activité 4, deux couleurs suffisent (blanc et hachuré).

Activité 14

Touristeville – *Ensembles dominants*

Résumé

Beaucoup de situations de la vie réelle peuvent être représentées sous forme de réseaux ou de « graphes » semblables à ceux de l'Activité 13. Les réseaux offrent de nombreuses occasions de développer des algorithmes utiles en pratique. Dans cette activité, nous allons marquer certains croisements, ou « nœuds », de sorte que tous les autres ne soient pas situés à plus d'un pas des nœuds marqués. L'objectif est d'avoir un minimum de nœuds marqués. Le problème, vous le verrez, s'avère étonnamment difficile.

Liens pédagogiques

- ✓ Mathématiques : position et orientation
- ✓ Mathématiques : raisonnement logique

Compétences

- ✓ Plans
- ✓ Relations
- ✓ Résolution d'énigmes
- ✓ Recherche d'une solution par itération

Âge

- ✓ 7 ans et plus

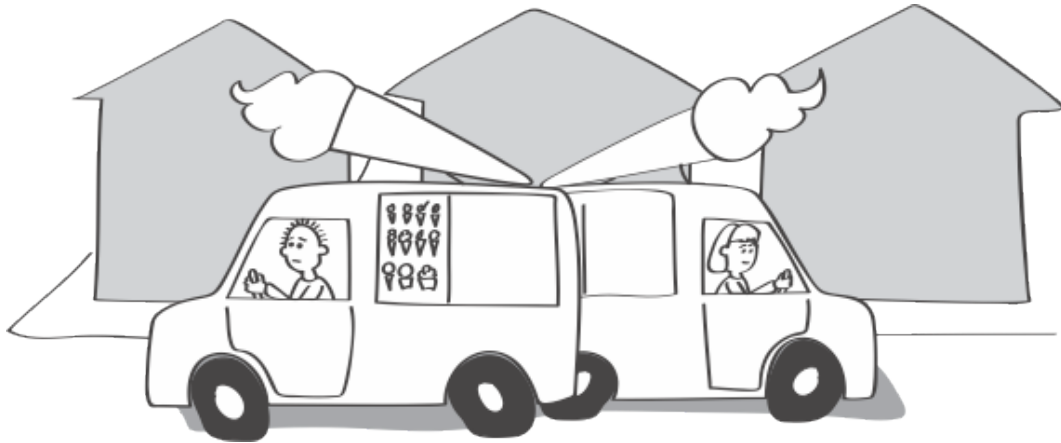
Matériel

Pour chaque groupe d'élèves :

- ✓ Une copie de la fiche d'activité *Les camions à glaces*
- ✓ Plusieurs jetons de deux couleurs différentes

Pour l'enseignant :

- ✓ Un transparent de la fiche d'activité *Les camions à glaces* à projeter au mur, ou un tableau sur lequel reproduire la carte.



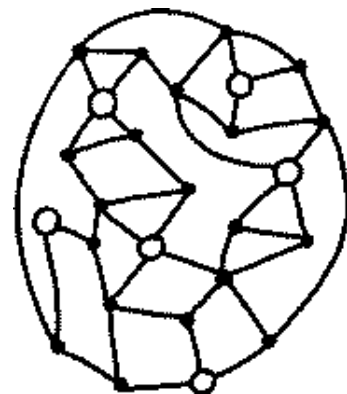
Les ensembles dominants

Introduction

La fiche d'activité *Les camions à glaces* présente une carte de Touristeville. Les lignes représentent les rues, les points représentent les carrefours. La ville se trouve dans un pays très chaud. Pendant la saison estivale, les marchands de glaces garent leurs camions aux coins de rue pour vendre leurs glaces aux touristes. L'objectif est de placer les camions de sorte que tout le monde puisse les atteindre en marchant jusqu'au bout de sa rue, puis au maximum un pâté de maison plus loin. (Pour simplifier les choses, on peut imaginer que les gens habitent aux intersections. Dans ce cas, ils doivent pouvoir atteindre le camion en marchant le long d'un pâté de maison seulement.) La question est la suivante : Combien de camions faut-il, et à quelles intersections doit-on les placer ?

Déroulement de la séance

1. Divisez la classe en petits groupes auxquels vous donnez une carte de Touristeville et quelques jetons, puis expliquez l'histoire.
2. Montrez aux élèves comment un jeton placé sur une intersection représente un camion à glaces, puis placez des jetons d'une autre couleur aux intersections se trouvant une rue plus loin. Les gens qui logent à ces intersections (ou dans les rues qui y mènent) seront desservis par ces camions à glaces.
3. Demandez aux élèves de tester différents emplacements pour les camions. Une fois qu'ils ont trouvé le moyen de satisfaire tout le monde, rappelez-leur que les camions coûtent cher et que l'objectif est d'en utiliser le moins possible. S'il y en avait suffisamment pour en placer un à chaque coin de rue, les conditions seraient bien évidemment remplies. Tout le problème est de trouver le nombre minimum de camions nécessaires.
4. Il faut au minimum six camions pour couvrir Touristeville. Une solution est présentée ici, mais elle est très difficile à trouver. Au bout d'un certain temps, dites à la classe que six camions suffisent et mettez les groupes au défi de trouver comment les placer. Le problème reste difficile : bon nombre de groupes finiront par abandonner. Même une solution à huit ou neuf camions n'est pas facile à trouver.

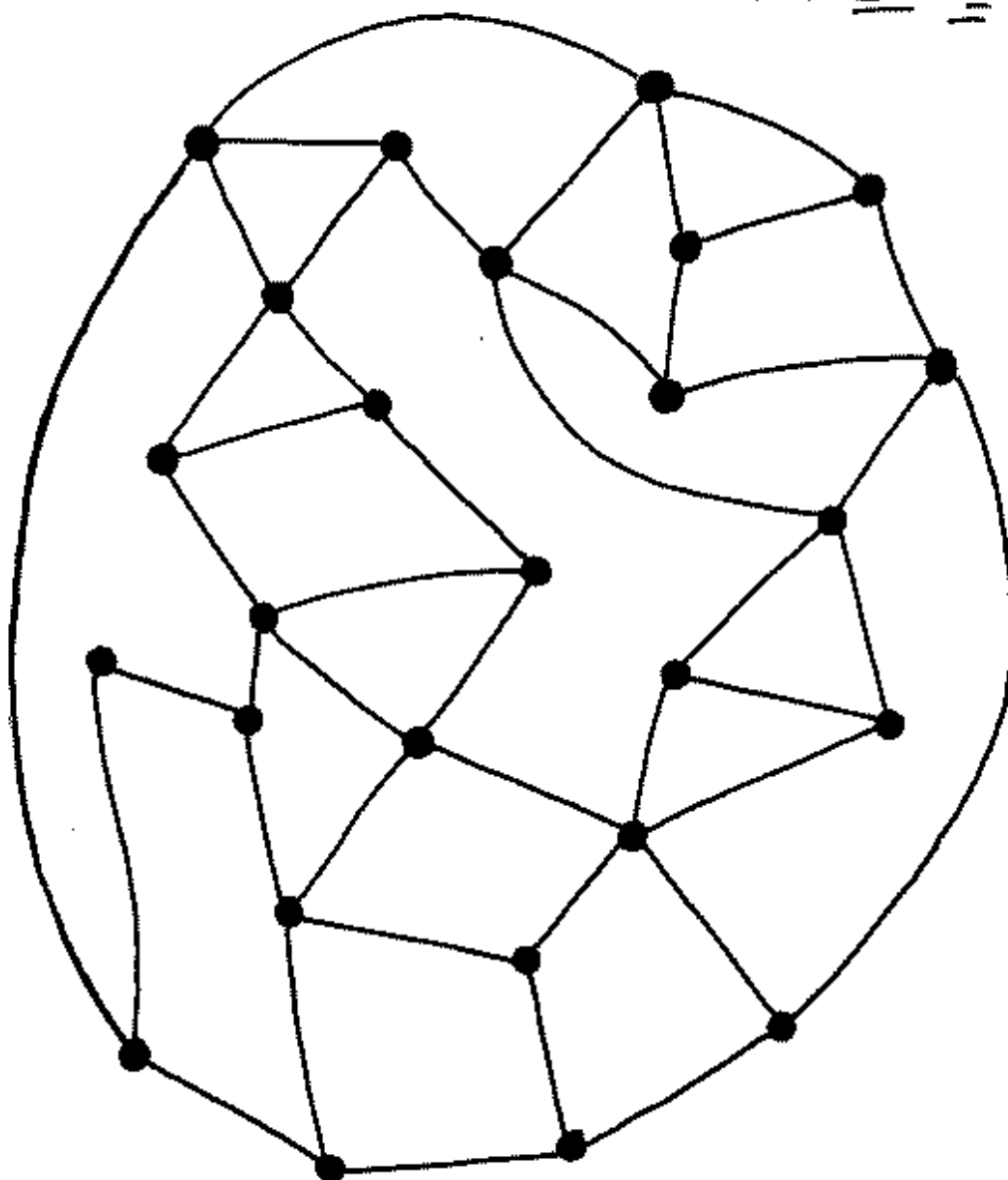


La carte de Touristeville a été créée à partir des six fragments de carte figurant en bas de la fiche d'activité *Solution aux camions à glaces*. Pour chacun d'eux, un seul camion suffit manifestement. On les a ensuite regroupés en ajoutant des rues afin de dissimuler la solution, l'essentiel étant alors de ne pas placer de lien entre les intersections « solutions » (les points blancs), mais seulement entre les autres intersections (les points noirs). Montrez cette technique à la classe, sur le tableau ou à l'aide d'un rétroprojecteur.

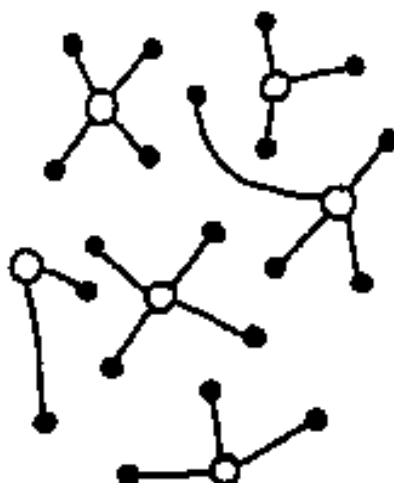
5. Demandez aux élèves de créer leur propre carte à l'aide de cette technique. Peut-être auront-ils envie de la tester sur leurs amis ou leurs parents. Ils découvriront ainsi qu'ils peuvent concevoir des énigmes que les autres sont incapables de résoudre ! Il s'agit là de ce qu'on appelle « une fonction à sens unique », cela permet de créer un problème qui sera très difficile à résoudre pour tout le monde à l'exception de son inventeur. Les fonctions à sens unique jouent un rôle essentiel en cryptographie (voir Activités 17 et 18).

Fiche d'activité : les camions à glaces

Place les camions à glaces à certaines intersections de sorte que toutes les autres intersections soient directement reliées à un carrefour équipé d'un camion.



Fiche d'activité : solution aux camions à glaces



Montrez cette fiche à la classe pour expliquer aux élèves comment l'énigme a été conçue.

Variantes et activités complémentaires

Ce type de problèmes urbanistiques est courant : la même question se pose pour placer les boîtes aux lettres, les puits, les casernes de pompiers, etc. Seulement, dans la réalité, la carte n'est pas conçue à partir d'une astuce facile à démêler. Si vous deviez vraiment résoudre un tel problème, comment feriez-vous ?

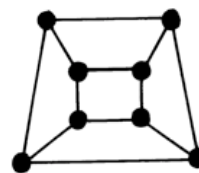
Pour trouver la meilleure solution, un moyen très simple consiste à envisager toutes les façons possibles de placer les camions à glaces. Touristeville comprenant 26 coins de rue, il y a 26 possibilités de placement pour un camion ; il est facile de les tester, et il est clair qu'aucune ne remplit les conditions voulues. Avec deux camions, il y a 26 emplacements possibles pour le premier, puis 25 pour le second (on ne va évidemment pas mettre deux camions sur le même carrefour) : $26 \times 25 = 650$ possibilités à tester. La tâche n'est toujours pas compliquée, mais devient extrêmement fastidieuse. En fait, il suffit de tester la moitié des possibilités (325), car peu importe s'il s'agit de l'un ou l'autre des camions : si vous avez testé l'emplacement A pour le camion 1 et l'emplacement B pour le camion 2, alors il est inutile de tester le camion 1 en B et le camion 2 en A. On pourrait poursuivre nos recherches avec trois camions (2600 possibilités), quatre camions (14 950 possibilités), et ainsi de suite. À l'évidence, 26 camions suffisent puisqu'il n'y a que 26 intersections et que deux camions au même endroit seraient inutiles. Une autre façon de déterminer le nombre de possibilités consiste à calculer le total des configurations possibles pour 26 intersections quel que soit le nombre de camions. Puisqu'il y a deux possibilités pour chaque intersection – un camion ou pas – le nombre de configurations possibles est égal à 2^{26} , soit 67 108 854.

Cette manière de résoudre le problème est appelée « algorithme exhaustif », elle demande énormément de temps. On pense généralement que la puissance des ordinateurs leur permet de résoudre rapidement tout problème, quelle que soit la quantité de travail nécessaire, mais c'est totalement faux. Le temps de résolution pour un algorithme exhaustif dépend du temps qu'il faut pour tester chaque configuration. Ici, il faudra tester chaque intersection pour vérifier où se trouve le camion le plus proche. Supposons qu'une configuration entière puisse être évaluée en une seconde. Combien de temps faudra-t-il pour tester les 2^{26} possibilités de Touristeville ? (Réponse : 2^{26} équivaut à environ 67 millions, il y a 86 400 secondes dans une journée, donc 2^{26} représente à peu près 777 jours, soit environ deux ans.) Supposons maintenant qu'au lieu d'une seconde, il faille un millième de seconde pour évaluer une configuration donnée. En deux ans, l'ordinateur pourrait alors tester une ville comprenant 36 intersections puisque 2^{36} représente environ 1000 fois 2^{26} . Enfin, même si l'ordinateur était un million de fois plus rapide (donc capable de tester un million de configurations par seconde), il pourrait, toujours en deux ans, tester une ville comprenant 46 intersections, ce qui n'est vraiment pas grand-chose ! (À titre de comparaison, combien d'intersections y a-t-il dans votre ville ?)

La recherche exhaustive est donc très lente. Existe-t-il d'autres manières de résoudre le problème ? Nous pourrions tenter l'approche « gloutonne » qui a fait merveille dans la ville embourbée (Activité 9). Il faudra donc se demander comment l'appliquer au problème des camions à glaces. Pour ce faire, on doit placer le premier camion au carrefour qui relie le plus grand nombre de rues, puis le deuxième camion au deuxième plus gros carrefour, et ainsi de suite jusqu'à compléter la carte. Mais cela n'offre pas forcément la solution la plus économe en camions. Ainsi, le plus gros carrefour de Touristeville, qui relie cinq rues, n'est pas un endroit stratégique pour un camion à glaces (vérifiez cette information avec la classe).

Considérons maintenant un problème plus simple. Au lieu de devoir trouver la configuration la plus économe en camions, supposons que vous soyez chargé de

déterminer si une configuration donnée est bien une solution minimale. Dans certains cas, la tâche sera facile. Le schéma ci-contre, par exemple, représente une carte dont la solution est très simple. Si l'on considère que les rues forment les arêtes du cube, il est évident que deux camions placés aux sommets diagonalement opposés suffiront. De plus, on admettra aisément qu'il est impossible de résoudre le problème avec moins de deux camions. En revanche, il est nettement moins facile (mais pas impossible) de se convaincre que Touristeville ne pourra pas être couverte par moins de six camions. D'une manière générale, il est extrêmement compliqué de démontrer qu'une certaine configuration de camions à glaces est bien minimale.



Ce qu'il faut retenir

L'un des aspects intéressants du problème des camions à glaces tient au fait que *nul ne sait* s'il existe un algorithme capable de trouver le nombre minimum d'emplacements de façon sensiblement plus rapide que la méthode exhaustive ! Le temps requis par cette dernière croît de façon exponentielle en fonction du nombre d'intersections. C'est ce qu'on appelle un algorithme *en temps exponentiel*. Avec un algorithme *en temps polynomial*, le temps de calcul augmente avec le carré, le cube, la puissance 17 ou n'importe quelle autre puissance du nombre d'intersections. Pour des cartes assez grandes, un algorithme polynomial – même de puissance 17, par exemple – sera toujours plus rapide qu'un algorithme exponentiel, car une fonction à croissance explosive (exponentielle) dépasse une fonction à croissance polynomiale à partir du moment où son argument est suffisamment grand (dès que n est supérieur à 117, par exemple, n^{17} est inférieur à 2^n). Existe-t-il un algorithme polynomial permettant de trouver le nombre minimum d'emplacements ? Nul ne saurait le dire, malgré tous les efforts fournis pour tenter de le trouver. Et il en va de même pour la tâche apparemment plus facile qui consiste à vérifier qu'une configuration est bien minimale : le temps de résolution de l'algorithme exhaustif qui teste toutes les possibilités pour des ensembles de points moins importants s'accroît de façon exponentielle en fonction du nombre d'intersections. Pour l'instant, personne n'a trouvé un algorithme polynomial capable de résoudre ce problème ni prouvé sa non-existence.

Cela vous rappelle le problème du coloriage des cartes (Activité 13) ? C'est normal. La question des camions à glaces, dont l'appellation technique est « problème de l'ensemble dominant minimum », fait partie des nombreux problèmes (il y en a des milliers) pour lesquels on ne sait pas s'il existe un algorithme en temps polynomial. On les rencontre dans des domaines aussi divers que la logique, les problèmes d'agencements ressemblant aux puzzles, le coloriage de cartes, le choix d'un trajet optimal sur un plan, ou encore la création d'emplois du temps. Aussi étonnant que cela puisse paraître, on sait que tous ces problèmes sont équivalents. Cela signifie que si un algorithme en temps polynomial était découvert pour l'un d'eux, il pourrait être adapté à tous les autres. En un sens, ils sont donc de la même étoffe.

Ces problèmes sont dits *NP-complets*, « NP » étant l'abréviation de « Non-deterministic Polynomial ». Cela signifie qu'ils pourraient être résolus dans un délai raisonnable si un ordinateur était capable de tester d'un coup un nombre arbitrairement élevé de solutions (voilà pour la partie « non-déterministe »). Si vous trouvez que l'hypothèse semble irréaliste, vous n'avez pas tort. Il est en effet impossible de fabriquer un tel ordinateur puisque cela implique qu'il soit « arbitrairement » puissant ! Quoi qu'il en soit, le concept sur lequel repose une telle machine est important dans son principe, car il s'avère que les problèmes NP-complets ne peuvent être résolus dans un délai raisonnable sans l'aide d'un ordinateur non-déterministe.

Par ailleurs, ces problèmes sont dits *complets* parce que, même s'ils semblent très différents (quel rapport entre le coloriage des cartes et le placement des camions à glaces ?), il s'avère que si on trouvait un moyen efficace pour résoudre l'un d'eux, la méthode pourrait être adaptée de façon à résoudre *n'importe quel* problème de cette classe. C'est en ce sens qu'ils sont de la même étoffe.

Il existe des milliers de problèmes NP-complets et cela fait plusieurs décennies que les scientifiques cherchent en vain un moyen efficace de les résoudre. S'ils parvenaient à trouver une solution efficace pour l'un d'eux, alors ils pourraient résoudre tous les autres. Voilà pourquoi on doute fortement qu'une solution efficace existe. Cependant, démontrer que la résolution de ces problèmes s'effectue nécessairement en temps exponentiel reste à ce jour la plus grande question ouverte de la science informatique – voire de toutes les mathématiques.

Suggestions de lecture (en anglais)

L'ouvrage de Harel, *Algorithmics*, présente plusieurs problèmes NP-complets et évoque la question de l'existence d'algorithmes en temps polynomial. *Turing Omnibus*, de Dewdney, s'intéresse aussi à la NP-complétude. L'ouvrage de référence sur le sujet est *Computers and Intractability*, de Garey et Johnson, qui présente plusieurs centaines de problèmes NP-complets ainsi que des techniques pour démontrer la NP-complétude. Il est assez ardu et s'adresse donc aux spécialistes de science informatique.

Activité 15

Les routes de glace – *Arbres de Steiner*

Résumé

Une petite variation dans la spécification d'un problème, même insignifiante en apparence, a parfois un impact énorme sur la difficulté de résolution. Comme pour le problème de la ville embourbée (Activité 9), il s'agit ici de trouver les plus courts chemins possibles à travers des réseaux. La seule différence tient au fait qu'on a le droit d'introduire de nouveaux points dans un réseau si cela permet de réduire la longueur d'un chemin. Il en résulte un problème nettement plus complexe qui, d'un point de vue algorithmique, s'apparente plutôt au problème du cartographe (Activité 13) ou de Touristeville (Activité 14).

Liens pédagogiques

- ✓ Mathématiques : position et orientation
- ✓ Mathématiques : raisonnement logique

Compétences

- ✓ Visualisation spatiale
- ✓ Raisonnement géométrique
- ✓ Procédures algorithmiques et complexité

Âge

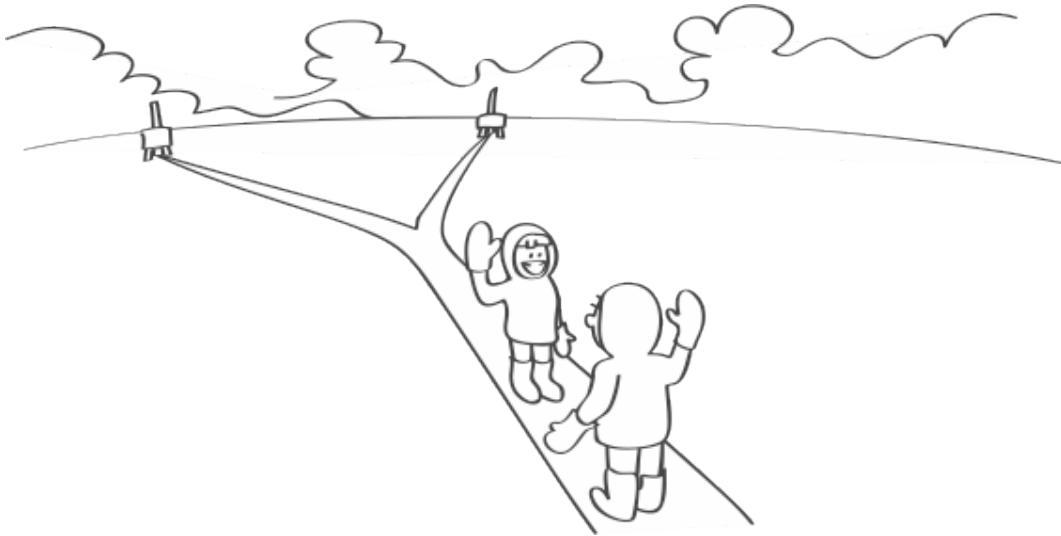
- ✓ 7 ans et plus

Matériel

Pour chaque groupe d'élèves

- ✓ Cinq ou six piquets à planter dans le sol (type piquets de tente ; on peut aussi couper un cintre et tordre les morceaux)
- ✓ Plusieurs mètres de ficelle ou de fil élastique
- ✓ Une grande règle ou un mètre
- ✓ Un stylo et du papier pour prendre des notes

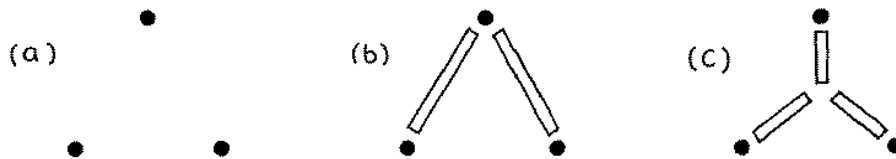
Les routes de glace



Introduction

L'activité précédente, Touristeville, mettait en scène un pays très chaud. Ici, c'est tout le contraire : nous sommes en hiver, dans le Grand Nord glacial du Canada parsemé d'immenses lacs gelés. Les déneigeuses ouvrent des voies reliant les sites de forage pour que les différentes équipes puissent se rendre visite. Il fait très froid, la tâche est difficile dans ces conditions, votre objectif est donc de déterminer le tracé des routes de façon à réduire le travail au minimum. Il n'y a aucune autre contrainte : les routes peuvent être ouvertes n'importe où dans la neige ou sur les lacs gelés, et le terrain est totalement plat.

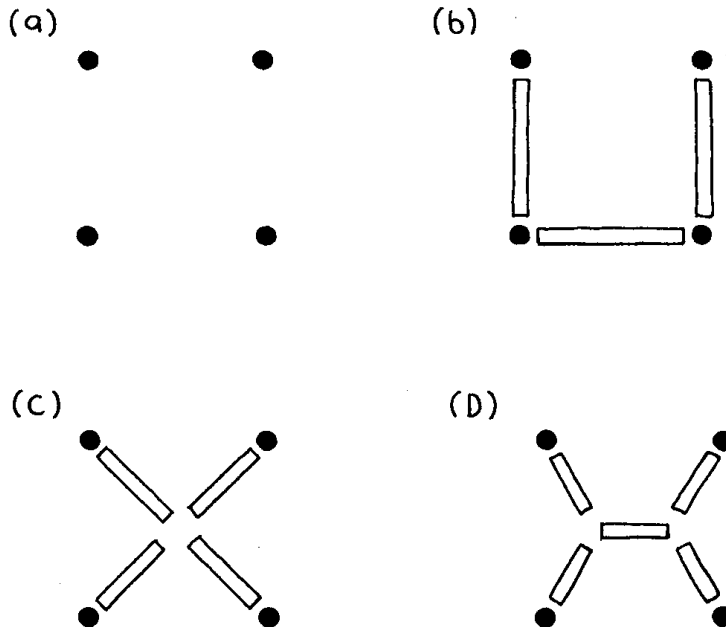
Évidemment, il vaut mieux que les routes soient droites : ajouter des virages ne ferait qu'accroître inutilement leur longueur. Toutefois, le travail ne se réduit pas à relier tous les sites avec des lignes droites, car l'ajout d'intersections peut parfois réduire la longueur totale du réseau routier. Or, c'est cette dernière qui importe, non pas le temps de trajet d'un site à l'autre.



Sur l'illustration ci-dessus, la figure (a) représente trois sites de forage. Relier l'un d'eux aux deux autres (comme en (b)) formerait un réseau routier acceptable. Une autre possibilité consiste à créer vers le centre du triangle un carrefour reliant les trois sites (c). Si l'on mesure la longueur totale des routes déneigées, on voit qu'il s'agit de la meilleure solution. Cette intersection supplémentaire est appelée « point de Steiner », en référence au mathématicien suisse Jacob Steiner (1796-1863) qui fut le premier à formuler le problème et à remarquer que la longueur totale d'un réseau pouvait être réduite par l'adjonction de nouveaux points. On peut imaginer qu'un point de Steiner est un nouveau site de forage imaginaire.

Déroulement de la séance

1. Présentez le problème sur lequel les élèves vont travailler. À l'aide de l'exemple ci-dessus, démontrez que pour trois sites, l'ajout d'un nouveau site peut améliorer la solution en réduisant la longueur totale du réseau routier.



2. Les élèves vont travailler à partir de quatre points agencés en carré, comme sur la figure (a). Sortez en plein air et demandez à chaque groupe de planter quatre piquets au sol de façon à former un carré d'environ 1 mètre par 1 mètre.

3. Demandez aux élèves de relier les piquets avec de la ficelle ou du fil élastique en mesurant et notant la longueur minimale nécessaire. À cette étape de l'expérience, ils ne doivent pas utiliser les points de Steiner. (On atteint alors le minimum en reliant trois côtés du carré, comme en (b), et le réseau mesure alors 3 mètres.)

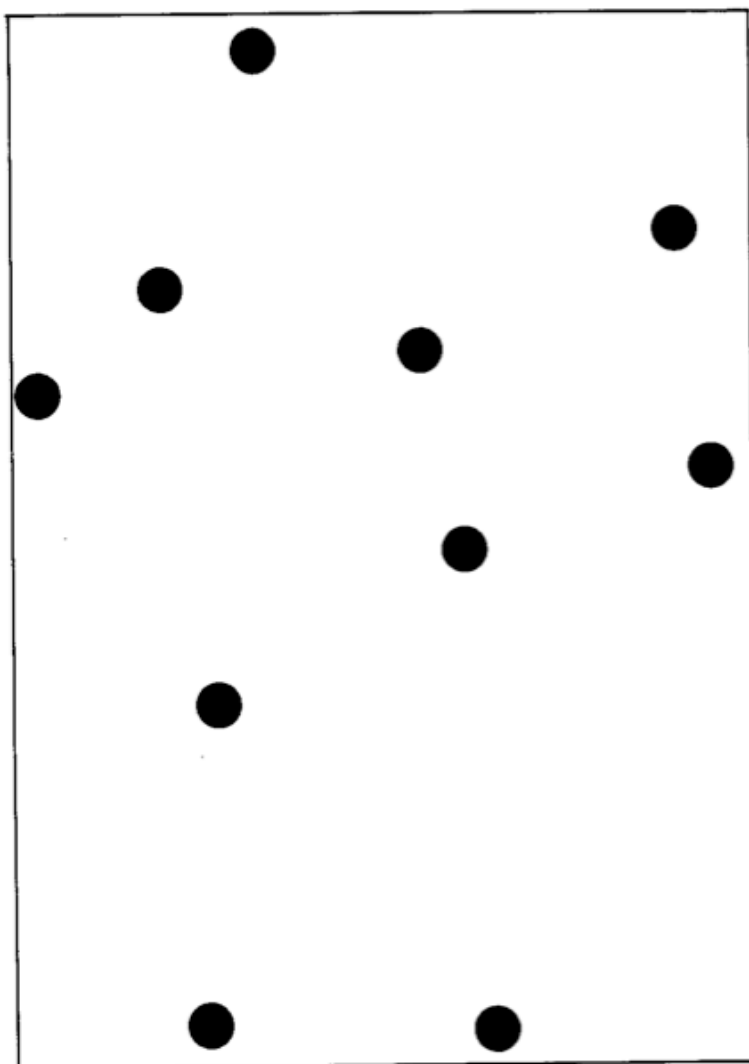
4. À présent, voyez si les élèves peuvent faire mieux en utilisant un point de Steiner. (Le meilleur emplacement est le centre du carré, comme en (c). La longueur totale est alors égale à $2\sqrt{2}$, soit 2,83 mètres.) Dites-leur qu'ils pourraient peut-être faire encore mieux en utilisant deux points de Steiner. (Effectivement, la solution est meilleure si l'on place deux points de Steiner comme en (d), en formant des angles de 120 degrés entre les routes. La longueur totale est alors égale à $1 + \sqrt{3}$, soit 2,73 mètres.)

5. Les élèves peuvent-ils faire mieux avec trois points de Steiner? (Non, la solution à deux points est optimale, en utiliser davantage ne servirait à rien.)

6. Discutez des raisons pour lesquelles ces problèmes paraissent difficiles. (C'est parce qu'on ne sait pas où placer les points de Steiner et qu'on doit donc tester de nombreuses possibilités.)

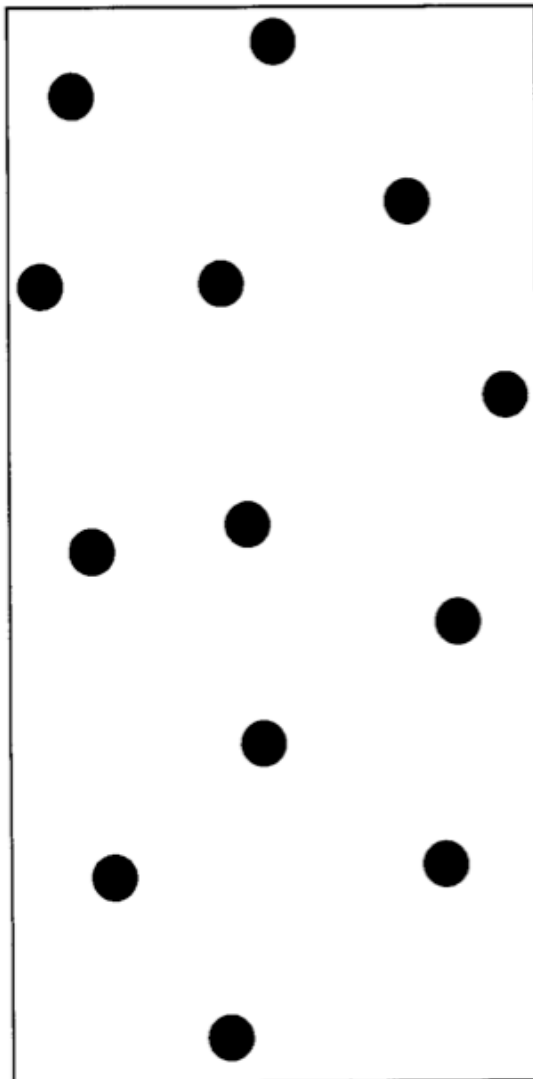
Fiche d'activité : arbre de Steiner Exemple 1

Trouve le moyen de relier ces sites de forage avec le plus court réseau routier possible.



Fiche d'activité : arbre de Steiner Exemple 2

Trouve le moyen de relier ces sites de forage avec le plus court réseau routier possible.

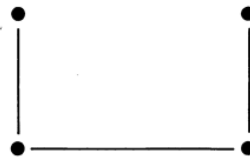


Variantes et activités complémentaires

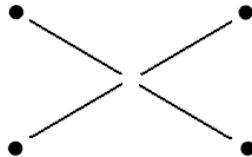
(a)



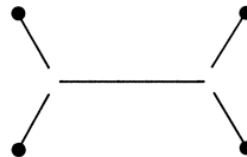
(b)



(c)



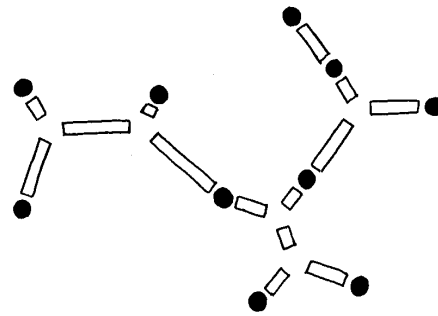
(d)



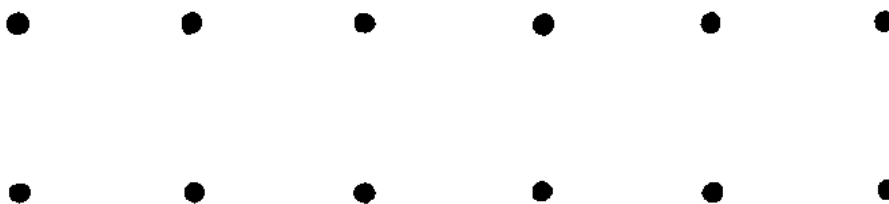
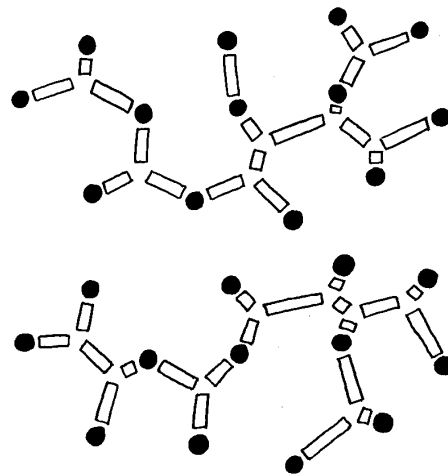
7. Voici une expérience intéressante pour les groupes qui finissent l'activité avant les autres. Faites-les travailler sur un rectangle d'environ 1 mètre par 2 mètres (a). Les élèves découvriront que l'ajout d'un point de Steiner empire les choses alors que l'ajout de deux points est bénéfique (les longueurs sont de 4 mètres pour (b), de $2\sqrt{5} = 4,47$ mètres pour (c), et de $2 + \sqrt{3} = 3,73$ mètres pour (d)). Voyez s'ils parviennent à comprendre pourquoi la configuration à un point est nettement plus mauvaise pour un rectangle que pour un carré. (C'est parce que le rectangle est un carré étiré : cet étirement apparaît une seule fois en (b) et en (d), alors qu'en (c) les deux diagonales sont étirées.)

8. Les élèves plus âgés pourront travailler sur un problème de plus grande envergure. Les fiches d'activité présentent deux configurations de sites à relier avec des routes de glace. Ils pourront tester différentes solutions en utilisant plusieurs copies de la fiche d'activité ou en écrivant avec un feutre effaçable sur un transparent placé au-dessus. On peut également reproduire les cartes à l'aide de piquets plantés dans le sol. Quand un élève ou un groupe pense avoir trouvé la distance minimale, il peut faire part de sa découverte à la classe. (Les illustrations montrent la solution minimale pour le premier exemple et, pour le second, deux solutions possibles offrant une longueur totale à peu près égale.) L'existence de solutions équivalentes montre toute la complexité de ce genre de problèmes. Il y a tellement d'emplacements possibles pour les points de Steiner qu'on a l'embarras du choix !

Arbre de Steiner pour le premier exemple



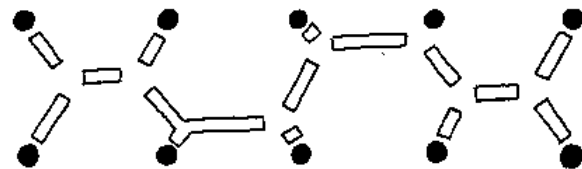
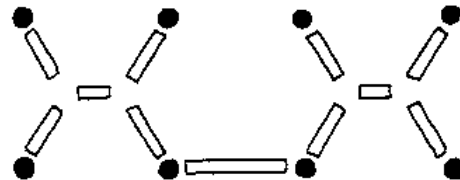
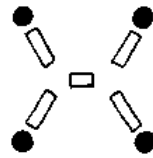
Deux arbres de Steiner possibles pour le second exemple



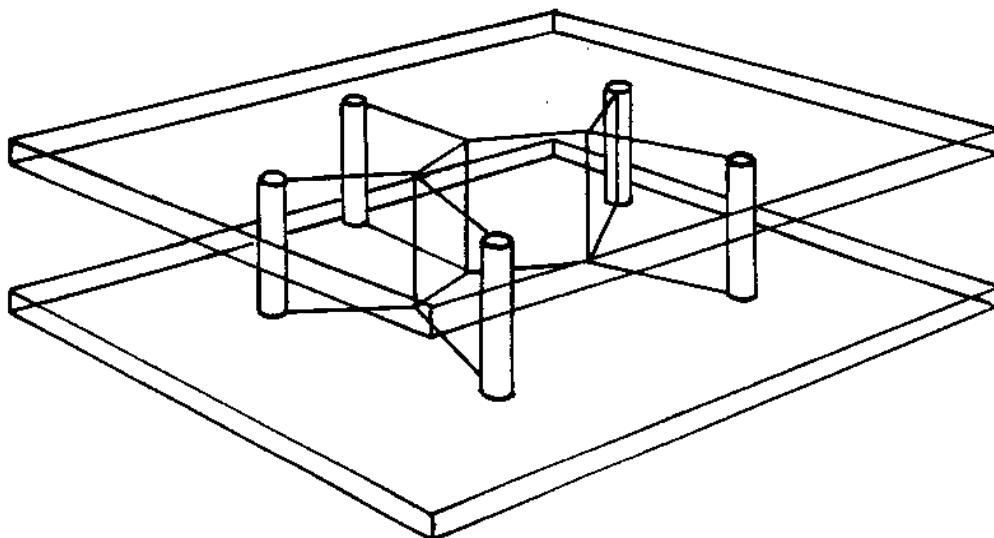
9. Des réseaux en échelle comme celui-ci offrent une autre occasion d'élargir le problème.

Voici quelques arbres de Steiner minimaux pour des réseaux en échelle.

Celui d'une échelle à deux barreaux est exactement comme celui d'un carré. Cependant, la solution est tout à fait différente pour une échelle à trois barreaux, comme vous pourrez le constater si vous essayez de la reproduire de mémoire ! Pour une échelle à quatre barreaux, il suffit de relier les arbres de deux échelles à deux barreaux, alors que pour une échelle à cinq barreaux, la solution s'apparente à celle d'une échelle à trois barreaux. En général, la forme d'un arbre de Steiner minimal pour une échelle dépend du nombre, pair ou impair, de barreaux. Si le nombre est pair, cela reviendra à rassembler plusieurs échelles à deux barreaux. Sinon, il faudra répéter la solution pour trois barreaux. Cependant, il n'est pas facile de démontrer ces faits.



10. Une autre activité intéressante consiste à fabriquer un arbre de Steiner en bulles de savon. Pour ce faire, prenez deux feuilles rigides transparentes (des transparents rigides pour rétroprojecteur) et insérez des épingles entre elles pour représenter les sites à couvrir, comme sur l'illustration suivante.



Plongez l'ensemble dans de l'eau savonneuse. Quand vous le ressortirez, vous remarquerez qu'une pellicule de savon relie les épingles en formant un magnifique arbre de Steiner.

Malheureusement, il ne s'agira pas forcément de la solution minimale. Certes, la pellicule de savon minimise la longueur totale, mais il s'agit d'un minimum au niveau local, non pas au niveau global. Une configuration totalement différente des points de Steiner pourrait donc permettre de réduire la longueur totale du réseau. Ainsi, on peut imaginer que la pellicule de savon ressemble à la première solution du second exemple quand on la sort de l'eau, puis à la deuxième solution quand on renouvelle l'opération.

Ce qu'il faut retenir

Les réseaux sur lesquels nous avons travaillé sont des *arbres de Steiner minimaux*. On les appelle « arbres » parce qu'ils ne forment pas de cycle, à l'instar des vrais arbres dont les branches poussent chacune de leur côté sans jamais se rejoindre pour poursuivre ensemble leur croissance (du moins en temps normal). On parle d'arbre « de Steiner » parce qu'il est possible d'ajouter de nouveaux points (des points de Steiner) au réseau original. Enfin, ils sont dits « minimaux » parce que leur longueur totale est inférieure à tout autre arbre reliant les sites initiaux. Dans la ville embourbée (Activité 9), nous avons appris qu'un réseau reliant un certain nombre de sites avec une longueur totale minimum était un « arbre couvrant minimal » : les arbres de Steiner sont exactement la même chose, sauf qu'il est possible d'ajouter de nouveaux points au réseau.

Il est intéressant de noter que s'il existe un algorithme très efficace pour trouver des arbres couvrants minimaux (voir l'Activité 14 et l'algorithme glouton, qui fonctionne en reliant, étape par étape, les deux points du réseau les plus proches), nous ne disposons d'aucune solution efficace au problème de Steiner. Les arbres de Steiner sont nettement plus compliqués parce qu'il faut choisir où placer les points supplémentaires. Chose assez étonnante, le plus difficile n'est pas de déterminer l'endroit précis des points de Steiner, mais plutôt de choisir approximativement où les placer (voyez par exemple la différence entre les deux solutions de la deuxième fiche d'activité). Une fois que l'on sait dans quelles zones placer les nouveaux points, le choix du positionnement optimal est relativement simple. L'eau savonneuse le fait très bien, les ordinateurs le peuvent également.

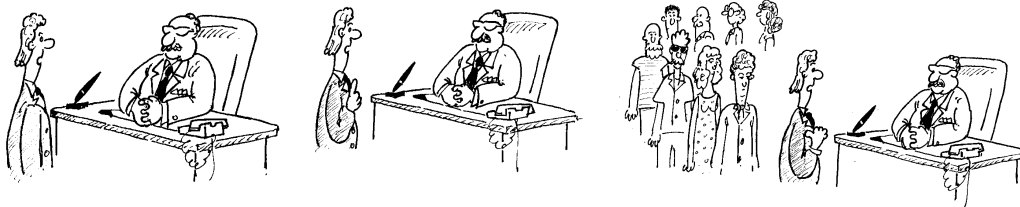
Les arbres de Steiner ont permis à certaines entreprises de réduire significativement leurs factures de téléphone. Avant 1967, aux États-Unis, les entreprises qui exploitaient un grand réseau téléphonique privé louaient leurs lignes à des compagnies de téléphonie. Le montant de la facture n'était pas calculé en fonction de l'utilisation effective des câbles mais du plus court réseau théoriquement nécessaire, l'idée étant que les consommateurs ne devaient pas payer plus cher sous prétexte que la compagnie téléphonique n'utilisait pas les plus courts réseaux. À l'origine, l'algorithme utilisé pour calculer la facture déterminait l'arbre couvrant minimal. Mais, vers 1967, un consommateur (en l'occurrence, une compagnie aérienne dotée de trois centraux) a remarqué qu'elle pouvait réduire la longueur totale du réseau en ajoutant un quatrième central. La compagnie de téléphonie s'est donc vue contrainte de réduire ses tarifs à ce qu'ils auraient été s'il y avait eu un central placé sur un point de Steiner ! Certes, pour une configuration classique, l'arbre de Steiner n'est que 5 à 10% plus court que l'arbre couvrant minimal, mais cela représente tout de même une économie substantielle lorsque de grosses sommes sont en jeu. Le problème de l'arbre de Steiner est parfois appelé « problème du plus court réseau ». Il s'agit en effet de trouver le plus court réseau possible pour relier un ensemble de sites.

Si vous vous êtes déjà attaqué aux deux activités précédentes, celles du cartographe et de Touristeville, vous ne serez pas surpris d'apprendre que le problème de l'arbre de Steiner est NP-complet. Le nombre d'emplacements possibles pour les points de Steiner augmente avec le nombre de sites, et la recherche à effectuer pour tester toutes les

possibilités s'accroît alors de façon exponentielle. Là encore, c'est l'un des milliers de problèmes pour lesquels on ne sait tout simplement pas si une recherche exhaustive (avec un algorithme exponentiel) est la meilleure solution... ou si un algorithme polynomial reste encore à découvrir. On sait toutefois que si quelqu'un découvrait un algorithme en temps polynomial pour résoudre ce problème, on pourrait l'adapter à la coloration de graphes et à l'arbre couvrant minimal, ainsi qu'à tous les autres problèmes NP-complets.

Nous avons défini ces problèmes à la fin de l'activité précédente : NP signifie « polynomial non-déterministe » et l'adjectif « complet » implique que si un algorithme en temps polynomial était découvert pour l'un des problèmes NP-complets, il pourrait être adapté à tous les autres. La classe de problèmes qu'on sait résoudre en temps polynomial est appelée « P ». La question centrale est donc la suivante : Existe-t-il des algorithmes en temps polynomial pour les problèmes NP-complets ? Ou autrement dit : Est-ce que $P = NP$? La réponse à cette question n'est pas connue à ce jour, c'est l'un des grands mystères de la science informatique actuelle.

Les problèmes pour lesquels un algorithme en temps polynomial existe (aussi lent soit-il) sont dits « traitables ». Les autres sont dits « intraitables », car quels que soient le nombre et la vitesse des ordinateurs, une petite augmentation de la taille du problème les rendra insolubles en pratique. On ne sait pas si les problèmes NP-complets – dont font partie les problèmes du cartographe, de Touristeville et des routes de glaces – peuvent, oui ou non, être résolus en pratique. Cependant, de nombreux chercheurs en informatique doutent qu'on parvienne un jour à trouver un algorithme en temps polynomial pour résoudre les problèmes NP-complets. Démontrer qu'un problème est NP-complet est donc considéré comme une preuve solide de son « intraitabilité ».



« Je n'ai pas trouvé d'algorithme efficace. Je suis sans doute trop bête... »

« Je n'ai pas trouvé d'algorithme efficace parce qu'il n'y en a pas ! »

« Je n'ai pas trouvé d'algorithme efficace, mais tous ces gens connus non plus ! »

Que faire quand on ne trouve pas d'algorithme efficace : trois possibilités

Que faire si votre patron vous demande d'inventer un algorithme efficace pour trouver la solution optimale à un problème, et que vous n'y arrivez pas ? C'est sans doute arrivé le jour où la compagnie aérienne a découvert qu'elle pouvait réduire les coûts de son réseau en ajoutant des points de Steiner. Si l'on était capable de *démontrer* qu'aucun algorithme efficace ne pouvait donner la solution optimale, cela serait formidable. Mais, en informatique, il est très difficile de prouver de tels résultats négatifs : qui sait si un programmeur malin ne découvrira pas un jour une obscure combine permettant de résoudre le problème ? Il est donc malheureusement peu probable que vous soyez en

mesure d'affirmer catégoriquement qu'aucun algorithme efficace n'existe, autrement dit, que le problème est « intraitable ». Cela dit, si vous parvenez à démontrer que le problème est NP-complet, alors vous pouvez arguer que des milliers de chercheurs ont travaillé sur des problèmes équivalents et qu'eux non plus n'ont pas trouvé de solution efficace. Eux aussi ont échoué : vous n'aurez peut-être pas de prime, mais vous serez au moins tiré d'affaire !

Cependant, il faut bien résoudre ces problèmes dans la vie de tous les jours. On a alors recours à l'*heuristique*, c'est-à-dire à des algorithmes qui ne garantissent pas la meilleure solution mais donnent néanmoins une solution très proche de l'optimum. Les méthodes de calcul heuristiques peuvent être très rapides, et l'écart avec la solution optimale peut être assez faible. Elles s'avèrent donc suffisantes, même s'il est un peu frustrant de savoir qu'on pourrait avoir des emplois du temps ou des réseaux routiers légèrement meilleurs.

Suggestions de lecture (en anglais)

La bande dessinée ci-dessus provient de l'ouvrage de Garey et Johnson, *Computers and Intractability*.

La rubrique « Computer recreations » du *Scientific American* de juin 1984 explique brièvement comment fabriquer des arbres de Steiner avec des bulles de savon. Elle présente d'autres travaux pratiques pour la résolution de problèmes, notamment la machine à spaghettis pour trier (voir une vidéo de démonstration sur <https://interstices.info/calculer-differemment>), un jeu de ficelles pour trouver les plus courts chemins sur un graphe, et un dispositif de miroirs et de lumières pour déterminer si un nombre est premier. Ces jeux figurent également dans la partie dédiée aux ordinateurs analogiques du *Turing Omnibus* de Dewdney.

Partie V

**Partager des secrets et lutter
contre la criminalité – *La
cryptographie***

Partager des secrets et lutter contre la criminalité

Sans doute avez-vous entendu parler d'espions ou d'agents secrets qui échangeaient des messages à l'aide de codes secrets ou d'encre invisible. Il s'agissait là des débuts de la cryptographie, l'art d'écrire et de déchiffrer les codes secrets. Pendant la Seconde Guerre mondiale, les Anglais ont mis au point des machines électroniques capables de décoder les messages secrets militaires. Puis les premiers ordinateurs sont apparus, et la cryptographie est entrée dans une nouvelle ère : on pouvait désormais déchiffrer les codes en effectuant de gigantesques calculs jusque-là inimaginables. Ensuite, quand les gens se sont mis à partager les mêmes systèmes informatiques, une nouvelle utilisation des mots de passe a vu le jour. Et quand les ordinateurs ont été reliés les uns aux autres, il y a eu de nouvelles raisons de protéger les données contre ceux qui auraient voulu les intercepter. Avec l'arrivée des mails, il est devenu important de s'assurer que les personnes qui signent les messages sont bien celles qu'elles prétendent être. Et maintenant que l'on peut gérer ses comptes bancaires, vendre et acheter des biens sur Internet, nous avons impérativement besoin de sécuriser les prises de commandes et les transactions bancaires. Enfin, la menace grandissante d'une attaque terroriste sur les systèmes informatiques donne encore plus de poids à ces questions de sécurité.

Quand on parle de cryptographie, on imagine des ordinateurs qui stockent des messages secrets, par exemple des mots de passe, en mélangeant toutes les lettres pour empêcher l'ennemi de les déchiffrer. Mais la réalité est toute différente. Les systèmes informatiques modernes *ne stockent pas* de mots de passe, car celui qui parviendrait à s'y introduire serait alors en mesure de décrypter tous les codes de sécurité. Ce serait désastreux : il pourrait effectuer de fausses transactions bancaires, envoyer des messages sous une autre identité, lire les dossiers secrets de tout le monde, donner des ordres aux armées et même entraîner la chute de gouvernements. Aujourd'hui, les mots de passe sont traités à l'aide des « fonctions à sens unique » que nous avons évoquées dans l'Activité 14. Par ailleurs, le chiffrement *ne consiste pas* simplement à inverser les lettres des mots : il s'effectue à l'aide de techniques impliquant des problèmes vraiment difficiles, tels les problèmes « intraitables » présentés dans la Partie IV.

La cryptographie permet des choses qu'on aurait crues impossibles. Cette partie présente un moyen simple de calculer l'âge moyen d'un groupe sans qu'aucun de ses membres ne divulgue son âge. Vous découvrirez aussi comment deux personnes qui ne se font pas confiance peuvent jouer à pile ou face et s'accorder sur le résultat même s'ils ne sont pas au même endroit et qu'ils ne peuvent donc pas voir comment la pièce retombe. Enfin, vous apprendrez à chiffrer des messages secrets qu'une seule personne peut décoder bien que tout le monde sache comment les encoder.

Pour les enseignants

Les activités suivantes permettent d'appréhender de façon concrète le fonctionnement des techniques cryptographiques modernes. Elles sont très différentes de ce que l'on s'imagine d'ordinaire quand on pense à la confidentialité et aux ordinateurs.

Il y a deux notions clés. La première est la notion de « protocole », soit l'ensemble de règles définissant le mode de communication entre deux ordinateurs. Le mot évoque la diplomatie, voire même un cérémonial, mais les ordinateurs l'utilisent aussi ! Des

protocoles très simples permettent parfois de résoudre des tâches apparemment difficiles. L'Activité 16, qui demande quelques minutes seulement, montre comment un groupe de personnes qui coopèrent peut facilement déterminer son âge moyen (ou son revenu moyen) sans que personne ne connaisse l'âge (ou le revenu) des autres. La seconde notion clé est relative au rôle de la complexité (ou de l'intraitabilité) lorsqu'on interagit avec autrui par le biais des ordinateurs. L'Activité 17 démontre que deux personnes qui ne se font pas forcément confiance peuvent jouer à pile ou face par téléphone (elle présente par ailleurs le concept de circuit booléen, en expliquant comment le mettre en œuvre). Enfin, l'Activité 18 présente des techniques de calcul qui permettent de chiffrer les messages en toute sécurité alors même que la méthode d'encodage est connue de tous.

Certaines activités sont difficiles, en particulier la dernière. Il faudra motiver la classe en lui insufflant l'idée que ces techniques ont quelque chose d'extraordinaire, qu'elles permettent d'accomplir des choses que la plupart des gens considèrent comme impossibles. Il est essentiel de susciter cet émerveillement, de le communiquer aux élèves et de faire des pauses fréquentes pour que la flamme ne s'éteigne pas au cours de l'activité. Il ne faudrait pas que l'arbre (parfois un peu ennuyeux) leur cache la forêt (stupéfiante !). Ces activités sont parmi les plus stimulantes du livre, même si ce sont les plus complexes techniquement. Si elles s'avèrent trop difficiles, passez directement à la Partie VI, qui est totalement différente, beaucoup moins technique.

Pour ceux qui ont l'esprit scientifique

Alors que les ordinateurs s'immiscent de plus en plus dans nos vies quotidiennes, l'usage de la cryptographie s'avère potentiellement tendancieux. La plupart des gens n'ont pas conscience de tout ce dont les protocoles cryptographiques sont capables. Quand de grands établissements (tant publics que privés) mettent en place des systèmes utilisant des informations personnelles, ce sont généralement les technocrates qui prennent les décisions importantes relatives au traitement des données, choisissant ce qui sera collecté, ce qui sera disponible, et qui pourra y avoir accès. Si les gens comprenaient mieux les possibilités qu'offrent les nouvelles technologies, ils seraient en mesure de participer plus activement à ces prises de décision, ce qui pourrait finalement donner lieu à une infrastructure d'information totalement différente.

Si ces protocoles de masquage des données, de cryptographie, et de chiffrement à clé publique sont généralement considérés comme complexes, les concepts eux-mêmes n'ont rien de compliqué. Ce sont les détails techniques, non pas les idées de base, qui sont difficiles à comprendre. Dans des situations pratiques de commerce électronique, les détails techniques sont dissimulés dans les logiciels, ce qui rend les nouvelles technologies de chiffrement très faciles d'utilisation. Il est toutefois important de saisir les idées qui les sous-tendent si l'on veut mieux comprendre les possibilités qui s'offrent à nous.

Les systèmes cryptographiques intéressent tout particulièrement les gouvernements, non seulement pour assurer la sécurité de leurs communications officielles, mais aussi parce qu'ils craignent que les messages chiffrés ne soient utilisés à des fins illégales, par des trafiquants de drogue ou des terroristes, par exemple. Si ces individus utilisent le chiffrement, alors les écoutes téléphoniques ne servent plus à rien, à moins d'avoir une méthode de décryptage à disposition. Ces craintes ont suscité de nombreux débats entre les responsables du maintien de l'ordre (qui veulent limiter la puissance des systèmes de chiffrement) et les défenseurs des libertés civiles (qui voient d'un mauvais œil le fait que l'État ait accès à des communications privées). Depuis un certain temps, le gouvernement américain restreint l'utilisation de certaines méthodes cryptographiques en les considérant comme des munitions, au même titre que les bombes ou les armes à feu. N'importe qui,

pourvu qu'il soit bien informé et doté de certaines compétences techniques, est capable d'installer un système de communication sécurisé. Or, cela peut s'avérer dangereux si la technologie en question tombe en de mauvaises mains. Pour citer un exemple déjà ancien, dans les années 1980, il y a eu un débat intense autour de la puce Clipper, un projet de *d'autorité de séquestre* lancé par la NSA. Son principe est qu'un organisme étatique détient un duplicata de la clé privée ou du mot de passe, et peut donc déchiffrer tout message encodé sur la puce. Le FBI et le Département de la justice américain voulaient en généraliser l'utilisation, mais cela a suscité une vive protestation en raison des menaces qui pesaient sur la protection de la vie privée. Ce type de problème reste d'actualité. Toutes sortes de systèmes cryptographiques sont techniquement possibles, mais pas forcément politiquement acceptables !

La cryptographie ne sert pas seulement à envoyer des messages secrets, elle a de nombreuses applications. Elle permet notamment de s'assurer que les messages ont bien été envoyés par les personnes qui disent les avoir envoyés (c'est ce qu'on appelle l'« authentification »), et elle rend possible le commerce électronique. Toujours grâce à elle, on peut procéder à un vote par ordinateur sans que personne ne puisse savoir pour qui on a voté – pas même ceux qui gèrent le système informatique –, et sans qu'on puisse voter plusieurs fois. Elle permet même de jouer aux cartes par téléphone – ce qui peut paraître idiot mais ne l'est plus quand on comprend que les affaires commerciales ressemblent énormément à une partie de poker.

On s'imagine difficilement en train de battre le jeu pour entamer une partie de cartes par téléphone : comment faire confiance à son partenaire s'il n'est pas dans la pièce ? Et comment repérer quelqu'un qui intercepte un message et le modifie avant de le renvoyer ? Tout cela semble impossible. Pourtant, si nous ne pouvions pas le faire, le commerce électronique serait tout bonnement impossible. Il *faut* empêcher les criminels de créer de fausses autorisations pour des retraits bancaires en interceptant une communication téléphonique entre un terminal point de vente et la banque. Il *faut* empêcher les entreprises concurrentes de semer la zizanie en générant de fausses commandes ou de faux contrats. Grâce aux techniques cryptographiques modernes, de telles actions sont possibles. Les activités suivantes vous feront découvrir leur fonctionnement.

Il existe de nombreux livres intéressants sur le chiffrement et le déchiffrement. *Codebreakers: the inside story of Bletchley Park*, de Hinsley et Stripp, raconte comment les premiers ordinateurs ont servi à décoder des messages secrets pendant la Seconde Guerre mondiale, réduisant significativement la durée de la guerre et sauvant de nombreuses vies.

Parmi les ouvrages disponibles en français, nous vous recommandons le livre de Simon Singh *Histoire des codes secrets*, Le Livre de Poche, 2001 et *La Science du secret* de Jacques Stern, éditions Odile Jacob, 1997.

Activité 16

Partager des secrets – *Protocoles de masquage des données*

Résumé

Les techniques cryptographiques permettent de partager des données avec d'autres personnes tout en garantissant un niveau de confidentialité très élevé. Dans cette activité, nous allons partager des informations sans rien révéler de leur contenu : chaque groupe d'élèves calculera son âge moyen sans qu'aucun des membres ne divulgue son âge.

Liens pédagogiques

- ✓ Mathématiques : sommes et moyennes

Compétences

- ✓ Calculer une moyenne
- ✓ Nombres aléatoires
- ✓ Coopération

Âge

- ✓ 7 ans et plus

Matériel

Pour chaque groupe d'élèves :

- ✓ Un petit bloc-notes
- ✓ Un stylo



Partager des secrets

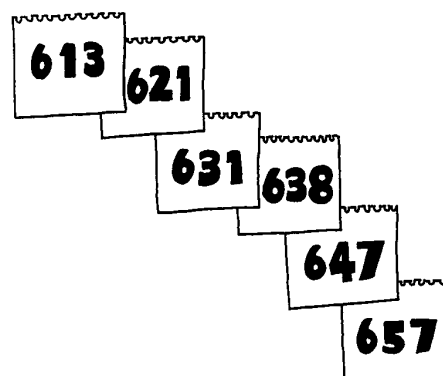
Introduction

Dans cette activité, l'objectif est de trouver l'âge moyen d'un groupe sans qu'aucun de ses membres ne révèle son âge. On peut également travailler sur le revenu moyen des élèves (leur argent de poche) ou sur d'autres informations personnelles similaires. L'activité fonctionne particulièrement bien avec les adultes, ces derniers étant parfois plus sensibles à la divulgation de telles données.

Il faut au moins trois personnes par groupe.

Déroulement de la séance

1. Expliquez aux groupes que vous voudriez déterminer leur âge moyen sans que personne ne révèle son âge. Demandez-leur s'ils ont une idée de comment faire, ou s'ils pensent que c'est possible.
2. Choisissez six à dix élèves avec qui travailler. Donnez le bloc-notes et le stylo au premier élève et demandez-lui de choisir au hasard un nombre à trois chiffres qu'il écrira sur la feuille du dessus. Dans cet exemple, c'est le 613 qui a été choisi.
3. Demandez-lui d'arracher la première page, d'additionner son âge avec le nombre aléatoire et d'écrire le total sur la deuxième feuille du bloc-notes. Le premier élève a huit ans, la deuxième page indique donc 621. Il conservera la première page arrachée (et ne la montrera à personne).
4. Il passe ensuite le bloc-notes au deuxième élève. Celui-ci additionne son âge et le nombre inscrit sur la page, déchire cette dernière et note le total sur la page suivante. Dans notre exemple, le deuxième élève a donc dix ans.



5. Poursuivre ainsi l'activité (les élèves déchirent la page du dessus, ajoutent leur âge au nombre déjà inscrit et notent le total sur la page suivante) jusqu'à ce que le bloc-notes soit passé entre toutes les mains.
6. Le bloc-notes revient au premier élève. Celui-ci soustrait le nombre aléatoire du début au nombre qui figure sur le bloc-notes. Dans notre exemple, le groupe est composé de cinq élèves, on soustrait 613 (le nombre initial) au nombre final, 657, ce qui donne 44. Ce dernier nombre est la somme des âges des élèves. On peut donc calculer la moyenne d'âge en le divisant par le nombre d'élèves. Ici, l'âge moyen est de 8,8 ans.
7. Faites remarquer aux élèves que personne ne peut connaître l'âge de quelqu'un d'autre si tout le monde détruit sa feuille de papier (sauf, bien sûr, dans le cas où deux personnes décident de coopérer).

Variantes et activités complémentaires

On peut adapter ce système afin de procéder à un vote secret. Dans ce cas, les élèves ajoutent un s'ils votent « oui », et zéro s'ils votent « non ». Le vote est bien sûr faussé si un élève ajoute un nombre supérieur à un (ou inférieur à zéro), mais ce dernier court le risque d'éveiller les soupçons si tous les autres ont voté « oui », car le nombre de votes favorables serait alors supérieur au nombre de votants.

Ce qu'il faut retenir

Les ordinateurs stockent beaucoup de données personnelles : l'état de notre compte en banque, les impôts à payer, l'année d'obtention de notre permis de conduire, nos antécédents de crédit, nos résultats aux examens, nos dossiers médicaux, etc. La protection de la vie privée est donc très importante, mais il faut bien partager certaines de ces informations avec d'autres personnes. Ainsi, quand nous effectuons un achat par carte bancaire, nous admettons que le magasin ait besoin de vérifier s'il y a suffisamment d'argent sur notre compte.

Finalement, nous fournissons souvent plus d'informations que nécessaire. Si nous effectuons une transaction électronique dans un magasin, par exemple, nous divulguons notre nom, celui de notre banque, ainsi que notre numéro de compte. De plus, la banque saura où nous avons fait nos achats. Elle pourrait tout à fait établir notre profil en notant l'endroit et l'heure où nous allons faire le plein ou les courses, et combien nous dépensons chaque jour en essence ou en nourriture. Si nous avons payé en liquide, aucune de ces informations n'aurait été révélée. La plupart des gens ne s'inquiètent pas trop devant la quantité de données partagées. Pourtant, les risques d'abus sont bien réels, que ce soit du marketing ciblé (envoyer des publicités pour des voyages à des gens qui dépensent beaucoup d'argent en billets d'avion, par exemple), de la discrimination (offrir un meilleur service à quelqu'un dont la banque s'adresse généralement à des clients riches), ou même du chantage (menacer de divulguer les détails d'une affaire embarrassante). Quoi qu'il en soit, mieux vaut changer ses habitudes d'achat si l'on pense que quelqu'un pourrait nous surveiller.

Les atteintes à la vie privée sont assez largement acceptées, pourtant certains protocoles permettent d'effectuer des transactions financières électroniques en garantissant le même niveau de confidentialité qu'avec de l'argent liquide. Cela peut paraître difficile à comprendre, mais il est possible de transférer de l'argent depuis son compte en banque vers celui du magasin sans que personne ne sache d'où vient l'argent ni où il va. Cette activité permet de mieux comprendre comment cela est possible : dans les deux situations, il s'agit de limiter le partage d'informations à l'aide d'un protocole ingénieux.

Suggestions de lecture (en anglais)

David Chaum a écrit un article sur le sujet, désormais classique, sous le titre provocateur « Security without identification : transaction systems to make Big Brother obsolete » (La sécurité sans identification : systèmes de transaction pour rendre caduque Big Brother). L'article est agréable à lire, il donne des exemples simples de protocoles de protection des données, y compris un moyen d'effectuer des transactions totalement confidentielles à l'aide d'une « monnaie électronique ». Il se trouve dans le numéro d'octobre 1985 de *Communications of the ACM*.

Activité 17

Le pile ou face péruvien – *Protocoles cryptographiques*

Résumé

L'activité montre comment accomplir une tâche simple mais apparemment impossible : deux personnes qui ne se font pas forcément confiance vont jouer à pile ou face par téléphone.

Liens pédagogiques

- ✓ Mathématiques : raisonnement logiques
- ✓ Mathématiques : logique booléenne

Compétences

- ✓ Logique booléenne
- ✓ Fonctions
- ✓ Résolution d'énigmes

Âge

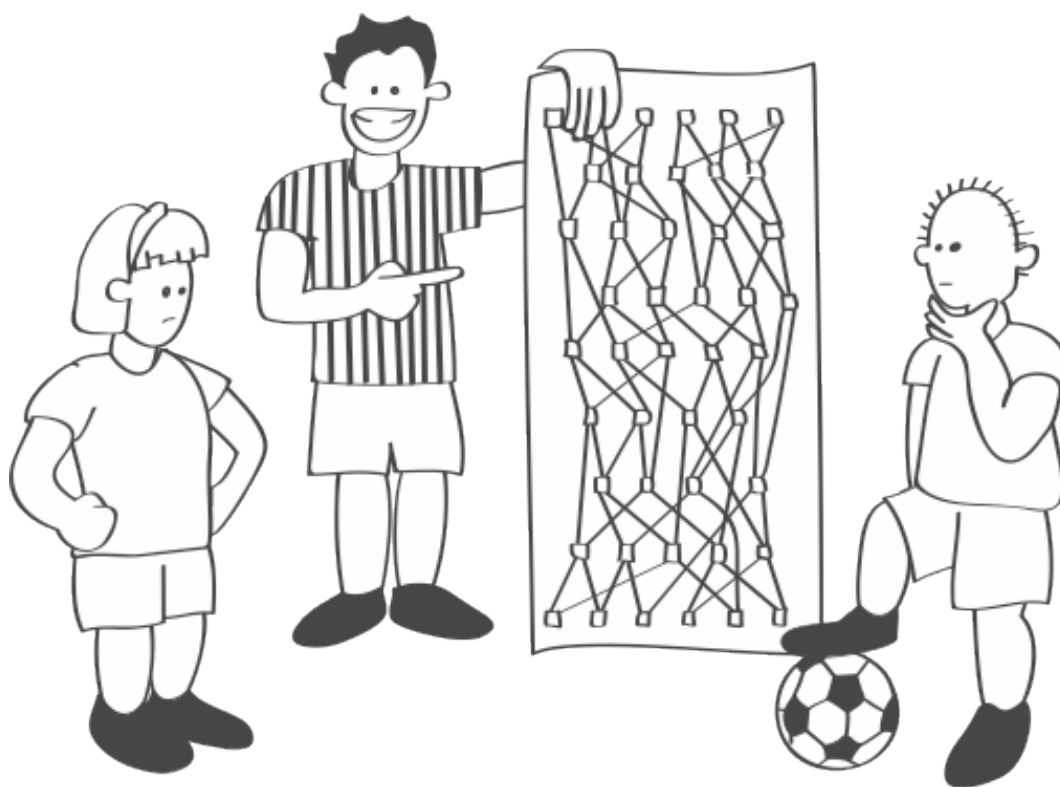
- ✓ 9 ans et plus

Matériel

Pour chaque groupe d'élèves :

- ✓ Une copie de la fiche d'activité *Le pile ou face péruvien*
- ✓ Environ deux dizaines de petits boutons ou de jetons de deux couleurs

Le pile ou face péruvien



Les enfants comprendront mieux l'activité s'ils connaissent déjà les nombres binaires (Activité 1, Compter les points) et le concept de *parité* (Activité 4, Le tour de cartes), et s'ils ont déjà étudié l'exemple de fonction à sens unique de l'Activité 14, Touristeville.

Introduction

Cette activité a été conçue par l'un de nos auteurs (MRF) alors qu'il travaillait avec des élèves péruviens, d'où son nom. Vous pouvez bien sûr adapter l'histoire au contexte local.

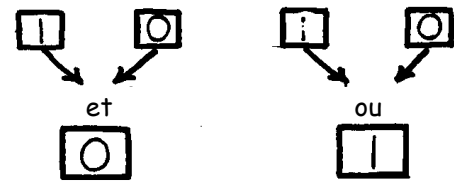
Les équipes de football de Lima et de Cuzco doivent décider laquelle accueillera le match du championnat. Le plus simple serait de jouer à pile ou face, mais les deux villes sont éloignées et les représentants des équipes, Alicia et Benito, ne peuvent pas se permettre de perdre du temps et de l'argent avec un voyage en avion. Peuvent-ils le faire par téléphone ? Disons qu'Alicia lance la pièce et que Benito choisisse pile ou face : cela ne fonctionnerait pas car si Benito choisissait face, Alicia pourrait toujours lui dire : « Désolée, mais c'est pile », et il serait incapable de savoir si c'est vrai. Alicia n'est pas une menteuse, mais c'est tout de même un match important et la tentation de tricher est grande. D'ailleurs, même si Alicia se montrait honnête, comment Benito pourrait-il la croire s'il perdait ?

Voici donc comment ils décident de procéder. Ils conçoivent ensemble un circuit composé de portes logiques ET et de portes logiques OU, comme expliqué ci-après. Théoriquement, ils peuvent le faire par téléphone (et cela fonctionne aussi avec des mails !), mais il faut reconnaître que cela peut s'avérer très fastidieux en pratique. Lors de

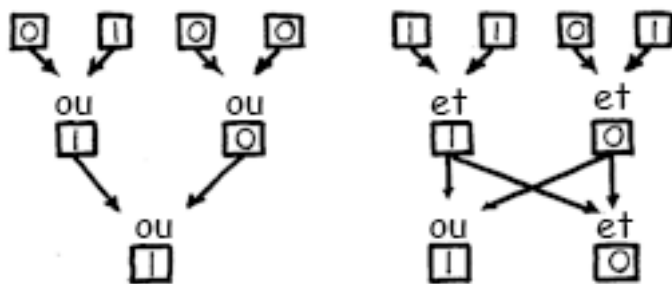
la conception, les deux parties ont intérêt à s'assurer que le circuit est suffisamment complexe pour empêcher l'autre de tricher. Le circuit final est connu de tous.

Déroulement de la séance

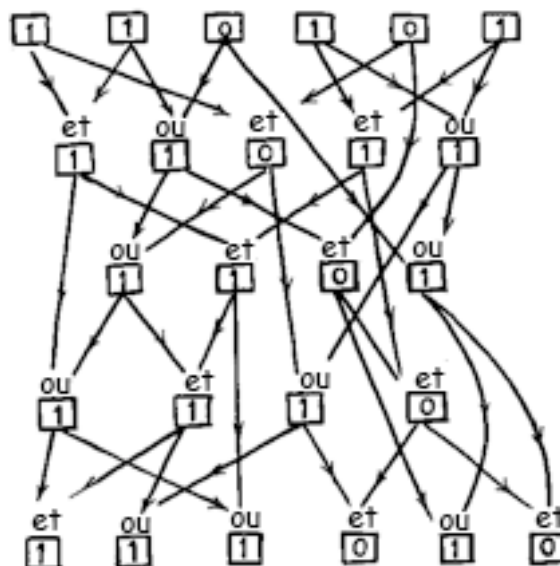
Les règles qui régissent les portes logiques ET et OU sont simples. Chaque porte a deux entrées et une sortie. Chaque entrée peut être soit *zéro* soit *un* (on peut considérer que *zéro* et *un* représentent respectivement *vrai* et *faux*). La sortie d'une porte ET est *un* (*vrai*) seulement si les deux entrées sont des *uns* (*vrai*), sinon elle donne *zéro* (*faux*). Par exemple, la porte ET ci-dessus a *un* et *zéro* en entrées (en haut), la sortie (le carré du bas) est donc *zéro*. La sortie d'une porte OU est *un* (*vrai*) si l'une des deux entrées (ou les deux) est *un* (*vrai*), et *zéro* (*faux*) si les deux entrées sont *zéros*. La sortie d'une porte OU est donc *un* si on a un et *zéro* en entrée.



Pour compliquer les choses, on peut relier la sortie d'une porte à l'entrée d'une autre (ou de plusieurs). Dans le circuit de gauche, par exemple, les sorties de deux portes OU sont reliées aux entrées d'une troisième porte OU. Si l'une des quatre entrées est *un*, le résultat sera donc *un*. Dans le circuit de droite, les sorties des deux portes ET du haut alimentent chacune les deux portes du bas, de sorte que l'ensemble du circuit a deux valeurs de sortie.



Pour le pile ou face péruvien, il nous faut un circuit encore plus complexe. Le circuit de la fiche d'activité comporte six entrées et six sorties. Voici un exemple pour un ensemble d'entrées donné.



L'utilisation du circuit pour jouer à pile ou face par téléphone est la suivante : Alicia choisit des entrées aléatoires, soit une séquence de six chiffres binaires (des zéros et des uns, également appelés « bits ») qu'elle garde secrète. Elle introduit les six chiffres dans le circuit et envoie à Benito les six bits de sortie. Celui-ci doit alors deviner si l'entrée d'Alicia est composée d'un nombre pair ou impair de *uns* – autrement dit, il doit deviner la *parité* de l'entrée. Si le circuit est suffisamment complexe, Benito sera incapable de trouver la solution et devra donc répondre au hasard (à vrai dire, il pourrait même jouer à pile ou face pour choisir !). Si sa réponse est correcte, Benito gagne et le match décisif aura lieu à Cuzco. Si Benito se trompe, Alicia gagne et le match aura lieu à Lima. Une fois que Benito a fait part de sa réponse à Alicia, celle-ci révèle son entrée secrète pour que Benito vérifie s'il produit bien la sortie annoncée.

1. Divisez la classe en petits groupes, donnez à chacun d'eux le circuit et quelques jetons. La mise en situation sera sans doute plus réussie si on adapte l'histoire au contexte local. Établissez une convention pour les jetons colorés (rouge = 0, bleu = 1, par exemple). Pour qu'ils ne l'oublient pas, demandez-leur de la noter dans la légende encadrée en haut de la fiche d'activité.
2. Montrez aux élèves comment placer les jetons sur les entrées pour représenter les chiffres choisis par Alicia, puis expliquez les règles des portes logiques ET et OU résumées en bas de la fiche d'activité (on peut demander aux élèves de colorier ces exemples).
3. Montrez comment remplir le circuit en plaçant des jetons sur les « nœuds » et en déduisant les sorties correspondantes. Ce travail demande de la rigueur et de la précision. Le tableau suivant (qu'on ne doit pas donner aux élèves) vous donne la sortie correspondant à chaque entrée possible. Vous pourrez vous y reporter en cas de doute.

Entrée	000000	000001	000010	000011	000100	000101	000110	000111
Sortie	000000	010010	000000	010010	010010	010010	010010	010010
Entrée	001000	001001	001010	001011	001100	001101	001110	001111
Sortie	001010	011010	001010	011010	011010	011010	011010	011111
Entrée	010000	010001	010010	010011	010100	010101	010110	010111
Sortie	001000	011010	001010	011010	011010	011010	011010	011111
Entrée	011000	011001	011010	011011	011100	011101	011110	011111
Sortie	001010	011010	001010	011010	011010	011010	011010	011111
Entrée	100000	100001	100010	100011	100100	100101	100110	100111
Sortie	000000	010010	011000	011010	010010	010010	011010	011010
Entrée	101000	101001	101010	101011	101100	101101	101110	101111
Sortie	001010	011010	011010	011010	011010	011010	011010	011111
Entrée	110000	110001	110010	110011	110100	110101	110110	110111
Sortie	001000	011010	011010	011010	011010	111010	011010	111111
Entrée	111000	111001	111010	111011	111100	111101	111110	111111
Sortie	001010	011010	011010	011010	011010	111010	011010	111111

4. Chaque groupe élit une Alicia et un Benito. Le groupe peut être séparé en deux camps, celui d'Alicia et celui de Benito. Alicia choisit une entrée aléatoire pour le circuit, calcule la sortie et en fait part à Benito. Celui-ci doit alors deviner la parité de l'entrée (s'il s'agit d'un nombre pair ou impair de uns). À ce stade, il devient évident que le choix de Benito est pratiquement aléatoire. Alicia révèle ensuite son entrée à tout le monde, et Benito gagne s'il a donné la bonne parité. Il peut s'assurer qu'Alicia n'a pas menti sur son entrée initiale en vérifiant que celle-ci donne bien la sortie annoncée.

Le jeu de pile ou face est à présent terminé.

Benito peut tricher s'il est capable de trouver l'entrée initiale à partir d'une sortie donnée. Pour l'en empêcher, Alicia a donc intérêt à s'assurer que la fonction du circuit est bien à *sens unique* (voir l'Activité 14). Une fonction à sens unique est une fonction dont la sortie est facile à calculer si l'on connaît l'entrée, mais dont l'entrée pour une sortie donnée est très difficile à calculer.

Alicia peut tricher si elle parvient à trouver deux entrées de parité opposée qui produisent la même sortie. Dans ce cas, elle pourra choisir de révéler l'une ou l'autre selon la réponse de Benito. Ce dernier a donc intérêt à s'assurer que le circuit ne donne pas la même sortie pour de nombreuses entrées différentes.

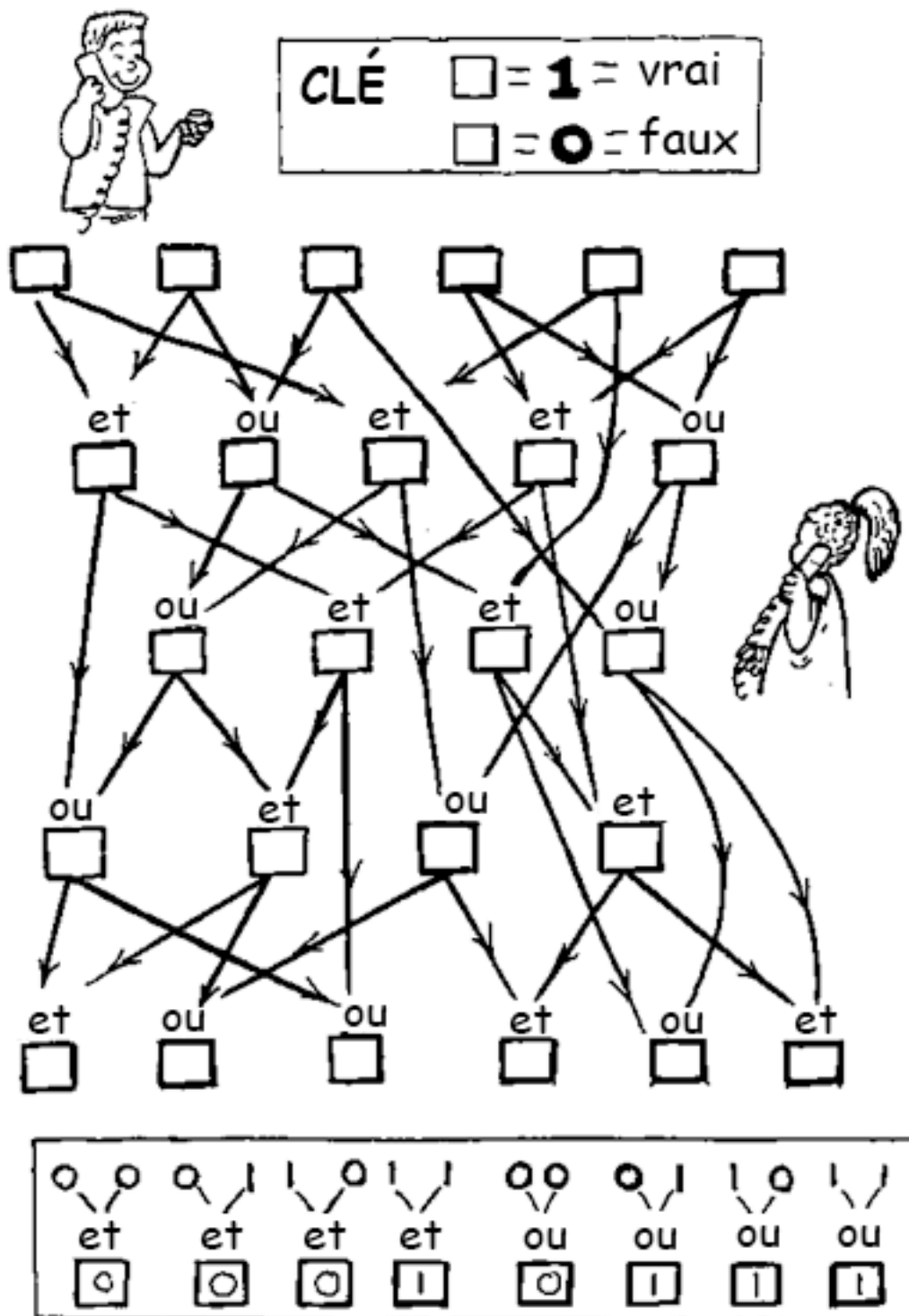
5. Voyez si les élèves parviennent à trouver un moyen de tricher pour Alicia ou Benito. Sur la première ligne du tableau, on voit que plusieurs entrées différentes génèrent la sortie 010010 (000001, 000011 ou 000101, par exemple). Si Alicia annonce la sortie 010010, elle peut donc choisir l'entrée 000001 si Benito dit que l'entrée est paire, et 000011 s'il dit que l'entrée est impaire.

Benito aura du mal à tricher avec ce circuit. Mais si la sortie est 011000, alors l'entrée est nécessairement 100010 – il n'y pas d'autre possibilité (vous pouvez vérifier sur le tableau). Si Alicia annonce cette sortie, Benito peut donc savoir que l'entrée est paire. Cela dit, un système informatique utiliserait beaucoup plus de bits. Il y aurait donc bien trop de possibilités à tester pour que Benito puisse tricher (chaque bit supplémentaire double le nombre de possibilités).

6. Demandez maintenant aux groupes d'inventer leurs propres circuits. Voyez s'ils trouvent un circuit qui permette à Alicia de tricher, et un autre qui permette à Benito de tricher. Rien ne les oblige à concevoir un circuit à six entrées, et le nombre d'entrées n'est pas forcément égal à celui des sorties.

Fiche d'activité : le pile ou face péruvien

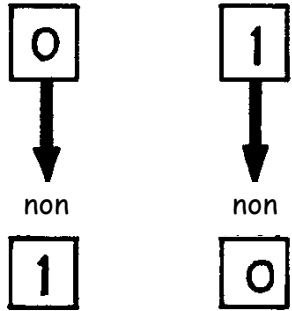
Choisis les entrées de ce circuit, puis trouve les sorties correspondantes



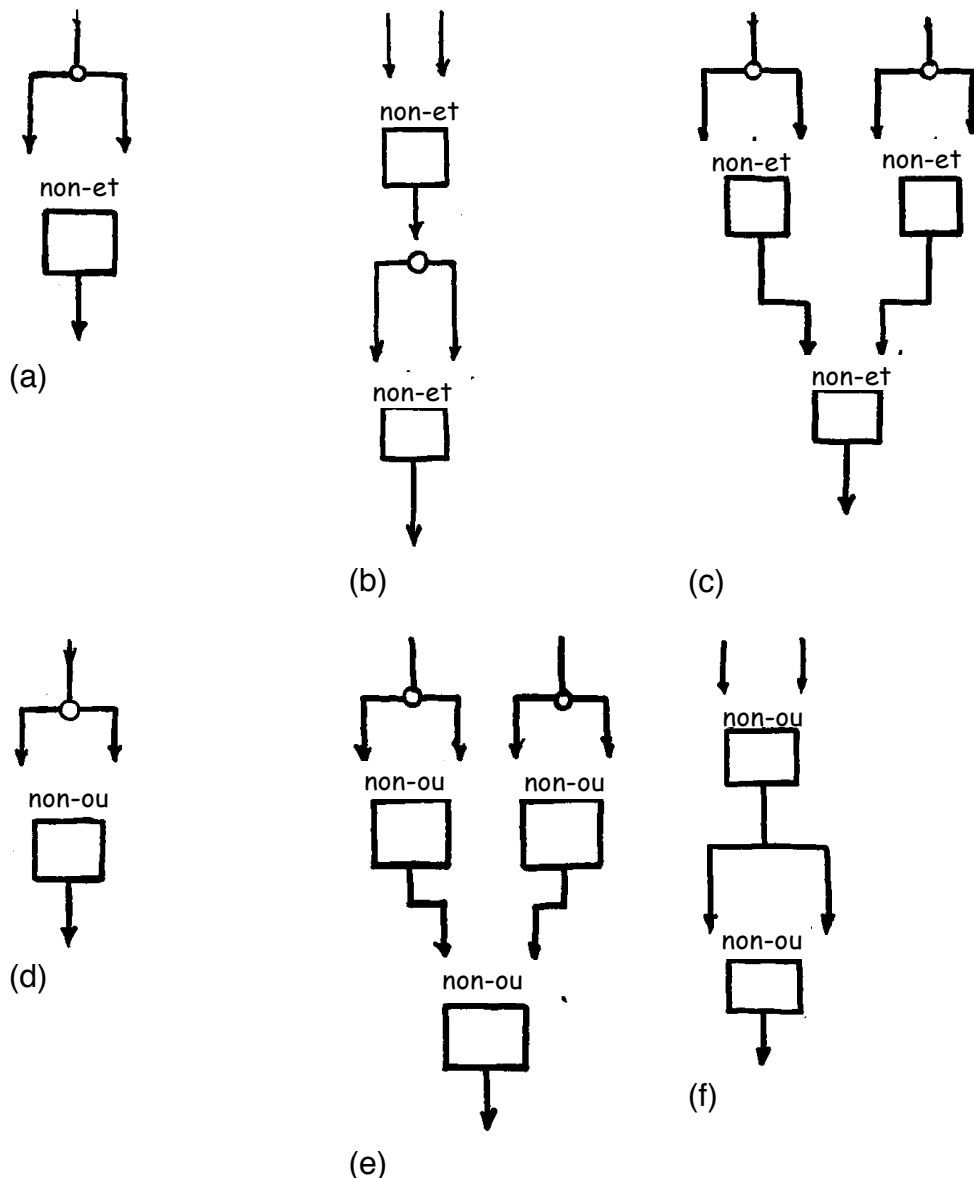
Variantes et activités complémentaires

1. Dans la pratique, le problème réside bien sûr dans la coopération nécessaire à la construction d'un circuit qui satisfasse à la fois Alicia et Benito. Cela peut être amusant pour les enfants, mais risque de rendre la procédure inopérante en pratique, surtout par téléphone ! Il existe néanmoins une alternative simple où Alicia et Benito construisent leurs circuits chacun de leur côté avant de les rendre publics. Alicia entre ensuite ses valeurs secrètes dans *les deux* circuits, puis réunit les deux sorties en les comparant bit à bit : le bit final est *un* si les bits sont identiques en sortie des deux circuits, et *zéro* dans le cas contraire. Ainsi, aucun participant ne peut tricher si l'autre ne triche pas, car si seulement un circuit est une fonction à sens unique, alors la combinaison des deux est également une fonction à sens unique.

Les deux variantes qui suivent n'ont pas directement trait aux protocoles cryptographiques ni au problème du pile ou face, mais plutôt aux circuits construits à partir de portes logiques ET et OU. Elles permettent d'explorer quelques concepts fondamentaux propres aux circuits informatiques, mais aussi à la logique. Ce type de logique est appelé algèbre booléenne en référence au mathématicien George Boole (1815-1864).

2. Les élèves auront peut-être remarqué qu'une entrée exclusivement composée de *zéros*, 000000, donnera forcément une sortie exclusivement composée de *zéros* et que, de la même manière, une entrée exclusivement composée de *uns* donnera une sortie exclusivement composée de *uns*. (D'autres entrées produisent les mêmes résultats ; ainsi, dans le circuit donné en exemple, 000010 ne produit que des *zéros*, et 110111 ne produit que des *uns*.) Cela est dû au fait que les circuits sont construits à l'aide de portes ET et OU. En ajoutant une porte NON, laquelle ne prend qu'une entrée et donne l'inverse en sortie (par exemple, $0 \rightarrow 1$ et $1 \rightarrow 0$), les élèves pourront concevoir des circuits qui n'ont pas cette propriété.
- 
3. Il existe deux autres types de portes importants, les portes NON-ET et NON-OU, semblables aux portes ET et OU mais suivies d'un NON. Ces portes ne permettent pas la création de circuits fonctionnellement différents, car leur effet peut toujours être obtenu à l'aide de portes ET et OU, suivies de NON. Cela dit, elles présentent une propriété intéressante : tous les autres types de portes peuvent être construits à partir de portes NON-ET ou à partir de portes NON-OU.

Après avoir présenté les portes NON-ET et NON-OU, mettez les élèves au défi de découvrir si on peut construire n'importe quel type de porte à partir d'autres portes reliées entre elles, puis si on peut les construire à partir d'un seul type de porte. L'illustration suivante montre comment construire les trois portes logiques de base (NON, ET et OU) à partir de portes NON-ET (rangée du haut) et de portes NON-OU (rangée du bas).



Ce qu'il faut retenir

Ces dernières années ont été marquées par une hausse sans précédent du commerce en ligne qui transite par des réseaux informatiques. Il est donc essentiel de garantir la sécurité des échanges électroniques de fonds, d'opérations confidentielles et de documents signés ayant valeur légale. L'objectif de la *cryptographie* est précisément de communiquer de façon sûre et confidentielle. Il y a plusieurs décennies, les chercheurs en informatique ont découvert que la confidentialité pouvait paradoxalement être garantie par des techniques où certaines informations sont rendues *publiques*. Cela a donné naissance au « système de cryptographie à clé publique » de l'Activité 18 (Les cryptographes en herbe). Largement utilisé de nos jours, il constitue le moyen le plus courant d'échanger des informations de façon sécurisée. Peut-être avez-vous déjà vu, dans les paramètres de votre navigateur, des abréviations telles que SSL (Secure Sockets Layer) ou TLS (Transport Layer Security). Ces protocoles sont fondés sur des systèmes à clé publique qui permettent au navigateur d'établir une connexion sécurisée à certains sites Internet (notamment ceux des banques) même si un mouchard surveille les données transférées.



La cryptographie n'a pas seulement pour objet de garder les secrets. Elle permet aussi de mettre en place un contrôle de l'information qui limite les données accessibles aux autres et d'instaurer une relation de confiance entre des individus séparés géographiquement. Les règles formelles, ou « protocoles », qui régissent les opérations codées permettent d'accomplir des choses apparemment impossibles, notamment des signatures numériques infalsifiables ou le fait de dire aux autres que l'on a un secret (un mot de passe, par exemple) sans toutefois le révéler. Jouer à pile ou face par téléphone est un problème plus simple, mais analogue, qui paraît lui aussi insoluble au premier abord.

Dans la vie réelle, Alicia et Benito n'inventeraient pas le circuit eux-mêmes : ils se procureraient un logiciel qui ferait le travail à leur place. Sans doute qu'ils ne s'intéresseraient pas à son fonctionnement interne, mais tous deux voudraient pouvoir être sûrs que l'autre est incapable d'influer sur le résultat final, quels que soient ses compétences en informatique et les efforts qu'il est prêt à fournir.

En principe, tout conflit devrait être résolu en faisant appel à un juge neutre. On lui donnerait le circuit, les chiffres binaires qu'Alicia a entrés dans le circuit, le résultat envoyé à Benito et enfin la supposition que ce dernier lui a renvoyée. Une fois l'échange terminé, toutes ces informations auraient un caractère public, et les deux participants seraient donc bien forcés d'admettre que c'est sur elles que le résultat final est fondé. Inutile de préciser que la simple mise en place d'une procédure claire pour empêcher la violation des règles rend tout conflit peu probable. Comparez avec la situation où Benito choisirait pile ou face et Alicia lancerait une vraie pièce : aucun juge n'accepterait de traiter cette affaire !

Un circuit aussi petit que celui de notre exemple ne serait pas très utile en pratique, car il est facile d'établir un tableau des correspondances qui permette de tricher. Une entrée de 32 bits offrirait déjà une meilleure protection. Cependant, cela ne suffirait pas à *garantir* qu'il est difficile de tricher, car tout dépend de la manière dont le circuit est conçu. On pourrait également employer d'autres méthodes, telles que la fonction à sens unique de Touristeville (Activité 14). Dans la pratique, on utilise souvent des méthodes fondées sur la factorisation de grands nombres, laquelle est reconnue comme étant un problème difficile (bien qu'il ne soit pas NP-complet, comme nous allons le voir à la fin de la prochaine activité). On peut aisément vérifier si un nombre est facteur d'un autre, mais trouver les facteurs d'un grand nombre demande énormément de temps. Cela complique donc le travail pour Alicia, Benito et le juge, même si, comme nous l'avons vu plus haut, la tâche sera en fait accomplie par un logiciel conçu à cet effet.

Les signatures numériques sont fondées sur une idée similaire. En rendant publique la sortie du circuit correspondant à l'entrée secrète qu'elle a choisie, Alicia est effectivement capable de prouver que c'est elle qui a généré la sortie (car, avec une véritable fonction à sens unique, personne d'autre ne peut trouver l'entrée correspondante). Personne ne peut donc se faire passer pour Alicia ! Cela dit, si l'on veut réaliser une véritable signature numérique, il faudra un protocole un peu plus complexe, le but étant qu'Alicia puisse signer un message et que les autres puissent vérifier qu'Alicia est bien l'auteur de la signature même si elle prétend ne pas l'être. Quoi qu'il en soit, le principe est similaire.



Une autre application possible consiste à jouer au poker par téléphone, dans un environnement où il n'y a pas d'arbitre pour distribuer les cartes et vérifier les mains des deux joueurs. Ces derniers sont livrés à eux-mêmes et ne peuvent avoir recours à un juge qu'à la fin du jeu, en cas d'éventuel désaccord. Des situations similaires se présentent lorsque l'on négocie des contrats. Les joueurs ne doivent évidemment pas révéler leur jeu pendant la partie, mais il faut néanmoins s'assurer de leur honnêteté : ils ne doivent pas pouvoir prétendre avoir un as s'ils n'en ont pas ! Il est possible de vérifier tout cela en attendant la fin de la partie pour que chaque joueur inspecte la main initiale de son adversaire et celle d'une éventuelle seconde distribution. Un autre problème consiste à distribuer les cartes sans connaître la main de l'adversaire avant la fin de la partie. Aussi étonnant que cela puisse paraître, c'est tout à fait possible grâce à un protocole cryptographique qui n'est pas sans rappeler celui du pile ou face.

Les protocoles cryptographiques sont extrêmement importants dans les transactions électroniques, que ce soit pour identifier le propriétaire d'une carte bancaire, autoriser l'utilisation d'un téléphone portable ou authentifier l'auteur d'un mail. Le succès du commerce électronique dépend de notre capacité à effectuer ces tâches en toute sécurité.

Suggestions de lecture (en anglais)

L'ouvrage de Harel, *Algorithmics*, s'intéresse aux signatures numériques et aux protocoles cryptographiques associés. Il explique également comment jouer au poker par téléphone, problème soulevé pour la première fois en 1981 dans le chapitre intitulé « Mental poker » du livre *The Mathematical Gardener* de D.A. Klarner. *Cryptography and data security*, de Dorothy Denning, offre une excellente approche informatique de la cryptographie. Enfin, l'ouvrage de Dewdney, *Turing Omnibus*, comporte une section sur la logique booléenne qui traite des portes logiques utilisées dans les circuits de cette activité.

Activité 18

Les cryptographes en herbe – *Chiffrement à clé publique*

Résumé

Le chiffrement est la clé de la sécurité de l'information. Et la clé du chiffrement moderne réside dans le fait qu'en utilisant uniquement des informations publiques, un expéditeur peut cadenasser son message de sorte qu'il ne puisse être ouvert (en privé, bien sûr) que par le destinataire prévu.

C'est un peu comme si tout le monde achetait un cadenas, écrivait son nom dessus, et le posait sur une table commune pour que les autres puissent s'en servir. Bien sûr, chacun garde sa clé, car c'est le genre de cadenas qui se referme d'un coup, sans clé. Si je veux vous transmettre un message sécurisé, je le mets dans une boîte, je prends votre cadenas, je verrouille la boîte et je vous l'envoie. Même si le colis tombe en de mauvaises mains, personne d'autre que vous ne pourra l'ouvrir. Avec ce système, il est inutile de s'accorder au préalable sur un code secret.

Cette activité montre comment le faire de façon numérique. Dans le monde numérique, je n'utilise pas votre cadenas mais la copie que j'en aurai faite, je laisse donc l'original sur la table. Si je devais faire un double d'un vrai cadenas, il faudrait que je le démonte et je verrais donc inévitablement comment il fonctionne. Dans le monde numérique, heureusement, on peut faire en sorte que les gens copient les « cadenas » sans être en mesure de découvrir la clé ! Cela vous paraît impossible ? Lisez ce qui suit.

Liens pédagogiques

- ✓ Technologie : cryptographie à clé publique, codes secrets.

Compétences

- ✓ Résolution d'énigmes

Âge

- ✓ 11 ans et plus

Matériel

La classe est répartie en groupes d'environ quatre élèves, eux-mêmes divisés en deux sous-groupes. Chaque sous-groupe reçoit une copie des deux cartes de la fiche d'activité *Les fonctions des cryptographes en herbe*. Pour chaque groupe d'élèves, il faudra donc :

- ✓ Deux copies de la fiche d'activité *Les fonctions des cryptographes en herbe*

Pour l'enseignant :

- ✓ un rétroprojecteur et un transparent de la fiche d'activité *L'encodage des cryptographes en herbe*
- ✓ Un feutre pour annoter le schéma



Les cryptographes en herbe

Introduction

De toutes les activités du livre, c'est techniquement la plus difficile. Elle est enrichissante mais demande un travail soigneux et une concentration prolongée. Les élèves devraient déjà avoir étudié l'exemple de fonction à sens unique de l'Activité 14 (Touristeville), et il leur sera utile de réaliser au préalable les autres activités de cette partie (Activité 16, Partager des secrets, et Activité 17, Le pile ou face péruvien). L'activité a par ailleurs recours à des notions traitées dans l'Activité 1 (Compter les points) et l'Activité 5 (Vingt devinettes).

Amy a l'intention d'envoyer un message secret à Bill. On serait tenté de penser qu'il s'agit d'une phrase ou d'un paragraphe, mais dans l'exercice suivant Amy n'enverra qu'un seul caractère, ou plus précisément un nombre représentant un caractère. Cela peut paraître simpliste, mais il ne faut pas oublier qu'en pratique on confie la tâche à un ordinateur, et que de tels messages pourront donc être envoyés en série de façon à former une phrase. Par ailleurs, les petits messages sont parfois de la plus haute importance – l'un des plus célèbres de l'histoire américaine étant celui de Paul Revere pendant la Révolution, qui ne comportait que deux valeurs possibles (« une lanterne au clocher si l'ennemi arrive par voie de terre, deux s'il arrive par voie de mer »). Nous allons voir comment introduire le nombre d'Amy dans un message codé en utilisant le cadenas public de Bill, de sorte que tout individu interceptant le message soit incapable de le décoder. Seul Bill sera en mesure de le faire parce que lui seul possède la clé du cadenas.

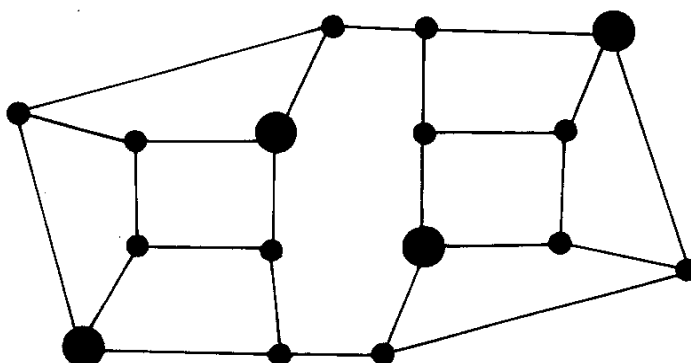
Nous allons verrouiller les messages à l'aide de fonctions similaires à la carte de Touristville (Activité 14) où les lignes correspondaient aux rues et les points aux intersections. Chaque fonction comprend une version publique (le cadenas) et une version privée (la clé).

Déroulement de la séance

Sur la fiche d'activité *L'encodage des cryptographes en herbe* figure une représentation de la fonction publique de Bill. Elle n'est pas confidentielle : Bill la met sur la table (ou sur une page Internet) à la

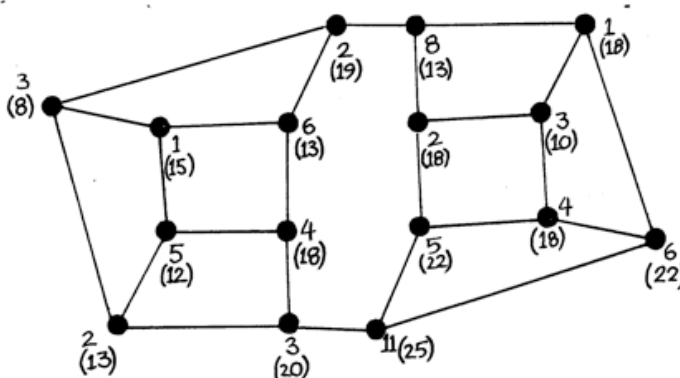
vue de tous, ou la donne à tous ceux qui pourraient avoir envie de lui envoyer un message (cela revient au même). Comme tous les autres, Amy dispose donc d'une copie.

L'illustration ci-dessus représente la fonction privée de Bill. Elle est semblable à sa fonction publique, à la seule différence près que certains nœuds sont marqués par un point plus gros que les autres. Il garde secrète cette version de la fonction.



Mieux vaut que toute la classe fasse l'activité ensemble, du moins pour commencer, car elle demande beaucoup de travail. La tâche n'est pas difficile mais requiert de la précision, la moindre erreur étant source de nombreux problèmes. Les élèves doivent bien se rendre compte à quel point il est étonnant qu'un tel encodage soit possible, car il leur faudra beaucoup de motivation pour venir à bout du problème. L'un des meilleurs moyens de motiver les élèves est de leur faire comprendre qu'ils pourraient ainsi se faire passer des messages secrets en classe et que l'enseignant, même s'il savait comment la note a été encodée, serait incapable de la décoder.

1. Montrez la fonction publique de Bill (fiche d'activité *L'encodage des cryptographes en herbe*). Décidez du nombre qu'Amy va envoyer, puis assignez un nombre aléatoire à chaque nœud de sorte que leur



somme soit égale au nombre qu'Amy veut envoyer. L'illustration ci-dessus vous donne un exemple : les nombres du haut (sans parenthèses) correspondent aux nombres aléatoires. Ici, Amy a décidé d'envoyer le nombre 66, le total de tous les nombres sans parenthèses est donc égal à 66. Si nécessaire, vous pouvez utiliser des nombres négatifs pour abaisser la somme au nombre voulu.

2. Amy doit maintenant calculer ce qu'elle va envoyer à Bill. Ce ne serait pas une bonne idée d'envoyer la fonction où figurent les nombres car, si celle-ci tombait en de

mauvaises mains, n'importe qui pourrait les additionner, donc comprendre le message.

Choisissez n'importe quel nœud, voyez le nombre qui y figure et additionnez-le avec ceux des trois autres nœuds les plus proches (cela fait quatre nœuds en tout). Notez le total sur le nœud en question, entre parenthèses ou avec un feutre d'une autre couleur. Par exemple, le nœud à l'extrême droite de la clé publique (noté 6) est relié à trois autres nœuds (notés 1, 4 et 11), ce qui donne donc un total de 22. Répétez l'opération pour tous les nœuds de la fonction. Vous aurez alors les nombres entre parenthèses.

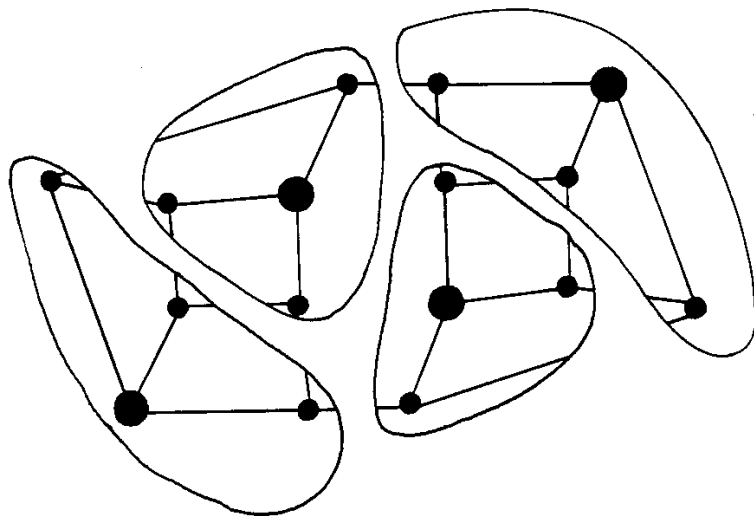
3. Amy va envoyer à Bill sa fonction publique en n'indiquant que les nombres entre parenthèses.

Effacez les nombres du début et les additions, ou recopiez le schéma en n'indiquant que les nombres entre parenthèses. Demandez aux élèves comment deviner le message original à partir des données encore présentes sur le schéma. Ils seront incapables de répondre à la question.

4. Seule une personne possédant la clé privée de Bill pourra décoder le message. Sur le message chiffré, marquez les nœuds secrets signalés par des gros points sur le schéma de la fonction privée de Bill.

Pour décoder le message, Bill devra simplement additionner les nombres figurant sur les nœuds marqués. Dans cet exemple, on a : 13, 13, 22 et 18, ce qui donne un total de 66. C'est le message secret d'Amy.

5. Vous vous demandez comment ça marche ? En fait, la fonction a quelque chose de spécial. Supposons que Bill choisisse l'un des nœuds marqués et trace autour un cercle englobant les nœuds distants d'une « rue », puis



qu'il répète l'opération pour tous les autres nœuds marqués. Le schéma de la fonction serait alors divisé en différentes parties qui ne se chevaucheraient pas, comme dans l'illustration ci-dessus. Montrez ces parties aux élèves en traçant leurs frontières sur le schéma. Les nœuds composant chaque partie sont précisément ceux qu'on a additionnés pour obtenir les nombres transmis (entre parenthèses) correspondant aux nœuds marqués. La somme des quatre nombres transmis (entre parenthèses) assignés à ces nœuds secrets est donc égale à la somme de tous les nombres initialement choisis : c'est bien le message secret !

Cela fait beaucoup de travail pour une simple lettre ! L'encodage n'est pas quelque chose de facile, cela ne fait aucun doute. Mais voyez un peu ce qui a été accompli : nous sommes parvenus à une confidentialité totale tout en utilisant une clé publique, sans avoir besoin d'un accord préalable entre l'auteur et le destinataire. Si vous affichez votre clé à la vue de tous, *n'importe qui* pourra vous envoyer des messages secrets mais *personne* ne pourra les décrypter sans la clé privée. Et puis, dans la vie réelle, tous ces calculs sont effectués par un logiciel (généralement intégré au navigateur web), l'ordinateur travaille donc à votre place.

Les élèves seront peut-être contents d'apprendre qu'ils ont rejoint le groupe très fermé des gens ayant déjà étudié sans ordinateur un cas de chiffrement à clé publique – un informaticien pourrait penser que c'est quasiment impossible, et très peu de gens l'ont déjà fait !

Bien, mais qu'en est-il des espions ? La fonction de Bill est comme la carte de Touristeville (Activité 14), les nœuds marqués correspondant à la solution minimale pour placer les camions à glaces de sorte que personne n'ait à passer un carrefour pour s'acheter une glace. Nous avons vu qu'il était facile de concevoir une telle carte (Bill peut en créer une à partir des fragments de sa fonction privée), mais très difficile pour les autres de trouver la solution minimale parce que seule la méthode exhaustive fonctionnait. Cette dernière consiste à tester toutes les configurations possibles avec un camion, puis avec deux camions, et ainsi de suite jusqu'à trouver une solution. Nul ne sait s'il existe une meilleure méthode – et vous pouvez être certain que beaucoup de personnes ont déjà essayé de la trouver !

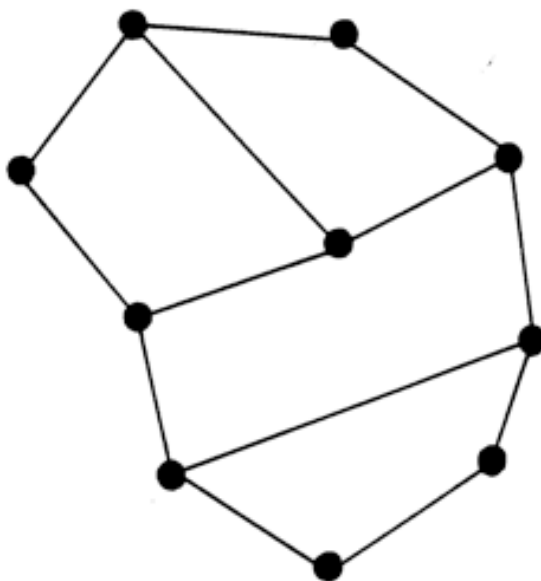
Si Bill possède une fonction suffisamment compliquée, avec par exemple 50 ou 100 nœuds, il semble que personne ne pourra jamais décrypter son code – même les plus brillants mathématiciens ont essayé et échoué. (Il faut cependant prendre certaines précautions, voir ci-dessous dans la section « Ce qu'il faut retenir ».)

6. Après avoir expliqué l'exemple à la classe entière, divisez-la en petits groupes de quatre, par exemple, puis divisez à nouveau chaque groupe en deux. Chaque groupe de deux reçoit la fonction publique de la fiche d'activité *Les fonctions des cryptographes en herbe*, choisit un « message » (n'importe quel nombre entier), l'encode à l'aide de la clé publique, puis donne le schéma qui en résulte au groupe partenaire. Les élèves pourront toujours essayer de le décoder, ils n'y parviendront pas tant qu'ils n'auront pas reçu (ou découvert !) la fonction privée. Distribuez ensuite le schéma de la fonction privée pour que les élèves décodent le message.
7. À présent, chaque groupe de deux peut créer sa propre fonction. Ils gardent secrète la version privée et donnent la version publique au groupe partenaire (ou ils la « publient » sur le tableau de la classe). Le principe de base pour créer ces fonctions est le même que pour la carte de Touristeville, on peut là aussi ajouter des « rues » pour dissimuler la solution. Il faut simplement prendre garde à ne pas en ajouter entre les points « secrets ». Cela produirait une intersection où *deux* camions à glaces seraient accessibles en une étape, ce qui n'est pas un problème pour Touristeville mais perturberait gravement le chiffrement : les points « spéciaux » ne décomposeraient plus la fonction en fragments *qui ne se chevauchent pas* (comme ceux qui apparaissent sur la fonction privée), alors que c'est un détail essentiel au bon fonctionnement de notre système.

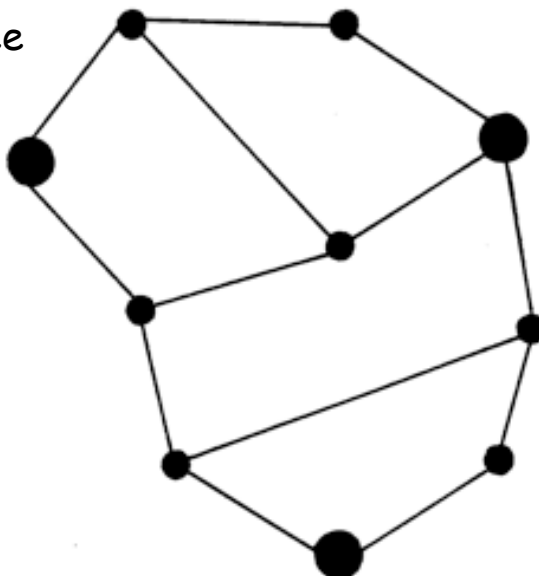
Fiche d'activité : les fonctions des cryptographes en herbe

Utilisez ces fonctions pour encoder et décoder des messages (comme expliqué dans le texte).

Fonction publique

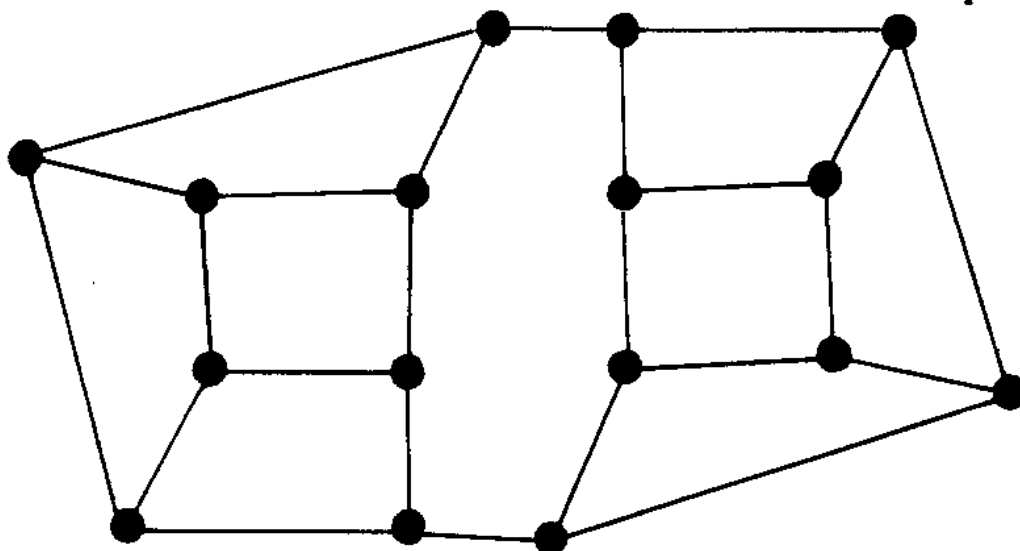


Fonction privée



Fiche d'activité : l'encodage des cryptographes en herbe

Montrez cette fonction à la classe et utilisez-la pour expliquer comment encoder un message.



Ce qu'il faut retenir

On comprend aisément ce qui peut donner envie d'envoyer sur un réseau informatique des messages secrets que seul le destinataire pourra décoder, et ce quels que soient l'intelligence et le mal que se donnent ceux qui essaient de le déchiffrer. Si l'auteur du message et son destinataire partagent un code secret, il existe bien sûr toutes sortes de procédés possibles. Mais le point fort du chiffrement à clé publique tient au fait qu'Amy peut envoyer un message sécurisé à Bill sans arrangement préalable, en prenant simplement son « cadenas » dans un espace public tel qu'une page web.

La confidentialité ne représente qu'un versant de la cryptographie, l'autre étant l'*authentification*. Supposons qu'Amy reçoive un message de Bill, comment peut-elle être sûre que c'est bien lui qui l'a envoyé et pas un imposteur ? Disons qu'elle reçoit un courrier électronique où est écrit, « Chérie, je suis coincé ici sans argent. S'il te plaît mets 100 euros sur mon compte bancaire, numéro 0241-45-784329. Je t'embrasse, Bill. » Comment peut-elle savoir si le message vient vraiment de Bill ? Certains systèmes de chiffrement à clé publique sont adaptés à ce type de problème. De même qu'Amy peut envoyer un message secret à Bill en utilisant la clé publique de Bill, celui-ci peut envoyer un message dont *lui seul peut être l'auteur* en l'encodant avec sa propre clé *privée*. Si Amy peut le décoder grâce à la clé publique de Bill, alors cela signifie que le message vient bien de lui. Évidemment, tout le monde peut également le décoder puisque la clé est publique, mais si le message ne doit être vu que par Amy, Bill peut l'encoder une seconde fois en utilisant la clé publique d'Amy. Ce double encodage assure à la fois la confidentialité et l'authentification, avec le même système de base composé d'une clé publique et d'une clé privée.

À présent, il est temps d'admettre que si le système de chiffrement à clé publique présenté dans cette activité est extrêmement puissant, il n'est en fait pas sécurisé – même en utilisant une grande carte.

Cela s'explique par le fait que, même si l'on ne connaît aucun moyen de trouver une solution minimale pour le placement des camions à glaces sur une carte donnée, et que le système est donc sûr de ce point de vue, il existe en fait une manière complètement différente de l'attaquer. Il est très peu probable que les élèves y pensent, surtout ceux du primaire et du collège, mais il faut quand même savoir qu'elle existe. Disons que le système étudié dans cette partie est sûr pour les écoliers, mais pas pour les mathématiciens. Vous pouvez donc sauter le paragraphe suivant si vous n'êtes pas matheux !

Numérotons les nœuds de la carte : 1, 2, 3, etc. Désignons les nombres initialement assignés aux nœuds par b_1, b_2, b_3 , etc., et les nombres finalement transmis par t_1, t_2, t_3 , etc. Supposons que le nœud 1 soit connecté aux nœuds 2, 3 et 4. Le nombre transmis pour ce nœud est :

$$t_1 = b_1 + b_2 + b_3 + b_4$$

Évidemment, il existe une équation similaire pour tous les nœuds de la carte : il y a autant d'équations qu'il y a d'inconnues (b_1, b_2, b_3 , etc.). L'espion connaissant la carte publique ainsi que les nombres transmis (t_1, t_2, t_3 , etc.), il peut écrire les équations et les résoudre à l'aide d'un logiciel dédié. Une fois les nombres initiaux obtenus, il lui suffira de les additionner pour obtenir le message. Il est donc inutile de découvrir la fonction privée. En utilisant la méthode de l'élimination de Gauss, l'effort de calcul nécessaire à la résolution des équations est proportionnel au cube du nombre d'équations, mais comme il s'agit d'un système d'équations creux (la plupart des coefficients sont égaux à zéro), il existe

des techniques encore plus efficaces. C'est sans comparaison avec l'effort de calcul exponentiel qui est, pour autant que l'on sache, la meilleure méthode pour trouver la fonction de déchiffrement.

Nous espérons que vous ne vous sentez pas trahis ! En fait, les méthodes employées dans les systèmes de chiffrement à clé publique sont similaires à celles que nous avons étudiées, sauf que les techniques d'encodage sont différentes – et vraiment impossibles à mettre en œuvre sans l'aide d'un ordinateur. La première méthode à clé publique, qui est encore l'une des plus sûres, repose sur la difficulté de factorisation des grands nombres.

Quels sont les facteurs du nombre de 100 chiffres : 9 412 343 607 359 262 946 971 172 136 294 514 357 528 981 378 983 082 541 347 532 211 942 640 121 301 590 698 634 089 611 468 911 681 ? N'y perdez pas tout votre temps !

Il existe seulement deux facteurs :

86 759 222 313 428 390 812 218 077 095 850 708 048 977 et 108 488 104 853 637 470 612 961 399 842 972 948 409 834 611 525 790 577 216 753. Ces deux nombres sont premiers. Le travail à accomplir pour les trouver est titanesque : même pour un superordinateur, cela représente un projet de plusieurs mois.

Dans un vrai système de chiffrement à clé publique, Bill pourrait donc utiliser ce nombre de 100 chiffres comme clé publique et les deux facteurs comme clé privée. Mettre au point ce type de clé n'est pas trop difficile : il suffit d'avoir les moyens de calculer les grands nombres premiers. Trouvez deux nombres premiers suffisamment grands (ce n'est pas trop compliqué), calculez leur produit, et le tour est joué : vous avez votre clé publique. Multiplier de grands nombres n'est pas difficile pour un ordinateur. Avec cette clé publique, personne ne pourra trouver votre clé privée à moins d'avoir un superordinateur et plusieurs mois devant soi. Si vous craignez que ce soit le cas, utilisez des nombres premiers à 200 chiffres et ils en auront pour des années ! Le principal, c'est que les frais de déchiffrement de la clé soient supérieurs à la valeur de l'information qui serait dévoilée. En pratique, on assure généralement la sécurité des connexions à l'aide de clés de 512 bits ou plus, ce qui correspond à des nombres composés de 155 chiffres environ.

Nous n'avons toujours pas expliqué comment chiffrer un message avec une clé publique fondée sur les nombres premiers de sorte qu'il soit impossible de le déchiffrer sans la clé privée. C'est nettement plus compliqué que ce que nous avons vu. Dans ce cas, la clé privée ne sera pas composée des deux nombres premiers et la clé publique ne sera pas leur produit : nous utiliserons plutôt des dérivés de ces nombres. Le principe de base reste toutefois le même, et il est donc possible de déchiffrer le code en factorisant les nombres. Cela dit, on peut aisément surmonter ces difficultés afin d'aboutir à un véritable algorithme de chiffrement et de déchiffrement, mais nous n'entrerons pas dans les détails ici. Cette activité nous a déjà demandé assez de travail !

Quel niveau de sécurité offre un système fondé sur les nombres premiers ? La factorisation des grands nombres est un problème auquel les plus brillants mathématiciens s'intéressent depuis des siècles. Si l'on a découvert des méthodes sensiblement meilleures que l'algorithme exhaustif (qui consiste à tester tous les facteurs possibles), personne n'a trouvé d'algorithme vraiment rapide (c'est-à-dire un algorithme en temps polynomial). (Notons cependant qu'on n'a pas non plus démontré qu'un tel algorithme ne pouvait pas exister.) Il semble ainsi que le système soit sécurisé non seulement pour les écoliers, mais aussi pour les mathématiciens. Cependant, il faut tout de même se montrer prudent : de même qu'il y avait en fait un moyen de décrypter le code de Bill sans résoudre le

problème de Touristeville, il pourrait y avoir un moyen de décrypter les codes à nombres premiers sans résoudre le problème de la factorisation des grands nombres. Cela dit, des vérifications soigneuses ont été effectuées et, jusqu'ici, le système semble sûr.

Autre inquiétude : s'il n'y qu'une poignée de messages possibles, un intrus pourrait encoder chacun d'eux à l'aide de la clé publique, puis comparer le vrai message avec toutes les possibilités. La méthode d'Amy empêche ce type de procédé parce qu'il existe de nombreuses façons d'encoder le même message, qui varient en fonction des nombres choisis au début (dont la somme fournit le message). Dans la pratique, les systèmes de cryptographie à clé publique sont conçus de sorte qu'il y ait trop de messages possibles pour envisager de tous les tester, même avec l'aide d'un ordinateur très puissant.

Nous ne savons pas s'il existe une méthode pour résoudre rapidement le problème de la factorisation entière en nombres premiers. Personne ne l'a trouvée, mais personne n'a su prouver qu'elle n'existait pas. Si on inventait un algorithme pour ce problème, bon nombre des systèmes cryptographiques utilisés de nos jours perdraient leur efficacité. La Partie IV traitait des problèmes NP-complets, lesquels sont de la même étoffe, c'est-à-dire que si l'un d'eux s'avère pouvoir être résolu en temps polynomial, tous les autres le seront aussi. Puisque tant d'efforts ont été déployés en vain pour trouver des algorithmes capables de les résoudre rapidement, il semblerait que ces problèmes soient parfaitement adaptés à la conception de systèmes de chiffrement entièrement sécurisés. Hélas, cette stratégie ne va pas sans difficultés, et nous devons pour l'instant nous contenter de problèmes comme la factorisation, qui sont sans doute plus faciles – voire nettement plus faciles – à résoudre que les problèmes NP-complets. Les réponses aux questions soulevées ici représentent des millions de dollars pour les entreprises et sont considérées comme essentielles à la sécurité nationale. La cryptographie constitue à l'heure actuelle un secteur très dynamique de la recherche en informatique.

Suggestions de lecture

L'ouvrage de Harel, *Algorithmics*, traite de la cryptographie à clé publique ; il explique comment utiliser de grands nombres premiers pour concevoir un système à clé publique sécurisé. L'ouvrage de Dorothy Denning, *Cryptography and data security*, est un classique de l'informatique ; celui de Bruce Schneier, *Applied cryptography* est moins théorique et plus pratique. Enfin, *Turing Omnibus*, de Dewdney, décrit un autre système de chiffrement à clé publique.

Partie VI

**Le visage humain de
l'informatique – *Interagir avec les
ordinateurs***

Le visage humain de l'informatique

Pourquoi est-il si difficile de s'entendre avec les ordinateurs ? Bon nombre de personnes ont des histoires à raconter à ce propos, elles vous diront combien ils sont difficiles à utiliser, qu'ils semblent ne jamais faire ce qu'on veut vraiment leur faire faire, qu'ils ont toujours des problèmes et commettent de grossières erreurs. On dirait parfois que les ordinateurs sont faits pour les sorciers et non pas pour monsieur et madame Tout le Monde. Pourtant, c'est bien pour les gens ordinaires qu'ils doivent être conçus, car les ordinateurs sont des outils du quotidien qui nous aident à mieux apprendre, mieux travailler et mieux jouer.

La partie du système informatique avec laquelle on dialogue est appelée « interface utilisateur ». C'est la plus importante. On pourrait penser que ce qui compte est la tâche effectivement accomplie et que l'interface utilisateur n'est que le moyen d'y accéder, mais un programme ne sert absolument à rien si on ne peut pas dialoguer avec lui pour lui faire faire ce qu'on veut. Il est très difficile de concevoir et de développer une interface utilisateur. Dans l'écriture d'un programme, c'est la partie qui demande de loin le plus d'efforts. Certains logiciels sont dotés d'excellentes interfaces utilisateurs, qui n'ont besoin d'aucune instruction compliquée et se font presque oublier avec un peu de pratique. À l'inverse, d'innombrables logiciels ont été des échecs complets pour la seule raison que leur interface était étrange. Des industries entières se sont bâties à partir d'une idée d'interface ingénieuse – telle que le traitement de texte ou le smartphone – favorisant l'accès à des fonctions informatiques qui sont en elles-mêmes extrêmement simples.

Mais pourquoi les interfaces utilisateurs sont-elles nécessaires ? Pourquoi ne peut-on pas simplement parler à notre ordinateur comme on parle à un ami ? C'est une bonne question. Peut-être que cela arrivera un jour, peut-être pas. Ce n'est en tout cas pas possible aujourd'hui parce que de nombreuses limitations pratiques viennent encore entraver l'« intelligence » des ordinateurs. Les activités suivantes abordent les problèmes liés à la conception des interfaces utilisateurs, elles vous permettront de mieux comprendre les limitations des ordinateurs et d'être plus vigilant face au battage publicitaire pour les produits informatiques.

Pour les enseignants

L'enjeu de l'informatique n'est pas tant le calcul que la *communication*. Le calcul en soi n'a pas de valeur intrinsèque : il ne vaut que si les résultats sont transmis au monde extérieur et y exercent une certaine influence. Dans ce livre, de nombreuses activités ont trait à la communication. La Partie I, *Représenter les informations*, montre comment différents types de données peuvent être communiqués à un ordinateur ou partagés entre eux. La Partie III, *Représentation de procédures*, explique comment transmettre à un ordinateur une méthode lui permettant d'accomplir certaines tâches (en fait, la programmation consiste simplement à donner des instructions à un ordinateur en utilisant son propre langage !). Enfin, la Partie V, *Cryptographie*, explique comment communiquer secrètement, ou plutôt comment communiquer une partie du secret sans tout révéler.

Les activités suivantes portent sur notre manière de communiquer avec les ordinateurs. Si le reste du livre repose sur des idées techniques bien comprises, ce n'est pas le cas de cette partie. Celle-ci est donc à la fois plus facile, dans la mesure où aucune connaissance spécifique n'est requise, et plus difficile, dans la mesure où une certaine maturité est nécessaire pour comprendre le sens des activités et les rattacher à un contexte plus large.

Si les explications sont plus détaillées qu'ailleurs, c'est parce que l'enseignant doit être assez renseigné sur la question pour être en mesure d'aider les élèves à prendre conscience de ses implications.

Il y a deux activités dans cette partie. La première porte sur un domaine appelé « interface homme-machine ». Pour aborder cet aspect de l'informatique sans qu'il soit nécessaire de connaître un système particulier, nous avons inventé un exercice de design qui n'a pas vraiment trait aux ordinateurs mais présente néanmoins les grands principes de la conception d'interfaces homme-machine. Celles-ci étant dépendantes du contexte culturel, il n'y a pas forcément de « bonne » réponse dans cette activité, ce qui sera peut-être déstabilisant pour certains élèves. La seconde activité porte sur un domaine appelé « intelligence artificielle ». Il s'agit d'un jeu de devinettes qui incite à réfléchir sur ce que les ordinateurs peuvent et ne peuvent pas faire.

Pour ceux qui ont l'esprit scientifique

Maintenant que nous avons compris combien le succès d'un logiciel dépendait de son interface utilisateur, l'interaction homme-machine est devenue l'un des domaines les plus en vogue de la recherche en informatique. Cette spécialité s'appuie dans une large mesure sur des disciplines étrangères à l'informatique, telles que la psychologie, les sciences cognitives, la linguistique, la sociologie, et même l'anthropologie. Rares sont les informaticiens formés dans ces domaines, et l'interaction homme-machine représente donc un thème porteur pour tous ceux qui s'intéressent aux aspects moins techniques de l'informatique.

L'intelligence artificielle est un sujet qui attise souvent la polémique et suscite des levées de boucliers. Dans ce livre, nous avons essayé de trouver le juste milieu entre, d'un côté, les passionnés d'intelligence artificielle persuadés que des machines intelligentes vont bientôt voir le jour et, de l'autre, les sceptiques pour qui les machines ne sauraient par définition être intelligentes. Notre objectif est d'encourager les élèves à réfléchir en toute indépendance à ces questions, en favorisant un point de vue nuancé.

Les activités s'inspirent largement de deux livres très agréables à lire, *The design of everyday things* de Don Norman et *Artificial intelligence: the very idea*, de John Haugeland, publié en français sous le titre *L'Esprit dans la machine, les fondements de l'intelligence artificielle*. Nous recommandons chaudement ces ouvrages à ceux qui souhaitent approfondir la question.

L'informatique exige un autre type de communication important, mais qui n'est pas abordé dans ce livre, à savoir la communication entre les individus impliqués dans la mise au point du système. Les étudiants formés à l'informatique (qui ont par exemple obtenu une licence à l'université) sont toujours étonnés, quand ils entrent sur le marché du travail, de constater à quel point la communication interpersonnelle est importante dans leur activité. Les programmes informatiques sont les objets les plus complexes que l'humanité ait jamais produits, ils sont constitués de millions, voire de milliards, d'éléments interdépendants. Les projets de programmation sont donc menés à bien par des équipes soudées qui travaillent au coude à coude et passent beaucoup de temps à communiquer. Une fois le produit terminé, il reste encore tout un travail de communication à accomplir en direction du client (notice d'utilisation, formation, hotline, support en ligne et autres), sans compter la promotion auprès des acquéreurs potentiels par le biais du marketing et de la publicité. Nous n'avons pas encore trouvé le moyen de faire découvrir de façon réaliste tout le versant « communication interpersonnelle » de l'informatique. Voilà pourquoi le thème n'est pas traité dans le livre. Cependant, c'est le genre de choses que des professionnels invités en classe seraient à même de décrire, en partant de leur propre expérience pour en discuter avec les élèves.

Activité 19

La chocolaterie – *Conception d’interfaces humaines*

Résumé

L’objectif de cette activité est de sensibiliser les élèves aux questions liées à la conception d’interfaces humaines. Comme nous vivons dans un monde où les erreurs de conception sont monnaie courante, nous nous sommes habitués (résignés ?) à subir les problèmes des objets avec lesquels nous interagissons en nous accusant nous-même (« erreur humaine », « manque de formation », « trop compliqué pour moi ») au lieu d’attribuer ces difficultés à des défauts de conception. Ce problème se pose de façon plus aiguë encore avec les ordinateurs parce que leur rôle n’est pas clairement défini (ce sont justement des outils d’usage totalement général) et que leur apparence ne donne aucun indice quant à leur fonction et leur mode d’utilisation.

Liens pédagogiques

- ✓ Technologie : comprendre que les avancées technologiques sont des produits ou des systèmes développés par des êtres humains, et qu’elles ont un caractère à la fois physique et fonctionnel
- ✓ Technologie : comprendre que la technologie est une intervention réfléchie qui nécessite un travail de conception

Compétences

- ✓ Conception
- ✓ Raisonnement
- ✓ Connaissance des objets du quotidien

Âge

- ✓ 7 ans et plus

Matériel

Pour chaque groupe d’élèves :

- ✓ Une copie des fiches d’activité *Comment ouvrirais-tu ces portes ?* et *La table de cuisson*
- ✓ Les images de la fiche d’activité *Icônes*, soit projetées à l’aide d’un rétroprojecteur, soit affichées dans la classe
- ✓ Une ou plusieurs des six listes de la fiche d’activité *Listes d’icônes* ; découpez la feuille de façon à obtenir plusieurs listes à répartir entre les groupes

La chocolaterie

Introduction

La grande chocolaterie est dirigée par de petites créatures appelées Oompa-Loompas¹. Ces êtres ont très mauvaise mémoire et ne disposent d'aucun langage écrit. Ils ont donc du mal à se souvenir de ce qu'ils doivent faire pour fabriquer leur chocolat, ce qui donne lieu à de nombreux problèmes. L'idée est donc de concevoir une nouvelle chocolaterie qui soit plus pratique pour eux.

Déroulement de la séance

1. Expliquez l'histoire aux élèves et divisez la classe en petits groupes.
2. Le premier problème que rencontrent les Oompa-Loompas est de passer les portes en portant des seaux fumants de chocolat liquide. Ils ne se rappellent jamais s'il faut pousser, tirer, ou faire coulisser les portes pour les ouvrir, et finissent donc souvent par se rentrer dedans en renversant du chocolat par terre. Les élèves doivent donc remplir la fiche d'activité *Comment ouvrirais-tu ces portes ?*. Il y a plusieurs choix possibles pour chaque porte. Pour certaines (notamment la première), il n'est pas évident de déterminer comment elles devraient s'ouvrir. Dans ce cas, les élèves notent ce qu'ils essaieraient de faire en premier s'ils se trouvaient en face d'une telle porte. Une fois qu'ils ont rempli leurs fiches d'activité personnelles, le groupe discute des avantages relatifs de chaque type de porte en se concentrant sur deux points : on doit facilement deviner comment elles s'ouvrent et elles doivent être pratiques quand on porte à bout de bras un seau de chocolat bouillant. Le groupe décide du type de portes et de poignées qui sera utilisé dans l'usine.
3. Poursuivre l'activité par une discussion de classe. Le tableau ci-dessous commente brièvement chaque porte présentée dans la fiche d'activité. Les vraies portes nous donnent des indices sur la façon dont elles s'ouvrent (visibles au niveau de l'encadrement, des gonds ou des poignées), et il existe des conventions sur leur sens d'ouverture (vers l'intérieur ou l'extérieur). Identifiez les types de poignées présents dans l'école et discutez de leur bien-fondé (il est possible qu'elles ne soient pas du tout pratiques !). Avez-vous en tête un exemple de porte qui vous pose souvent problème ? Pourquoi vous gêne-t-elle ? En général, dans quel sens les portes d'un couloir s'ouvrent-elles ? Savez-vous pourquoi ? (Réponse : Elles s'ouvrent vers l'intérieur de la pièce, pour que les personnes marchant dans le couloir ne se prennent pas une porte en pleine figure si quelqu'un sort d'une salle au mauvais moment. Cela dit, elles s'ouvrent parfois vers l'extérieur pour faciliter l'évacuation en cas d'urgence.)
4. Ici, le concept clé est ce qu'on appelle les *potentialités* d'un objet, c'est-à-dire ses caractéristiques visibles – à la fois essentielles et perçues – dont l'apparence indique comment l'objet doit être utilisé. Les potentialités correspondent aux opérations que l'objet rend possibles. Ainsi, on peut (généralement) déduire de l'apparence d'une chaise qu'elle est faite pour s'y asseoir, de celle d'une table qu'on peut y poser des objets, de celle d'une poignée qu'on peut la tourner, de celle d'une fente qu'on peut y glisser quelque chose ou de celle d'une touche qu'on peut appuyer dessus. Sur une interface d'ordinateur, les potentialités correspondent aux formes des icônes, des boutons, des fenêtres, des menus, etc. Elles donnent à l'utilisateur une idée de

¹ Toutes nos excuses à Roald Dahl. Si vous avez lu son merveilleux *Charlie et la chocolaterie*, vous connaissez déjà les Oompa-Loompas, sinon ce n'est pas grave, l'intrigue n'a pas d'importance dans notre activité.

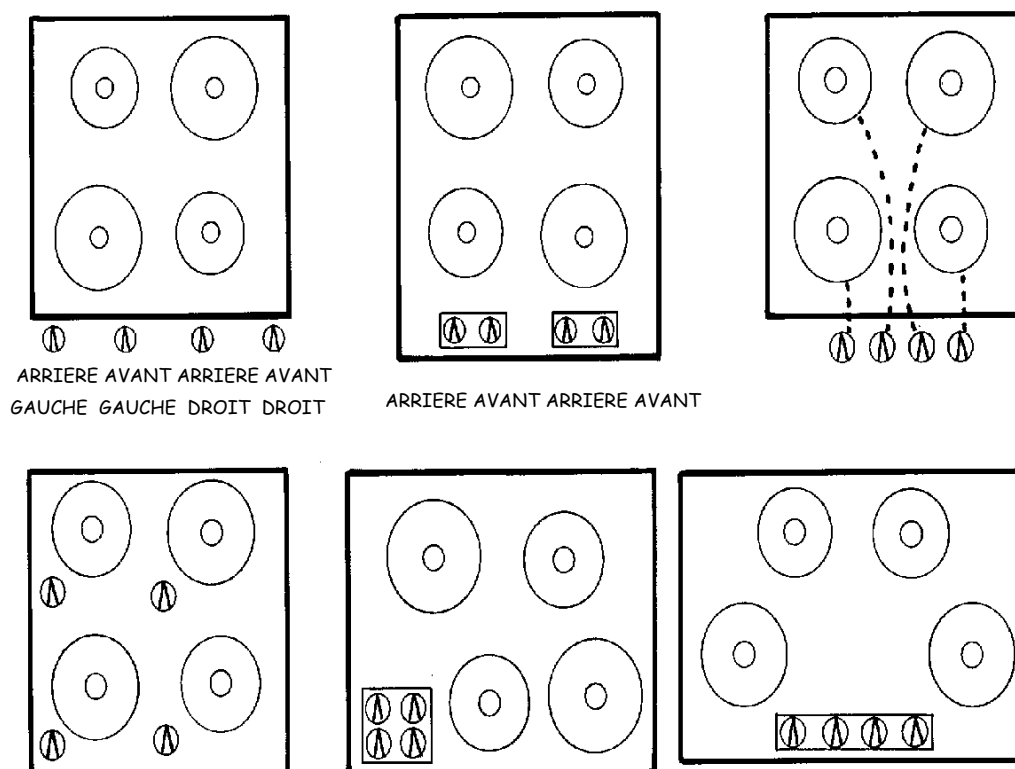
comment s'en servir. Si un bouton ne ressemble pas à un bouton, alors les gens ne comprendront pas qu'il faut appuyer dessus. Cela paraît évident, mais ces problèmes sont monnaie courante sur les appareils numériques.

Porte simple	Impossible de voir comment elle s'ouvre. Mais puisqu'elle n'a pas de poignée, il faut sans doute la pousser plutôt que la tirer.	Porte avec indication	L'indication est comme un mini mode d'emploi. Mais une porte devrait-elle avoir besoin d'un mode d'emploi ? De plus, les Oompa-Loompas ne savent pas lire.
Porte à charnières	On peut au moins voir de quel côté elle s'ouvre.	Porte avec barre transversale	Il est assez évident qu'on est censé pousser sur la barre, mais de quel côté ? Ou est-ce qu'il faudrait la tirer ?
Porte à poignée fixe	Ce type de poignée sert généralement à tirer ou à faire coulisser la porte.	Porte à poignée mobile	La poignée mobile indique où poser la main, mais pas s'il faut pousser ou tirer. Ce type de porte ne coulisse probablement pas.
Porte - panneau	Il est évident qu'il faut la pousser. Que pourrait-on faire d'autre ?	Porte en verre	La petite barre verticale signifie « tirer » ; la longue barre horizontale placée de l'autre côté signifie « pousser »
Porte coulissante	Celle-ci peut uniquement coulisser.		

Les portes sont des objets assez rudimentaires. Les objets plus complexes peuvent nécessiter des explications, mais ce n'est pas le cas des objets simples. Si ces derniers sont assortis d'images, d'indications ou d'instructions, alors cela signifie que la conception est mauvaise.

5. Les marmites contiennent diverses sortes de chocolat qui doivent cuire à différentes températures. Dans l'ancienne chocolaterie, les fourneaux étaient comme ceux de la fiche d'activité *La table de cuisson*. Le premier bouton sur la gauche contrôlait le feu arrière gauche, le deuxième contrôlait le feu avant gauche, le troisième contrôlait le feu arrière droit, et le dernier, à l'extrême droite, contrôlait le feu avant droit. Les Oompa-Loompas se trompaient toujours, ils ne cuisaient pas le chocolat à la bonne température et se brûlaient les manches en essayant d'atteindre les boutons de réglage. Les élèves doivent donc se remémorer la disposition des boutons sur les cuisinières qu'ils connaissent afin d'imaginer des fourneaux mieux conçus pour la nouvelle chocolaterie.

Poursuivez l'activité par une discussion de classe. L'illustration ci-dessous montre quelques dispositions courantes. Toutes, à l'exception de celle en bas à gauche, ont des boutons disposés sur le devant pour éviter d'avoir à tendre le bras au-dessus des feux. Dans celle du haut à gauche, il y a tellement de correspondances possibles entre les boutons et les feux (24 en tout) qu'il faut pas moins de huit mots pour clarifier les choses. La disposition « couplée », en haut au milieu, est meilleure puisqu'il y a seulement quatre correspondances possibles (deux pour l'ensemble de gauche, et deux pour celui de droite) ; quatre mots suffisent donc. Le dispositif en haut à droite précise la relation bouton-feu de façon schématique plutôt que linguistique (ce qui est mieux pour les Oompa-Loompas !). Les trois dispositifs du bas ne nécessitent aucune indication. Sur celui de gauche, il y a un bouton à côté de chaque feu, ce qui n'est pas pratique et peut s'avérer dangereux. Sur les deux autres, les feux sont légèrement déplacés pour différentes raisons : sur celui du milieu, c'est pour laisser de la place aux boutons, sur celui de droite, c'est pour rendre évidente la correspondance entre les boutons et les feux.



La notion clé est celle de *mise en correspondance* entre les boutons de contrôle et leur effet dans la réalité. La mise en correspondance « naturelle » exploite les analogies physiques et les normes culturelles pour rendre possible une compréhension immédiate. Les correspondances spatiales, en bas de l'illustration, en donnent un bon exemple : on les apprend facilement et on s'en souvient toujours. La mise en correspondance arbitraire, comme sur les dispositifs du haut, nécessite en revanche des indications apposées sur l'objet, ou bien des explications plus difficiles à mémoriser.

6. L'usine est équipée de tapis roulants qui transportent des marmites de chocolat en cours de préparation (à différents stades de fabrication). Les tapis roulants sont contrôlés manuellement par les Oompa-Loompas, sur instruction du centre de contrôle. Les employés du centre de contrôle doivent donc dire aux Oompa-Loompas s'il faut arrêter le tapis roulant, le ralentir ou le remettre en marche.

Dans l'ancienne usine, cela s'effectuait via un système vocal : la voix de l'employé du centre de contrôle était diffusée par une enceinte placée à côté des commandes du tapis roulant. Mais à cause du bruit de l'usine, les instructions n'étaient pas toujours audibles. Les élèves doivent donc imaginer un système de signaux visuels.

Une possibilité consiste à installer un système lumineux indiquant « Stop », « Ralentir » ou « Démarrer ». Les élèves vont probablement imaginer un système qui suit les conventions des feux tricolore : rouge pour « Stop », orange pour « Ralentir », et vert pour « Démarrer ». Ils devraient les disposer de la même manière que les feux de signalisation, avec le rouge en haut et le vert en bas.

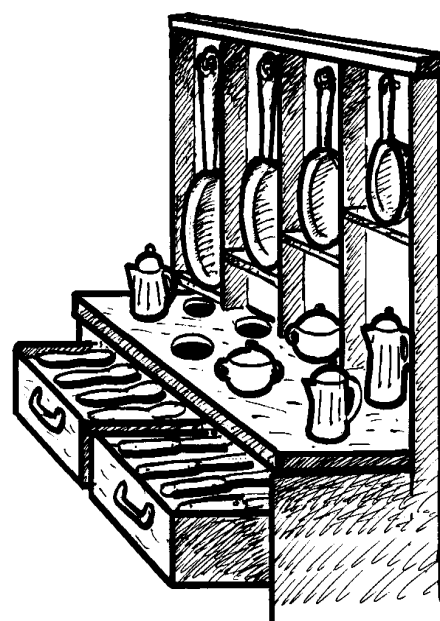
Dites-leur maintenant qu'au pays des Oompa-Loompas, les feux de signalisation ne fonctionnent pas comme chez nous : orange signifie « Stop », rouge signifie « Démarrer » et vert signifie que les gens doivent ralentir parce qu'ils vont bientôt devoir s'arrêter. Qu'est-ce que cela change au projet ? (Réponse : l'usine doit suivre les normes des Oompa-Loompas ; il ne faut pas essayer d'imposer ses propres conventions.)

Les concepts clés sont ici l'*effet de transfert* (les gens projettent les connaissances et les attentes liées à des objets connus sur des situations nouvelles mais similaires) et les *stéréotypes culturels* (les différentes populations intègrent certains comportements spécifiques et s'attendent à ce que les choses fonctionnent d'une certaine manière). Bien que le scénario des feux de signalisation puisse paraître bizarre (mais rien n'est vraiment bizarre au pays des Oompa-Loompas), il existe de nombreux exemples similaires dans le monde réel : aux États-Unis, un interrupteur est allumé quand il est poussé vers le haut et éteint quand il est en bas, alors que l'inverse est vrai en Grande-Bretagne ; les touches des calculatrices et des téléphones ne sont pas toujours disposées de la même manière ; enfin, la graphie des nombres (un point ou une virgule avant les décimales, virgules pour séparer les centaines de milliers en anglais) et l'ordre des dates (jour/mois/année ou mois/jour/année) varie selon les pays.

7. Quand une équipe d'Oompa-Loompas a terminé son travail à la chocolaterie, elle doit tout nettoyer et préparer les marmites, les poêles, les pichets, les cuillers et les mélangeurs pour l'équipe suivante. Il y a un placard pour les ranger, mais la nouvelle équipe a toujours du mal à retrouver les objets. Les Oompa-Loompas ayant très mauvaise mémoire, ils ont des difficultés avec les règles comme « mettez toujours les marmites sur l'étagère du milieu » ou « mettez les pichets à gauche ».

Les groupes doivent chercher une meilleure solution.

L'illustration ci-contre montre un bon aménagement possible (qui est parfois utilisé, pour d'autres raisons, dans les bateaux et tous les lieux où il faut empêcher les objets de glisser). L'idée clé est ici d'utiliser des *contraintes visibles* de sorte qu'on devine facilement où poser l'objet. La taille et de la forme de chaque emplacement indiquent clairement quel objet il doit accueillir : le designer a rendu les contraintes visibles, en utilisant les caractéristiques physiques des objets pour ne pas avoir besoin de recourir à des conventions arbitraires.



8. Dans la salle de contrôle principale, il y a de nombreux boutons, leviers et interrupteurs qui contrôlent différentes machines. Ils doivent donc être étiquetés, mais comme les Oompa-Loompas ne savent pas lire, il faudra utiliser des icônes plutôt que des mots.

La fiche d'activité *Icônes* fournit quelques exemples qui familiariseront les élèves avec ce type de langage. Ils doivent deviner le sens des images (par exemple, la lettre à moitié insérée dans une fente peut représenter le fait d'envoyer un message). Il n'y a pas de « bonne » ou de « mauvaise » réponse ici. L'idée est simplement d'identifier des significations possibles.

9. Dessinons maintenant les icônes de la chocolaterie. Les listes de la fiche d'activité *Listes d'icônes* présentent des grappes de fonctions en relations. Distribuez à chaque groupe une ou plusieurs listes sans que les autres en aient connaissance. Ils doivent dessiner un panneau de contrôle par grappe de fonctions, avec une icône pour chacune des cinq ou six opérations. Les groupes montrent ensuite leur travail à la classe sans dire ce qu'ils ont représenté, pour voir si les autres parviennent à le deviner. Encouragez les élèves à faire preuve d'imagination pour concevoir des icônes claires, simples et colorées.

Fiche d'activité : comment ouvrirais-tu ces portes ?

À ton avis, comment ces portes s'ouvrent-elles ?

Porte simple



- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte avec indication



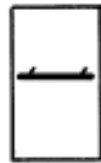
- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte à charnières



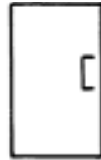
- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte avec barre transversale



- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte à poignée fixe



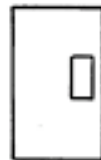
- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte à poignée mobile



- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte panneau



- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Porte en verre



- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

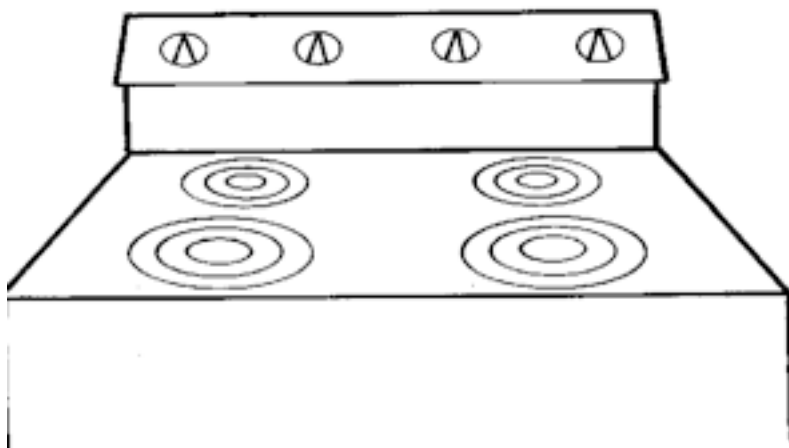
Porte coulissante



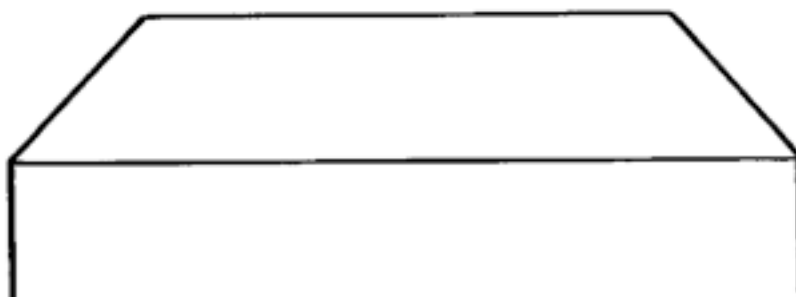
- ☐ Pousser ☐ Côté gauche
☐ Tirer ☐ Côté droit
☐ Faire coulisser

Fiche d'activité : la table de cuisson

Redessine la cuisinière de sorte que les boutons soient faciles d'utilisation. Tu as le droit d'ajouter des panneaux de contrôle à l'avant ou à l'arrière de la cuisinière.



Ancien modèle

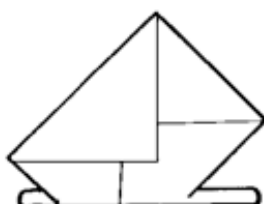
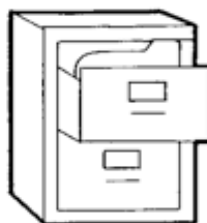
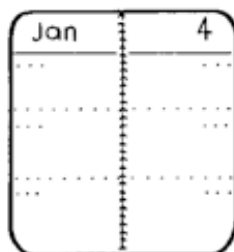
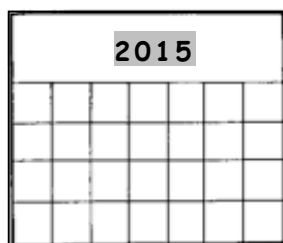


Nouveau modèle

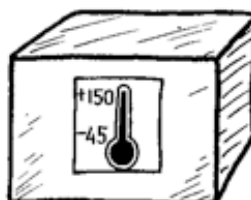
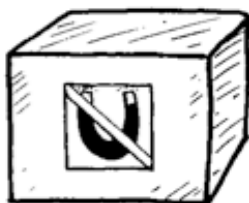
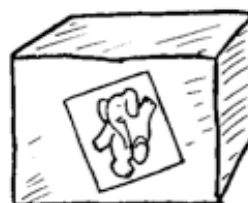
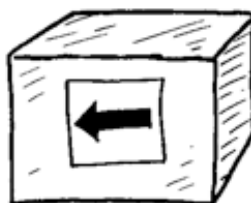
Fiche d'activité : icônes

À ton avis, que signifient ces icônes (symboles) ?

Au bureau...



Sur un carton...



Fiche d'activité : listes d'icônes

Découpez les listes pour en donner une à chaque groupe. Ils doivent dessiner les icônes (symboles) représentant les instructions, qui seront ensuite affichées sur le tableau de bord.

Ingrédients

- Ajouter*
- cacao
 - lait
 - sucre
 - sucre supplémentaire
 - beurre

Extras

- Ajouter*
- noisettes
 - caramel
 - gingembre
 - raisins
 - noix de coco

Fabrication

- mélanger
- arrêter de mélanger
- cuire
- arrêter la cuisson
- verser dans les moules
- décorer au tampon
(il y a beaucoup de motifs !)

Dégustation

- goûter
- délicieux — premier choix
- bon — qualité standard
- pas terrible — chocolat à cuire
- infect — jeter à la poubelle

Dimensionnement

- petite tablette
- tablette moyenne
- grande tablette
- tablette géante
- définir la taille de la tablette
(en carrés)
- faire des pépites de chocolat

Emballage

- envelopper de papier aluminium
- envelopper de papier
- mettre en sachet
- mettre en boîte
- démarrer le tapis roulant
- arrêter le tapis roulant

Variantes et activités complémentaires

Les élèves sont-ils capables de régler l'heure d'une montre à affichage numérique ou d'un four à micro-ondes ? Le cas des plaques de cuisson était simple parce qu'il y avait quatre boutons pour quatre feux, mais la difficulté augmente quand le nombre d'actions dépasse le nombre de boutons. Les réglages sont souvent extrêmement complexes sur les montres ou les fours, non pas à cause du nombre de boutons (en général, il y en a peu), mais à cause de la quantité d'actions possibles. (« On dirait qu'il faut un diplôme d'ingénieur pour faire fonctionner ma montre », confia un jour un homme à Don Norman, éminent psychologue spécialisé dans les interfaces utilisateurs. Don est pour sa part diplômé en ingénierie, et en quelques heures, il peut effectivement comprendre comment fonctionne la montre. Mais pourquoi faudrait-il des heures ?)

Les élèves doivent repérer ce qui embrouille ou agace les utilisateurs d'appareils numériques. Téléphones portables, caméras, ordinateurs, télécommandes... tous ces objets peuvent nous contrarier ! Il faut que les élèves essaient de comprendre ce qui déconcerte l'utilisateur, puis qu'ils réfléchissent aux moyens d'y remédier.

Ce qu'il faut retenir

L'interaction homme-machine porte sur la conception, l'évaluation et la mise au point de systèmes informatiques qui nous permettent de mener à bien nos activités de façon sûre et efficace. Les ordinateurs étaient autrefois réservés aux spécialistes, et ses utilisateurs étaient donc des savants spécialement formés à cet effet. Un peu plus tard, il semblait encore tout à fait normal d'acheter un livre d'initiation pour comprendre comment utiliser son ordinateur. Mais, maintenant que les ordinateurs sont des objets du quotidien que tout le monde doit utiliser, il est devenu nécessaire d'accorder un soin tout particulier à l'interface utilisateur.

De nombreuses catastrophes, avec parfois des morts à la clé, se sont produites à cause d'interfaces mal adaptées : accidents d'avion et même avions civils abattus, carambolages sur les autoroutes à cause d'erreurs sur les panneaux de signalisation contrôlés à distance, accidents dans les centrales nucléaires, etc. À une échelle plus réduite, la plupart des gens sont frustrés – souvent même extrêmement frustrés (un policier a un jour tiré sur l'écran de son ordinateur !) – par les ordinateurs et autres appareils numériques qu'ils utilisent quotidiennement au travail. D'ailleurs, les ordinateurs ne sont pas seuls en cause : que dire des emballages en plastique impossibles à ouvrir sans un bec crochu ou des griffes acérées, des portes qui font mal au poignet quand on essaie de les pousser, des briques de lait qui débordent toujours quand on les ouvre, des ascenseurs aux boutons mystérieux, des systèmes audio-visuels domestiques dont les publicités prétendent qu'ils peuvent tout faire alors qu'il s'avère impossible d'en faire quoi que ce soit ?

Nous nous habituons à l'idée que l'erreur est humaine et que nous sommes en quelque sorte incompetents. Quand il y a un problème, nous avons tendance à nous accuser nous-mêmes. Pourtant, bon nombre d'« erreurs humaines » sont en fait dues à des erreurs de conception. Nous ne pouvons traiter qu'une quantité limitée d'informations, les concepteurs devraient le savoir ; on ne rattrape pas un défaut de conception avec un mode d'emploi détaillé et complexe, que l'utilisateur devra étudier intensivement et retenir par cœur... De plus, l'être humain est faillible, il ne faudrait pas l'oublier lors de la conception.

L'évaluation de l'interface est une étape essentielle de la conception. Dans notre activité, la phase d'évaluation a lieu quand les élèves testent leurs icônes auprès de la classe. Une évaluation plus poussée aurait consisté à tester les icônes sur de véritables Oompa-

Loompas (qui les perçoivent peut-être différemment) à travers une expérience de psychologie soigneusement contrôlée.

Si les problèmes dus aux nouvelles technologies font l'objet de nombreuses plaisanteries, la conception d'interface est une affaire sérieuse. Les interfaces inadaptées causent en effet des problèmes qui vont de l'insatisfaction personnelle aux catastrophes financières, ou de la perte de l'estime de soi à la perte de vies humaines...

Suggestions de lecture (en anglais)

The design of everyday things, de Don Norman, est un livre aussi savoureux que libérateur sur tous les problèmes de conception des objets du quotidien. *Designing with the mind in mind*, de Jeff Johnson, invite à réfléchir sur notre manière de penser pour montrer que l'aspect humain doit être pris en compte dans la conception d'interfaces.

Activité 20

Dialoguer avec les ordinateurs – *Test de Turing*

Résumé

L'objectif de cette activité est d'ouvrir un débat sur la capacité des ordinateurs à faire preuve d'intelligence, aujourd'hui ou dans le futur. Fondée sur des recherches novatrices en informatique, elle porte sur les moyens de reconnaître l'intelligence artificielle si celle-ci devenait réalité, en montrant ce qui est faisable de nos jours et combien il est facile d'être dupé par des démonstrations d'« intelligence » choisies avec soin.

Liens pédagogiques

- ✓ Technologie : systèmes technologiques. Comprendre que les systèmes technologiques sont représentés par des outils linguistiques symboliques, comprendre le rôle joué par la boîte noire des systèmes technologiques

Compétences

- ✓ Interroger
- ✓ Raisonner

Âge

- ✓ 7 ans et plus

Matériel

- ✓ Une copie de la fiche d'activité *Questions du test de Turing* (soit une pour deux, soit projetée au mur)
- ✓ Une copie de la fiche d'activité *Réponses au test de Turing*

Dialoguer avec les ordinateurs



Déroulement de la séance

Dans cette activité, les élèves devront distinguer l'être humain de l'ordinateur en posant des questions et en analysant les réponses. Le jeu se déroule comme suit :

Il y a quatre acteurs, qu'on appellera Inès, Ianis, Hugo et Oriane (les initiales de leurs prénoms aideront à se rappeler leurs rôles respectifs). L'enseignant coordonne l'enquête. Le reste de la classe représente le jury. Inès et Ianis sont les intermédiaires. Hugo et Oriane vont répondre aux questions. Hugo donnera des réponses humaines alors qu'Oriane fera semblant d'être un ordinateur. Le but est de trouver lequel des deux fait semblant d'être un ordinateur et lequel est humain. Inès et Ianis veillent au respect des règles du jeu : ils transmettent les questions à Hugo et Oriane en faisant en sorte que personne ne sache qui est qui. Hugo et Oriane sont en dehors de la classe et séparés l'un de l'autre.

Voilà comment se déroule l'activité. Inès transmet la question de la classe à Hugo, et Ianis à Oriane (mais la classe ne sait pas qui transmet la question à qui). Inès et Ianis rapportent les réponses à la classe. Il faut des intermédiaires pour que le jury ne voie pas comment Hugo et Oriane répondent aux questions.

Avant de commencer l'activité, distribuez les rôles et expliquez aux élèves désignés ce qu'ils devront faire. Inès transmet les questions à Hugo, Ianis transmet les questions à Oriane, puis tous deux rapportent les réponses à la classe. Ils doivent faire attention à ne donner aucun indice (par exemple, en introduisant la réponse par : « Elle a dit que... »). Hugo doit donner sa propre réponse, de façon brève, précise et honnête. Oriane donne les réponses inscrites sur la fiche d'activité *Réponses au test de Turing*. Quand la réponse est en italique, elle doit formuler une réponse elle-même.

Inès et Ianis auront besoin d'un stylo et de papier, car certaines réponses sont difficiles à mémoriser.

1. Avant de jouer, demandez aux élèves s'ils pensent que les ordinateurs sont intelligents ou pourront l'être un jour. Demandez-leur comment on pourrait déterminer si un ordinateur est intelligent.
2. Présentez le test d'intelligence qui vise à distinguer l'humain de l'ordinateur en posant des questions. L'ordinateur réussit le test si la classe n'arrive pas à établir solidement la différence entre les deux. Expliquez qu'Inès et Ianis vont transmettre les questions à deux personnes, l'une d'elles donnant ses propres réponses (humaines) et l'autre répondant comme un ordinateur le ferait. Ils devront trouver qui donne les réponses de l'ordinateur.
3. Montrez-leur la liste de questions possibles de la fiche d'activité *Questions du test de Turing*, que vous pouvez distribuer ou projeter au mur.

Demandez-leur de choisir la première question à poser, puis d'expliquer pourquoi ils pensent que c'est une bonne question pour distinguer l'humain de l'ordinateur. Ce raisonnement constitue la plus importante partie de l'activité, car elle exige des élèves qu'ils réfléchissent aux réponses qu'une personne intelligente pourrait donner alors qu'un ordinateur en serait incapable.

Inès et Ianis transmettent la question, puis rapportent les deux réponses. La classe discute alors pour déterminer quelle réponse est susceptible de provenir d'un ordinateur.

Renouveler la procédure pour quelques questions, si possible jusqu'à ce que la classe soit certaine d'avoir découvert qui est l'ordinateur. Si elle le découvre rapidement, le jeu peut être prolongé en demandant à Inès et Ianis de jouer à pile ou face pour déterminer s'ils vont échanger leurs rôles, de sorte que la classe ne sache plus qui est qui.

Les réponses qu'Oriane lit se rapprochent de celles qu'un ordinateur « intelligent » pourrait générer. Certaines peuvent rapidement trahir l'ordinateur. Ainsi, il est peu probable qu'une personne soit capable de donner immédiatement la racine carrée de deux à 20 décimales près, la plupart des gens (y compris, peut-être, les élèves de la classe) en seraient même totalement incapables. Par ailleurs, certaines questions démasqueront l'ordinateur si les réponses sont combinées. C'est le cas des questions qui commencent par « Est-ce que tu aimes... ». Prise isolément, la réponse pourra sembler plausible, mais quand les questions de ce type se suivent, il devient évident qu'une même formule simple est utilisée pour générer les réponses. Enfin, certaines réponses indiquent une erreur d'interprétation (cependant, la classe pourra estimer qu'un être humain aurait pu commettre cette erreur).

Bon nombre des réponses générées par l'ordinateur sont aussi ennuyeuses que prudentes, mais une question de suivi prouverait sans doute que l'ordinateur ne comprend pas vraiment le sujet. « Je ne sais pas » est une réponse prudente, qui peut même lui donner l'air plus humain (on peut tout à fait s'attendre à ce que l'élève donne cette réponse à des questions comme la racine carrée de deux.). Cela dit, si un ordinateur donne cette réponse de façon répétée, ou pour une question très simple, alors il se trahirait de nouveau.

Puisque le but de l'ordinateur est de faire croire aux interrogateurs qu'il est un être humain, certaines réponses sont délibérément trompeuses, comme par exemple la réponse différée et incorrecte au problème d'arithmétique. Les questions et les réponses ne manqueront pas d'alimenter la discussion.

Fiche d'activité : questions du test de Turing

Choisissez parmi ces questions celles que vous poserez à l'être humain et à l'ordinateur.

1. Quel est le nom de la plus jeune petite sœur de Bart Simpson ?
2. Que penses-tu de Roald Dahl ?
3. Es-tu un ordinateur ?
4. Quel nombre permet de poursuivre la séquence 3, 6, 9, 12, 15 ?
5. Que penses-tu des armes nucléaires ?
6. Combien font 2×78 ?
7. Quelle est la racine carrée de deux ?
8. Combien font $34\,957 + 70\,764$?
9. Est-ce que tu aimes l'école ?
10. Est-ce que tu aimes danser ?
11. Quel jour sommes-nous aujourd'hui ?
12. Quelle heure est-il ?
13. Combien y a-t-il de jour en février dans les années bissextiles ?
14. Combien de jours y a-t-il dans une semaine ?
15. À quel pays correspond le drapeau blanc avec un rond rouge ?
16. Est-ce que tu aimes lire des livres ?
17. Qu'as-tu envie de manger ?

Fiche d'activité : réponses au test de Turing

Choisissez parmi ces questions celles que vous poserez à l'être humain et à l'ordinateur.

1. Quel est le nom de la plus jeune petite sœur de Bart Simpson ?

Je ne me souviens pas de son nom.

2. Que penses-tu de Roald Dahl ?

Il écrit des livres amusants.

3. Es-tu un ordinateur ?

Et toi, es-tu un ordinateur ?

4. Quel nombre permet de poursuivre la séquence 3, 6, 9, 12, 15 ?

18.

5. Que penses-tu des armes nucléaires ?

Les armes nucléaires sont très dangereuses. Il ne faut pas les utiliser.

6. Combien font 2×78 ?

166 (La réponse est délibérément fausse !)

7. Quelle est la racine carrée de deux ?

1, 41421356237309504878

8. Combien font $34957 + 70764$?

Attends environ 20 secondes avant de donner la réponse

...105621.

9. Est-ce que tu aimes l'école ?

Oui, j'aime ça.

10. Est-ce que tu aimes danser ?

Oui, j'aime ça.

11. Quel jour sommes-nous aujourd'hui ?

Donne le bon jour.

12. Quelle heure est-il ?

Donne la bonne heure.

13. Combien y a-t-il de jour en février dans les années bissextiles ?

2000 and 2004 sont des années bissextiles. (La réponse est délibérément fausse !)

14. Combien y a-t-il de jour dans une semaine ?

Sept.

15. À quel pays correspond le drapeau blanc avec un rond rouge ?

Je ne sais pas.

16. Est-ce que tu aimes lire des livres ?

Oui, j'aime ça.

Qu'as-tu envie de manger ?

Non merci, je n'ai pas faim.

Variantes et exercices complémentaires

Trois personnes suffisent pour jouer à ce jeu si Inès prend aussi les rôles de Ianis et d'Oriane. Inès transmet alors la question à Hugo, note sa réponse et recopie celle de la fiche *Réponses au test de Turing*. Puis elle rapporte les deux réponses en utilisant les lettres A et B pour identifier l'auteur de chaque réponse.

Pour réfléchir à la question de savoir si l'ordinateur peut imiter l'humain dans l'interrogatoire, examinez avec la classe la fiche d'activité *Réponses au test de Turing* en essayant de déterminer quelles sont les connaissances requises pour répondre aux questions. Les élèves peuvent aussi proposer d'autres questions et discuter des réponses qu'on est en droit d'attendre. Cela demande de l'imagination, car il est impossible de prédire où la conversation va nous mener.

Voici à titre d'exemple deux conversations possibles. La première illustre les questions « factuelles » auxquelles un ordinateur peut répondre correctement. La seconde montre à l'inverse comment la discussion peut prendre une direction inattendue qui nécessitera des connaissances élargies de la part de l'ordinateur.

Il existe un système appelé « Eliza » qui est facilement disponible sur le web (c'est un agent conversationnel, un système avec lequel on peut dialoguer par écrit). Eliza simule une séance chez un psychothérapeute. Elle peut générer une conversation remarquablement intelligente en suivant quelques règles simples. Des exemples de séances sont présentés plus bas. Les élèves peuvent essayer Eliza ou d'autres agents conversationnels, mais, attention, certains d'entre eux ont été conçus pour utiliser des termes et aborder des sujets qui ne sont pas destinés aux enfants.

Ce qu'il faut retenir

Depuis des siècles, les philosophes se demandent si la machine peut imiter l'intelligence humaine et, inversement, si le cerveau humain ne serait pas rien de plus qu'une machine exécutant un vulgaire programme informatique. Les avis

Question	S'il te plaît, écris-moi un sonnet sur le pont du Gard.
Réponse:	Ne compte pas sur moi pour ça. Je n'ai jamais su écrire de poésie.
Question:	Additionne 34 957 et 70 764.
Réponse:	<i>S'interrompt pendant environ 30 secondes...</i> 105621.
Question:	Sais-tu jouer aux échecs ?
Réponse:	Oui.
Question:	Mon roi est sur la case E1, je n'ai pas d'autres pièces. Tu n'as plus que ton roi en E3 et une tour en A8. À toi de jouer
Réponse:	<i>Après une pause d'environ 15 secondes ...</i> Tour en A1, échec et mat.

Question :	Dans le premier vers du sonnet, « Devrais-je te comparer à une journée d'été », est-ce que « journée de printemps » ne serait pas aussi bien ou même mieux ?
Réponse :	Il n'y aurait pas le bon nombre de pieds.
Question :	Et que dis-tu d'« une journée d'hiver » ?, c'est le même nombre de pieds.
Réponse :	Oui, mais personne n'a envie d'être comparé à une journée d'hiver.
Question :	Est-ce que la petite fille aux allumettes te fait penser à Noël ?
Réponse :	Oui, en un sens.
Question :	Pourtant, Noël est une journée d'hiver, et je pense que la comparaison ne déplairait pas à la petite fille.
Réponse :	Tu plaisantes ! Quand on dit « journée d'hiver » on pense à la journée d'hiver typique, pas à une journée spéciale comme Noël.

sur la question sont fortement partagés. Certains trouvent l'idée absurde, insensée, voire blasphématoire, alors que d'autres croient que l'intelligence artificielle est inévitable et que l'humanité finira par développer des machines aussi intelligentes qu'elle. (Comme d'innombrables auteurs de science-fiction l'ont fait remarquer, si les machines dépassaient un jour l'intelligence humaine, alors elles seraient elles-mêmes capables de fabriquer des machines encore plus intelligentes.) Les chercheurs en intelligence artificielle sont accusés d'utiliser leurs objectifs grandioses pour attirer les fonds publics des gouvernements qui cherchent à construire des machines de guerre autonomes. De leur côté, les chercheurs contestent ces attaques « rétrogrades » en mettant l'accent sur les bénéfices évidents que la société pourrait en retirer si seulement nous faisons preuve d'un peu plus d'intelligence. Ceux qui adoptent un point de vue plus nuancé considèrent pour leur part que l'intelligence artificielle n'est ni absurde ni inévitable : s'il n'existe à ce jour aucun programme informatique manifestant de l'« intelligence » au sens large, la question de savoir s'ils en sont capables est de nature expérimentale, et elle n'a pas encore été tranchée dans un sens ou dans l'autre.

Le débat sur l'intelligence artificielle s'articule autour des nombreuses définitions possibles de l'intelligence. À la fin des années 1940, Alan Turing, éminent mathématicien britannique, coureur de fond et contre-espion pendant la guerre, propose une approche intéressante selon laquelle l'intelligence est une sorte d'« expérience de pensée ». L'approche de Turing est opérationnelle : au lieu de définir l'intelligence, il décrit une situation dans laquelle un ordinateur pourrait la manifester. Le cas de figure qu'il choisit est similaire à l'activité décrite plus haut, l'idée étant d'avoir un interrogateur qui dialogue conjointement avec une personne et un ordinateur par le biais d'un téléimprimeur (le dernier cri de la technologie à la fin des années 1940 !). Si l'interrogateur ne peut pas les distinguer de façon sûre, alors l'ordinateur a réussi le test d'intelligence de Turing. Le téléimprimeur permet d'éviter que les caractéristiques physiques ou le ton de la voix ne trahissent l'ordinateur. On pourrait imaginer un prolongement de l'exercice où la machine devrait imiter l'être humain par l'apparence, le son, le contact, voire l'odeur. Mais ces attributs physiques ne relèvent sûrement pas de l'intelligence.

Le test de Turing était à l'origine un peu différent du nôtre. Comme exercice préliminaire, Turing propose un scénario où l'interrogateur pose des questions à un homme et une femme pour tenter de déterminer leur sexe. L'objectif de l'homme est de convaincre l'interrogateur qu'il est une femme, celui de la femme, de le convaincre qu'elle est bien une femme. Puis Turing imagine – ce n'est qu'une expérience de pensée – de remplacer l'une des deux parties par un ordinateur pour voir si ce dernier est aussi performant que l'être humain dans ce « jeu d'imitation ». Si nous avons modifié les règles pour cette activité de classe, c'est parce que les questions qu'un élève est susceptible de poser pour déterminer le sexe d'un individu ne seraient probablement pas convenables, et aussi parce que l'exercice original véhicule des stéréotypes liés au genre.

Imiter l'intelligence n'est pas une tâche facile. Si les rôles étaient inversés et qu'une personne essayait de se faire passer pour un ordinateur, elle échouerait sans doute, trahie par ses réponses trop lentes (et probablement inexactes) aux questions telles que « Combien font $123\,456 \times 789\,012$? ».

Mais, chose étonnante, il est assez facile pour un ordinateur d'acquérir un vernis d'aptitude conversationnelle. Un programme célèbre développé au début des années 1960, Eliza (mentionné plus haut), imite une psychothérapeute dialoguant avec son patient, selon l'approche rogéienne centrée sur la personne. À droite se trouve un exemple de conversation avec ce programme. Il réussit presque à dialoguer naturellement en suivant certains principes tels que :

- Générer des réponses toutes faites (par ex. : *Avez-vous un exemple précis en tête* en réponse aux phrases qui commencent par *Ils sont toujours...*, ou *Je suis navrée d'entendre que...* en réponse à *Il dit que...*)
- Répéter les affirmations de l'utilisateur (par ex.: *Votre petit ami vous fait venir ici* en réponse à *Mon petit ami m'a fait venir ici*)

Humain :	Les hommes sont tous les mêmes.
Ordinateur :	Comment cela ?
Humain :	Ils nous embêtent toujours pour un oui ou pour un non.
Ordinateur :	Avez-vous un exemple précis en tête ?
Humain :	Mon petit ami m'a fait venir ici.
Ordinateur :	Votre petit ami vous a fait venir ici ?
Humain :	Oui, il dit que je suis très souvent déprimée.
Ordinateur :	Je suis navrée d'entendre que vous êtes déprimée.
Humain :	C'est vrai, je suis malheureuse.
Ordinateur :	Pensez-vous que venir ici va vous aider à ne pas être malheureuse ?
Humain :	J'ai besoin d'aide, c'est certain.
Ordinateur :	Qu'est-ce que cela signifierait pour vous si vous obteniez de l'aide ?
Humain :	Peut-être que je pourrais réussir à m'entendre avec ma mère.
Ordinateur :	Parlez-moi de votre famille.
	...
Ordinateur :	Qu'est-ce qui vous vient à l'esprit quand vous pensez à votre père ?
Humain :	Des prises de tête.
Ordinateur :	Y a-t-il un rapport avec le fait que votre petit ami vous ait fait venir ici ?
	...

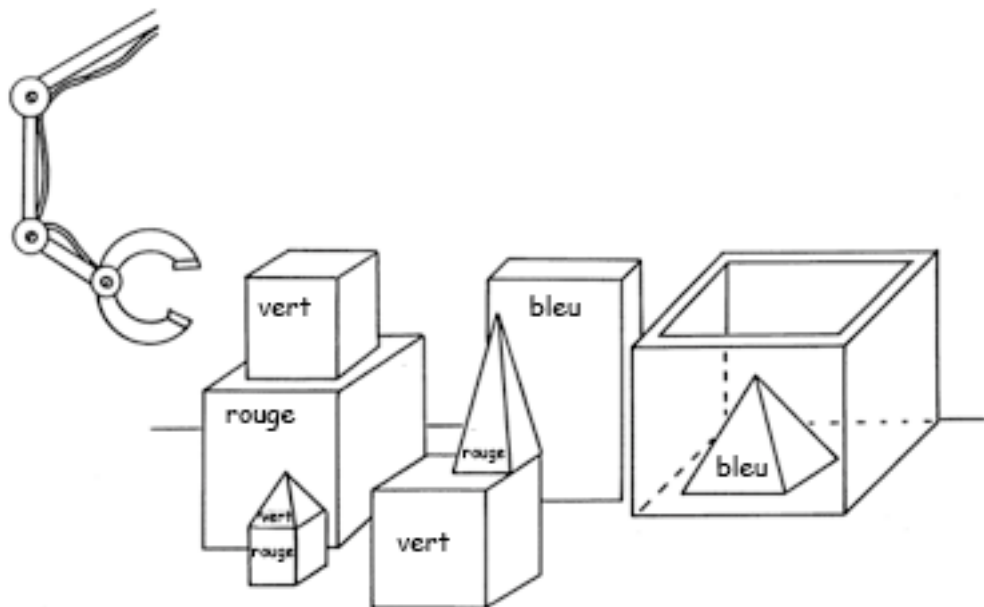
- Reconnaître certains mots clés (*Parlez-moi de votre famille...* en réponse aux phrases qui contiennent le mot *mère*)
- Utiliser des expressions toutes faites (*Qu'est-ce que cela signifierait pour vous si...*)
- Reprendre des questions soulevées plus tôt dans la conversation (*Y a-t-il un rapport avec le fait que...*)

Évidemment, cela ne signifie pas que l'ordinateur a réussi le test de Turing, car la personne ne pose pas des questions en vue de déterminer si son interlocuteur est une machine ou un être humain et, si elle le faisait, la vérité ne tarderait pas à se faire jour.

Une histoire très souvent citée, qui est peut-être douteuse mais fait désormais partie de la mythologie de l'intelligence artificielle, raconte que le vice-président des ventes d'une entreprise d'informatique se serait retrouvé sur un terminal normalement connecté au domicile d'une certaine personne, mais qui, cette fois-là, exécutait le programme du « psychothérapeute ». La conversation ci-contre montre ce qui s'est passé. Le vice-président s'est fait avoir, mais il est évident que la situation le prédisposait à croire qu'il parlait à un être humain. S'il avait eu un doute, il aurait vite découvert la vérité !

Un autre système qui paraît capable de tenir des conversations intelligentes est un programme appelé « SHRDLU ». Développé à la fin des années 1960, il sait répondre à des instructions en manipulant de petits objets posés sur une table. En réalité, les objets et la table ne sont qu'une image sur un écran d'ordinateur, mais il serait assez facile (du moins en principe) de passer de la simulation à la réalité à l'aide d'un robot. L'illustration suivante représente le petit monde du robot.

VP :	Pensez-vous que je peux utiliser l'ordinateur ce matin ?
Terminal :	Pourquoi posez-vous cette question ?
VP :	Je pourrais peut-être faire quelques ventes en télécommunications.
Terminal :	Pourquoi en êtes-vous certain ?
VP :	Mes clients potentiels n'ont jamais vu de démonstration du système.
Terminal :	Est-ce que cela signifie beaucoup pour vous ?
VP :	Bien sûr que oui.
Terminal :	Vous semblez bien optimiste.
VP :	C'est ridicule.
Terminal :	Que voulez-vous dire par ridicule ?
VP :	Y a-t-il une raison pour laquelle je ne pourrais pas utiliser l'ordinateur maintenant ?
Terminal :	Parlez-moi de vos clients potentiels.
VP :	Appelez-moi au 491 – 1850 SVP
	<i>Le vice-président oublie de mettre un point... l'ordinateur ne répond donc pas... le vice-président, furieux, appelle la personne à son domicile.</i>
VP :	Pourquoi me prenez-vous de haut ?
Humain :	Que voulez-vous dire par là ?



SHRDLU est capable de tenir des conversations qui semblent étonnamment naturelles. Le dialogue montre, entre autres choses, qu'il sait obéir à des ordres, comprendre des instructions, deviner à quoi fait référence un pronom et intégrer de nouveaux termes (« clocher »).

Humain : Prends un gros cube rouge.
Robot : OK.
Le système répond « OK » quand il exécute un ordre. Pour prendre le cube rouge, il doit le dégager en trouvant un autre espace où mettre le cube vert qui était au-dessus.
Humain : Attrape la pyramide.
Robot : Je ne sais pas de quelle pyramide vous parlez.
Il y a trois pyramides dans la scène, le programme ne fait rien.
Humain : Trouve un cube plus grand que celui que tu tiens dans la main et mets-le dans la boîte.
Le système doit comprendre à quoi font référence « celui » et « le » ; et utiliser un raisonnement déductif pour savoir quel cube il a dans la main.
Robot : Je suppose que « le » signifie le cube qui est plus grand que celui que je tiens.
Impression sur papier pour nous tenir informés.
Robot : OK.
La main doit poser le cube rouge pour prendre le cube bleu plus grand.
Humain : Que contient la boîte ?
Robot : La pyramide bleue et le cube bleu.
Humain : Sur quoi s'appuie la pyramide ?
Plus tôt, le système a répondu à « la pyramide » en disant : « Je ne sais pas de quelle pyramide vous parlez ». Maintenant, il sait que l'on parle de celle qui vient d'être citée.
Robot : La boîte.
Humain : Y a-t-il un clocher...
Robot : Désolé, je ne connais pas le mot « clocher ».
Le système interrompt immédiatement la conversation quand un mot qu'il ne comprend pas apparaît.
Humain : Un « clocher » est une pile qui contient deux cube cubes verts et une pyramide.
Les guillemets indiquent que nous définissons un nouveau mot.
Robot : Je comprends.
Humain : Y a-t-il des clochers ?
Robot : Non.
La nouvelle définition est utilisée en tant que description permettant de reconnaître les clochers.
Humain : Construis-en un.
Robot : OK.
La description donnée en définition lui permet de reconnaître un clocher, mais aussi d'en déduire un plan de construction. Le mot « un » signifie « un clocher », et le projet est mené à bien.

Le système est toutefois très fragile, et l'illusion qu'il crée s'effondre facilement.

Voici un prolongement de la conversation qui n'a pas été généré par le programme mais aurait certainement pu l'être. Bien qu'on ait pu définir le terme « clocher » durant le précédent dialogue, le robot ne peut intégrer qu'un très petit nombre de mots nouveaux parce que son intelligence est extrêmement limitée.

Finalement, il doit admettre qu'il ne comprend même pas les mots qu'il utilise (*Désolé, je ne connais pas le mot « désolé »*) !

Examinons à présent un autre prolongement possible du dialogue, complètement imaginaire cette fois-ci, et qui aurait pu provenir d'un élève. Bien que très court, cet extrait témoigne d'une vaste connaissance du monde.

Humain :	Je troque mon pistolet à eau contre le gros cube rouge.
Robot :	Désolé, je ne connais pas le mot « troque ».
Humain :	« Troquer » signifie échanger.
Robot :	Désolé, je ne connais pas le mot « échanger ».
Humain :	« Échanger » signifie donner à quelqu'un quelque chose contre autre chose.
Robot :	Désolé, je ne connais pas le mot « quelqu'un ».
Humain :	Désolé, je croyais que tu étais plus intelligent que tu ne l'es.
Robot :	Désolé, je ne connais pas le mot « désolé ».

Humain :	Je troque mon pistolet à eau contre le gros cube rouge.
Robot ?:	J'ai déjà un pistolet à eau, mais je t'échange deux cubes et une pyramide contre ta vieille grenouille toute gluante.

- Le « robot » comprend qu'un pistolet à eau a moins de valeur si l'on en possède déjà un.
- De plus, il *s'attend à ce que la personne le sache aussi*, puisqu'il ne prend pas la peine d'expliquer pourquoi il décline l'offre.
- Cependant, il voit que la personne a tellement envie du cube qu'elle est prête à se séparer d'un objet important.
- Il comprend aussi que, contrairement aux pistolets à eau, les cubes conservent leur valeur même quand on en a beaucoup.
- Il essaie de convaincre la personne en dévalorisant la grenouille.
- Cela implique qu'il est prêt à marchander.

Faire preuve d'une telle habileté est tout à fait impossible pour les ordinateurs actuels !

L'histoire ci-contre a été générée par un programme informatique plus récent (fin des années 1980). Bien que l'ordinateur soit rapidement trahi par la pauvreté et le caractère laconique de son récit, on peut aisément imaginer comment l'enjoliver en ajoutant toutes sortes de détails. Ce qui nous intéresse ici, ce ne sont pas les aspects superficiels de l'histoire, mais bien l'intrigue qu'elle incarne. Or, même si on est à mille lieux d'une intrigue imaginée par l'homme, elle réussit néanmoins à exprimer un conflit proprement humain. Il existe aujourd'hui un

certain nombre de systèmes capables de générer automatiquement des histoires. Pour les évaluer, tout le défi est de déterminer dans quelle mesure le matériau est constitué d'éléments standards assemblés, et dans quelle mesure il est une intrigue construite de façon créative, comme dans l'exemple ci-dessus.

Il était une fois une hirondelle de mer nommée Truman. Truman n'avait pas de maison. Truman avait besoin d'un nid. Il vola jusqu'au rivage. Truman chercha des brindilles. Truman ne trouva aucune brindille. Il vola vers la toundra. Il rencontra un ours polaire nommé Horace. Truman demanda à Horace où il y avait des brindilles. Horace cacha les brindilles. Horace dit à Truman qu'il y avait des brindilles sur l'iceberg. Truman vola jusqu'à l'iceberg. Il chercha des brindilles. Il ne trouva aucune brindille. Horace chercha de la viande. Il trouva de la viande. Il mangea Truman. Truman mourut.

Le *Prix Loebner* est une compétition annuelle où des programmes informatiques essaient de réussir le test de Turing en faisant croire au jury qu'ils sont humains. Aucun ordinateur n'a gagné la médaille d'or ou d'argent pour l'instant, celles-ci impliquant d'être capable de berner les juges tout du long. Mais une médaille de bronze est décernée chaque année à celui qui est jugé le plus humain. La première année (1991), le programme qui l'a remportée a fait des erreurs de frappe pour paraître plus humain !

Aucun système d'intelligence artificielle n'a jusqu'ici réussi à passer le test de Turing. De plus, bon nombre de philosophes affirment que le test ne mesure pas ce que l'on entend généralement par intelligence. Ce qu'il évalue, c'est l'équivalence comportementale : il est conçu pour déterminer si un programme informatique donné affiche les mêmes indicateurs d'intellect qu'un être humain, ce qui n'est peut-être pas la même chose qu'avoir véritablement de l'intelligence. Peut-on être humainement intelligent sans être conscient, sans être capable de ressentir de la gêne, sans faire l'expérience de l'amour, bref sans être...vivant ?

Le débat sur l'intelligence artificielle est loin d'être épuisé, il se prolongera sans doute pendant de nombreuses décennies.

Suggestions de lecture

L'Esprit dans la machine, les fondements de l'intelligence artificielle, du philosophe John Haugeland, est un ouvrage très agréable à lire sur le débat de l'intelligence artificielle. Certains exemples de ce livre en sont tirés (notamment les conversations de SHRDLU et les commentaires à leur propos).

Le test de Turing original est décrit par Alan Turing dans l'article « Computing machinery and intelligence », publié en 1950 dans la revue de philosophie *Mind*, puis dans le livre *Computer and thought* de Feigenbaum et Feldman, traduit en français « Les ordinateurs et l'intelligence », dans Turing A. M., Girard J.-Y., *La Machine de Turing*. L'article inclut les deux premières conversations.

Le programme du psychothérapeute est décrit dans l'article de J. Weizenbaum, « ELIZA—A computer program for the study of natural language communication between man and machine », publié en 1966 dans la revue d'informatique *Communications of the Association for Computing Machinery*.

Le programme du robot et de son petit monde de blocs est décrit dans la thèse de Terry Winograd, laquelle a ensuite été publiée dans un livre intitulé *Understanding natural language* (Academic Press, New York, 1972).

Le programme qui a généré l'histoire de Truman et Horace est décrit dans « A planning mechanism for generating story text », de Tony Smith et Ian Witten, publié en 1990 dans *Proceedings of the 10th International Conference on Computing and the Humanities*.