

La pensée informatique

par Jeannette M. Wing

Cet article fait suite aux divers interviews que nous avons faits et qui nous invitaient à une réflexion sur les fondements de notre discipline et ses aspects philosophiques et épistémologiques. Aujourd'hui l'article de Jeannette Wing nous conduit à réfléchir sur l'utilité et l'ubiquité de la pensée informatique et ses implications, mais aussi sur l'essence même de cette pensée.

Jeannette Wing est professeure d'informatique, sur une *chaire du président*, au département d'informatique de l'Université Carnegie Mellon. Elle est actuellement directrice adjointe à la direction des sciences et technologies des ordinateurs et de l'information à la fondation nationale des sciences des États-Unis, son titre exact en anglais est « *Assistant Director, Computer and Information Science and Engineering Directorate, National Science Foundation* ». Elle a accepté que nous traduisions et publions l'article *Computational Thinking* que l'on peut trouver dans les *Communications of the ACM* vol. 49, n° 3, Mars 2006, pp. 33-35 et nous l'en remercions. A noter que cet article a déjà été traduit en chinois et en espagnol. Je remercie Roger Mohr pour son aide et ses commentaires.

Pierre Lescanne

La pensée informatique est un ensemble d'attitudes et d'acquis universellement applicables que tous, et pas seulement les informaticiens, devraient apprendre et maîtriser.

NdT: alors que le français possède les mots: ordinateur, calcul, informatique, l'anglais n'offre que des traductions utilisant la racine « comput »: computer, computing, computer science, qui, tout en limitant la richesse de la langue, permettent un certain nombre de jeux de mots qui sont difficilement traduisibles. Le mot « thinking » pose lui aussi un problème et nous avons décidé de le traduire le plus souvent par « approche ». Nous avons donc décidé de traduire l'expression du titre de l'article, « *computational thinking* », qui revient fréquemment par l'expression française « *pensée informatique* » quand nous le traduisons par un substantif et « *penser informatiquement* » quand nous le traduisons par un infinitif en espérant être fidèle à la « pensée » de l'auteur.

La pensée informatique s'est élaborée à partir de l'analyse de la puissance et des limites des processus informatiques, qu'ils soient exécutés par des machines ou par des humains. En effet, c'est grâce aux méthodes et aux modèles de l'informatique que nous pouvons résoudre les problèmes et concevoir les systèmes qu'aucun d'entre nous ne serait capable d'appréhender tout seul. La pensée informatique nous confronte donc à l'énigme de l'intelligence de la machine. Qu'est-ce qu'un humain peut réaliser mieux qu'une machine ? Qu'est-ce qu'une machine peut réaliser mieux qu'un humain ? Plus généralement, qu'est-ce qui est calculable ? Aujourd'hui nous n'avons que des bribes de réponses à de telles questions.

La pensée informatique est un acquis fondamental, qui n'a pas à être limité aux informaticiens. Nous devrions l'inclure dans les savoirs fondamentaux que nous transmettons à nos enfants, qui s'ajouterait à la lecture, à l'écriture et à l'arithmétique. Alors que la presse écrite a permis la

diffusion des trois savoirs (lire-écrire-compter), le projet que nous proposons est incestueux, car ce sont les processus de calcul (*computing*) et les ordinateurs (*computers*) qui favorisent la diffusion de la pensée informatique (*computational thinking*).

La pensée informatique conduit à résoudre des problèmes, à concevoir des systèmes et à comprendre le comportement humain, en s'appuyant sur les concepts fondamentaux de la discipline et en y incluant une large collection d'outils intellectuels qui reflètent l'étendue de la science qu'est l'informatique.

Face à un problème donné, on peut se demander s'il est difficile et quelle est la meilleure façon de l'aborder. L'informatique en s'appuyant sur des fondations solides répondra précisément à ces deux questions. Affirmer la difficulté d'un problème prend alors en compte la puissance de la machine - c'est-à-dire du dispositif de calcul qui produira la solution - et doit alors considérer l'ensemble des instructions de la machine, ses contraintes de ressources et son environnement d'exploitation.

Pour résoudre un problème efficacement, on peut se demander si une solution approchante est assez bonne, si on peut utiliser l'aléatoire à son avantage et si des faux positifs ou des faux négatifs sont autorisés. Penser informatiquement, c'est reformuler un problème apparemment difficile en un problème que nous savons résoudre, par exemple par réduction, par plongement, par transformation ou par simulation.

Penser informatiquement, c'est penser récursivement ; c'est penser parallélisme ; c'est interpréter des données comme du code et du code comme des données ; c'est vérifier des types comme une généralisation de l'analyse dimensionnelle ; c'est reconnaître les vertus et les dangers des alias qui consistent à donner plus d'un nom à quelqu'un ou à quelque chose ; c'est reconnaître à la fois le coût et la puissance de l'adressage indirect et des appels de procédure ; c'est apprécier un programme non seulement pour sa correction et son efficacité, mais pour son esthétique et c'est évaluer la qualité de la conception d'un système par sa simplicité et son élégance.

Penser informatiquement, c'est utiliser l'abstraction et la décomposition quand il s'agit d'affronter une tâche large et complexe ou quand il s'agit de concevoir un système large et complexe ; c'est séparer les différents aspects ; c'est choisir une représentation appropriée pour un problème ou modéliser les aspects pertinents d'un problème pour les rendre accessibles ; c'est utiliser des invariants pour décrire succinctement et déclarativement le comportement d'un système ; c'est avoir confiance que l'on peut sans danger utiliser, modifier et influencer un système large et complexe sans en comprendre tous les détails ; c'est modulariser en anticipant de nombreux utilisateurs, ou stocker des informations en prévoyant des utilisations à venir.

Penser informatiquement c'est envisager le pire événement en pensant en terme de prévention, de protection et de récupération ; cela se fait par la redondance, le contrôle des dommages et la correction d'erreurs ; penser informatiquement c'est parler d'un interblocage quand il s'agit d'un embouteillage et parler d'interfaces quand il s'agit de contrats ; c'est éviter de faire jouer les coudes quand on peut synchroniser des rendez-vous.

Penser informatiquement c'est utiliser des raisonnements heuristiques pour trouver une solution ; c'est faire des plans, apprendre et ordonnancer en présence d'incertitude ; c'est chercher, chercher et chercher encore pour produire une liste de pages web, une stratégie gagnante ou un contreexemple ; c'est utiliser une masse de données pour accélérer un calcul ; c'est faire un compromis entre temps et espace et entre puissance de calcul et capacité de stockage.

Penser en informaticien signifie beaucoup plus que savoir programmer, cela requiert de penser à plusieurs niveaux d'abstraction

Examinons des exemples de la vie courante. Quand votre fille va à l'école, elle met dans son cartable les affaires dont elle a besoin, c'est du stockage prévisionnel d'informations. Quand votre fils perd ses gants et que vous lui suggérez de retourner sur ses pas, c'est du backtracking. Quand vous arrêtez de louer des skis et décidez d'en acheter, vous faites de l'algorithmique en ligne. Quand vous vous demandez quelle file choisir à la caisse du supermarché, vous modélisez les performances d'un système multi-serveur. Pourquoi votre téléphone continue-t-il à fonctionner pendant une coupure d'électricité? C'est parce que l'opérateur met en œuvre la tolérance aux fautes et la redondance de conception. Comment les tests de Turing publics et complètement automatisés arrivent-ils à discriminer les ordinateurs des humains ou pourquoi l'analyse de l'écriture manuscrite, dans les formulaires « CAPTCHA », est-elle un excellent moyen d'identifier un humain ? C'est parce que l'un et l'autre exploitent la difficulté qu'ont les agents informatiques pour résoudre les problèmes profonds de l'intelligence artificielle.

La pensée informatique sera intégrée à la vie de tous les jours, quand les termes « algorithme » et « précondition » feront partie du vocabulaire courant, quand le non déterminisme et le glanage de cellules (ou ramasse miette) auront chez Monsieur Tout-le-monde la signification courante des informaticiens et quand les arbres seront dessinés les branches en bas et les racines en haut.

Nous avons été témoins de l'influence de la pensée informatique sur les autres disciplines. Par exemple, l'apprentissage par ordinateur a transformé les statistiques. En effet, l'apprentissage statistique est aujourd'hui utilisé pour des problèmes d'un ordre de grandeur, en taille et en dimension, inimaginable il y a quelques années seulement. De fait, les départements de statistiques dans toutes sortes d'organisations embauchent des informaticiens et les facultés d'informatique absorbent les départements de statistique existants ou en création.

L'intérêt naissant des informaticiens pour la biologie a pour origine leur confiance dans le bénéfice que les biologistes peuvent tirer de la pensée informatique. Leur contribution va au delà de la capacité à rechercher des motifs dans de très grandes masses de données. L'espoir est que les structures de données et les algorithmes – nos abstractions informatiques et nos méthodes – puissent représenter la structure des protéines d'une manière qui permette d'en élucider la fonction. La bioinformatique est en train de changer la façon dont les biologistes pensent. De même, la théorie computationnelle des jeux est en train de changer la façon dont les économistes abordent leur discipline, la nanoinformatique la façon de penser des chimistes et l'informatique quantique celle de penser des physiciens.

Cette façon de pensée fera partie du bagage non seulement des autres scientifiques, mais aussi de tout un chacun. L'informatique ubiquitaire est aujourd'hui ce que la pensée informatique sera demain. L'informatique ubiquitaire qui était le rêve d'hier devient la réalité d'aujourd'hui et la pensée informatique sera la réalité de demain.

Ce qu'elle est et ce qu'elle n'est pas

L'informatique c'est l'étude du calcul, c'est-à-dire de ce qui peut être calculé et du comment le calculer ; elle confère à la pensée informatique les caractéristiques suivantes.

Conceptualiser n'est pas programmer. L'informatique n'est pas la programmation. Penser comme un informaticien va bien au delà de la capacité à programmer, cela requiert de penser à différents niveaux d'abstraction.

Le fondamental contre la routine. Un savoir faire fondamental est quelque chose que chacun doit maîtriser pour être performant dans une société moderne. Ironiquement, la pensée restera routinière tant que l'informatique n'aura pas résolu le grand défi de l'IA de faire penser les ordinateurs comme les humains.

C'est aux humains de penser, pas aux ordinateurs. La pensée informatique est un moyen pour les humains de résoudre des problèmes ; ça n'est pas de faire en sorte que les humains pensent comme les ordinateurs, car les ordinateurs sont barbant et ennuyeux, tandis que les humains sont malins et imaginatifs. Nous les humains rendons les ordinateurs passionnants et grâce à eux, nous utilisons notre intelligence pour nous confronter à des problèmes que nous aurions délaissé avant l'âge du calcul et pour construire des systèmes qui ne sont limités que par notre imagination.

Un complément à la pensée mathématique et à la pensée technologique. L'informatique s'appuie sur les mathématiques, étant donné que comme toutes les sciences, elle se fonde formellement sur elles, mais l'informatique s'appuie aussi sur la pensée technologique, étant donné que nous construisons des systèmes qui interagissent avec le monde réel. Les contraintes du calculateur sous-jacent obligent les informaticiens à penser en terme de calcul, c'est-à-dire informatiquement, et pas seulement mathématiquement, tandis que la liberté de construire un monde virtuel permet à nos ingénieurs de fabriquer des systèmes qui vont au delà du monde physique.

Des idées pas des artefacts. Présents partout et affectant nos vies quotidiennes, il n'y a pas que les artefacts logiciels et matériels que nous fabriquons, il y a aussi et surtout les concepts informatiques que nous utilisons pour affronter et résoudre des problèmes, organiser notre vie de tous les jours, communiquer et interagir avec les autres.

Pour tous et partout. La pensée informatique sera une réalité quand elle sera tellement intégrée dans l'aventure humaine et qu'elle disparaîtra en tant que philosophie explicite.

Beaucoup de gens assimilent l'informatique à la programmation et les parents dont les enfants sont étudiants en informatique n'entrevoient qu'un spectre étroit d'offres d'emploi. Beaucoup de gens pensent que la recherche fondamentale en informatique est terminée et qu'il ne reste plus que l'ingénierie. Au contraire, la pensée informatique est une perspective qui doit guider les éducateurs, les chercheurs et les praticiens en même temps que nous agissons pour changer l'image que la société a de notre discipline. En particulier, nous devons essentiellement toucher les lycéens ainsi que leur professeurs et leurs parents pour leur envoyer deux messages.

Des problèmes intellectuellement séduisants et scientifiquement stimulants restent à comprendre et à résoudre. Le champ des problèmes et de leur solution n'est limité que par notre curiosité et notre créativité.

Après des études en informatique on peut faire ce que l'on veut. Après des études en littérature ou en mathématiques on peut aborder toute sorte de carrières, il en est de même pour l'informatique. Après des études d'informatique on peut envisager une carrière en médecine, droit, commerce, science politique ainsi que dans tout type de science ou de technologie et même dans les arts.

Les enseignants d'informatique devraient créer des modules pour les étudiants de licence, et pas seulement dans les filières informatiques, qu'ils pourraient intituler «Les chemins de la pensée informatique ? ». Parallèlement, nous devrions présenter aux lycéens les méthodes et les modèles informatiques et plutôt que de nous lamenter sur le déclin des filières d'informatique et la baisse du soutien à la recherche dans ce domaine, nous devrions nous inspirer de l'intérêt du public pour l'aventure intellectuelle de notre discipline. Ainsi nous pourrions diffuser la joie, l'enthousiasme et le dynamisme de l'informatique, avec comme objectif: populariser la pensée informatique.