

France-IOI : l'apprentissage de l'algorithmique pour tous

Loïc Février, Mathias Hiron, Arthur Charguéraud

► **To cite this version:**

Loïc Février, Mathias Hiron, Arthur Charguéraud. France-IOI : l'apprentissage de l'algorithmique pour tous. Georges-Louis Baron, Éric Bruillard, Vassilis Komis. Sciences et technologies de l'information et de la communication en milieu éducatif: Analyse de pratiques et enjeux didactiques., Oct 2011, Patras, Grèce. Athènes: New Technologies Editions, pp.213-220, 2011, <ISBN : 978-960-6759-75-8>. <edutice-00676167>

HAL Id: edutice-00676167

<https://edutice.archives-ouvertes.fr/edutice-00676167>

Submitted on 3 Mar 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

France-IOI : l'apprentissage de l'algorithmique pour tous

Loïc Février¹, Mathias Hiron², Arthur Charguéraud²

loic.fevrier@france-ioi.org, mathias.hiron@gmail.com, arthur@france-ioi.org

¹ Équipe MOCAH, UPMC
² France-IOI

Résumé. Depuis 1996, la France participe aux Olympiades Internationales d'Informatique. L'association France-IOI, en charge de la préparation de l'équipe de France, a formé à la programmation et à l'algorithmique plusieurs centaines de collégiens et lycéens par l'intermédiaire de son site internet. Dans cet article, nous présenterons les Olympiades d'Informatique ainsi que l'association France-IOI et nous expliquerons les méthodes mises en œuvre pour enseigner à des jeunes la programmation et l'algorithmique de manière ludique.

Mots-clés: algorithmique, olympiade, apprentissage en ligne, ludique

Introduction

L'informatique fait partie des domaines que de nombreux jeunes découvrent et apprennent de façon autodidacte. Si certains aspects des sciences du numérique sont introduits petit à petit dans les programmes scolaires de nombreux pays, l'apprentissage de l'informatique continue à se faire en grande partie en tant que discipline extrascolaire.

Pour encourager les jeunes à s'initier à ces domaines et à continuer leur apprentissage, de nombreux concours ont été créés, à des échelles locales (Manev et al., 2008), nationales (Dicks et al., 2008, Combéfis & Leroy, 2011) ou internationales (Revilla et al., 2008 ; Van der Vegt, 2010 ; Haberman et al., 2011). Ces concours incitent non seulement les jeunes à se perfectionner, mais motivent également les adultes qui les organisent à développer des outils pour les faire découvrir au plus grand nombre (Clark, 2003, Verhoeff, 2010, Idlbi, 2009) et à permettre aux participants de se former. Des formations extrascolaires ont été mises en place (Zur et al., 2010, Italiani, 2011), souvent destinées à de petits groupes d'élèves sélectionnés, et dans certains cas à un large public (Kolstad & Piele, 2008, Charguéraud & Hiron, 2008).

Nous présentons ici comment la responsabilité de la sélection de l'équipe de France pour les Olympiades Internationales d'Informatique, l'un des plus prestigieux concours du domaine, nous a amené au fil des années à développer des outils et méthodes d'enseignement de la programmation et de l'algorithmique à destination d'un public bien au delà des élèves motivés par la participation à un concours.

De la préparation aux Olympiades à l'enseignement de l'algorithmique

Les Olympiades

Depuis 1959 avec la création et le succès de l'Olympiade Internationale de Mathématique (IMO), le modèle des olympiades internationales scientifiques a été repris dans de nombreux domaines et, en 1987, l'UNESCO a proposé la création des IOI (International Olympiad in Informatics), une compétition internationale de programmation et d'algorithmique destinée aux jeunes élèves du monde entier. L'objectif principal était de stimuler chez les jeunes l'intérêt pour l'informatique et son aspect scientifique, tout en permettant aux plus talentueux d'entre eux de se rencontrer et de partager des expériences scientifiques et culturelles. Depuis la première édition, organisée en Bulgarie en 1989, ce concours est organisé chaque année avec un succès grandissant, et regroupe maintenant plus de 80 pays, soit plus de 300 participants de tous les continents.

Lors des deux épreuves de 5 heures organisées au cours de la semaine que durent les IOI, chacun des participants, collégien ou lycéen (âgé de moins de 19 ans), doit résoudre 3 problèmes d'algorithmique. Pour résoudre un sujet, il faut trouver un algorithme efficace et le programmer en C, C++ ou Pascal. Le programme doit lire les données d'entrée du problème puis afficher le résultat attendu, le tout sans dépasser les contraintes de temps et de mémoire imposées. Un programme soumis est évalué automatiquement sur un ensemble de fichiers tests. Les premiers tests correspondent à des données de petite taille : ils permettent à un programme lent mais correct d'obtenir des points. Les derniers tests ne peuvent être traités qu'avec un programme ayant la complexité attendue. Ainsi, un programme bogué n'obtiendra aucun point tandis qu'un programme correct obtiendra un nombre de points qui dépend de son efficacité. À la fin des deux épreuves, les candidats sont classés et les meilleurs d'entre eux reçoivent des médailles d'or (premier douzième), d'argent (premier quart) ou de bronze (première moitié).

L'association France-IOI

La participation de la France à cette compétition a démarré modestement en 1996, avec une équipe de deux candidats sélectionnés à l'époque par le biais du concours français d'informatique Prologin, organisé depuis 1992 par des étudiants d'une école d'ingénierie informatique, EPITA. Les résultats assez faibles des premières participations ont mis en évidence que, en l'absence de cours de programmation ou d'algorithmique dans le secondaire, la simple sélection des meilleurs jeunes programmeurs autodidactes français ne suffirait probablement jamais à obtenir des résultats honorables à cette compétition de haut niveau. Malgré les solides bases en programmation dont faisaient preuve ces jeunes, la mise en place d'une formation supplémentaire en algorithmique s'avérait indispensable. La volonté d'obtenir de meilleurs résultats lors des Olympiades a donc amené les responsables de cette sélection à mettre en place petit à petit une véritable formation à la programmation et à l'algorithmique.

Cette formation a rapidement pris la forme d'un site web d'entraînement à l'algorithmique, mis en ligne en 2001 et sur lequel des exercices sont proposés, permettant aux candidats de valider leurs solutions par le biais d'un système d'évaluation automatique de leurs programmes. Le succès de ce système a encouragé les responsables à continuer leur effort et à créer en 2004, une association dédiée entièrement à l'enseignement de la programmation et l'algorithmique auprès des jeunes.

L'objectif initial d'obtenir des résultats corrects aux IOI a été largement atteint et les équipes sélectionnées obtiennent désormais des médailles chaque année, et se classent occasionnellement parmi les tous meilleurs mondiaux. Ainsi lors des IOI 2011, un candidat français formé sur cette plate-forme a obtenu une médaille d'or et la 9e place du classement mondial. Au total, les candidats français ont remporté 2 médailles d'or, 5 d'argent et 20 de bronze.

Au delà de l'obtention d'un bon classement lors d'un concours annuel, le véritable succès de cette aventure dépasse déjà largement le cadre d'une simple compétition. La plate-forme d'apprentissage (www.france-ioi.org) initialement créée pour la formation de quelques candidats aux IOI a évolué vers un outil qui permet aux jeunes francophones du monde entier de s'initier aux bases de la programmation et de découvrir petit à petit l'algorithmique, avec la possibilité de progresser jusqu'à un niveau très élevé, sans obstacle. Autour de ce projet se réunit un nombre grandissant de bénévoles passionnés par l'enseignement de ces domaines, qui ajoutent régulièrement de nouveaux exercices et cours. Le nombre d'utilisateurs augmentant, plusieurs centaines d'exercices sont désormais résolus chaque jour.

En parallèle avec la création de cours et d'exercices, et l'ajout de fonctionnalités (conseils automatisés, système d'entraide), l'expérience acquise via le site et les stages en présentiel organisés régulièrement permet aux entraîneurs de développer des techniques d'enseignement qui leur sont propres. En particulier, des méthodes permettent aux élèves d'améliorer rapidement leur capacité de réflexion sur les problèmes d'algorithmique les plus difficiles (Charguéraud & Hiron, 2008).

Enseigner l'algorithmique

Les bénévoles de France-IOI ne se fixent pas de limites définitives sur l'étendue de l'enseignement proposé sur le site d'entraînement, mais travaillent en priorité sur les notions et concepts considérés comme étant au programme des IOI (Forisek, 2008), ce qui couvre déjà un large champ de l'algorithmique.

Les concepts de base de la programmation que sont la notion d'instruction, de variable, de condition et de boucle sont depuis peu enseignés de la seconde à la terminale dans les lycées français, dans le cadre de l'enseignement des mathématiques. L'association y dédie de nombreux cours et exercices sur le site et porte une attention toute particulière à leur amélioration constante, avec comme objectif de proposer un contenu adapté à l'enseignement dans un cadre scolaire, aussi bien qu'à un apprentissage autonome des élèves dès le collège.

Une fois ces bases maîtrisées, le reste des concepts de la programmation est enseigné : types de données, fonctions, logique booléenne et opérateurs binaires, tableaux et structures. La priorité est avant tout donnée à la maîtrise de ces concepts et de leur utilisation, tandis que la présentation des aspects syntaxiques non indispensables à la parfaite maîtrise des concepts fondamentaux est repoussée au maximum. L'objectif n'est pas d'apprendre tous les détails d'un langage de programmation en particulier, mais d'être parfaitement à l'aise avec des principes de la programmation indépendants du langage.

Lorsque ces concepts ont été acquis et validés par la résolution des nombreux exercices associés, la formation se concentre sur les aspects purement algorithmiques. Les structures de données comme la pile, le tableau cumulatif, l'arbre binaire et bien d'autres sont découvertes au fil des exercices. Les algorithmes de graphes, tels que la recherche de plus court chemin ou la construction d'arbres couvrants minimaux sont présentés progressivement. Des techniques de balayages ou des algorithmes géométriques font également partie d'un programme très chargé mais que des élèves de lycée motivés sont capables d'acquiescer en un ou deux ans d'entraînement régulier sur le site.

Lors de cet apprentissage, la priorité est donnée non pas à l'acquisition de connaissances, mais au développement des capacités de raisonnement des élèves, et des méthodes de réflexion sont enseignées et pratiquées au fil des exercices (Charguéraud & Hiron, 2008).

Principes généraux et leur mise en œuvre pour un enseignement de l'algorithmique

La plate-forme de formation et les méthodes d'enseignement mises en place par l'association France-IOI reposent sur un ensemble de principes que nous allons rapidement énoncer.

Le premier correspond à un objectif général qui consiste à *apprendre à chercher*.

Un problème d'algorithmique est avant tout un problème intéressant, s'inspirant de la vie courante, que l'on a envie de résoudre : passer du temps à chercher la solution d'un exercice, faire des dessins, tourner le problème dans tous les sens, le reformuler, apprendre à le simplifier, et finalement trouver un algorithme et arriver à le programmer, voilà ce qui est important. Une fois que les élèves ont appris à chercher, ont découvert qu'ils étaient capables de trouver la solution de problèmes qui n'étaient pas évidents au premier abord, il est possible de leur faire découvrir par eux-mêmes la plupart des algorithmes dits « classiques », puis leur apprendre à les modifier, les combiner pour inventer de nouveaux algorithmes adaptés aux problèmes rencontrés.

Pour que l'apprentissage puisse être efficace, l'élève doit pouvoir avancer à son rythme, sans être bloqué, et pouvoir être aidé ; l'aspect ludique est également

important. Pour réellement progresser, la recherche personnelle est essentielle, mais le savoir découvert doit être formalisé, en particulier grâce aux corrections.

Ces principes ont guidé nos choix dans la mise en place progressive de notre site d'entraînement qui comprend des cours permettant de présenter la syntaxe de quelques langages (Caml et C/C++), syntaxe qui sera maîtrisée par la résolution de nombreux exercices de difficulté algorithmique progressive, allant du classique « Hello World » jusqu'au niveau des Olympiades, et au delà.

1. *s'amuser* : les problèmes sont proposés sous forme de petites histoires amusantes avec possibilité de résoudre les exemples à la main. Les élèves vont également évoluer selon des critères que l'on peut généralement retrouver dans des jeux d'aventure (expérience, autonomie, altruisme...). L'élève gagne des points à chaque exercice résolu.
2. *avancer à son rythme* : les solutions sont évaluées automatiquement et les accès aux cours et exercices se débloquent en fonction de la progression de l'élève, qui peut donc travailler au rythme qui lui convient.
3. *se faire aider* : un système de conseils automatiques a été mis en place et il est aussi possible de demander de l'aide aux élèves ayant déjà résolu un exercice, sous la surveillance des entraîneurs qui assurent la modération des messages échangés. Aider quelqu'un fait gagner des points, ce qui assure le bon fonctionnement du système d'entraide.
4. *avoir accès à une correction* : une bonne correction est l'occasion de présenter le cheminement permettant d'aller de l'énoncé du problème à une implémentation de la solution, ou une solution plus courte et plus simple que celle que l'élève avait utilisée ; il est plus important d'expliquer comment trouver la solution, que de montrer la solution elle-même. Les solutions constituent rapidement l'essentiel des cours, un algorithme n'étant expliqué qu'une fois trouvé par l'élève.
5. *rencontrer d'autres personnes* : des stages sont organisés, ce qui permet des interactions plus approfondies entre les élèves et avec les entraîneurs. Cela permet également d'enseigner ce qui n'est pas encore disponible sur le site et d'améliorer notre enseignement grâce aux retours des élèves.
6. *se comparer aux autres* : l'élève peut évaluer sa progression, voir ses points faibles grâce à des retours chiffrés. Ceux-ci permettent aussi de situer chaque élève par rapport aux autres et ainsi de favoriser l'émulation.

Les outils développés

Notre objectif est d'intéresser le maximum de personnes possible à l'algorithmique et à la programmation. Il doit donc être très facile de commencer à résoudre les premiers exercices et nous avons mis un certain nombre d'outils en place pour nous en assurer.

Installer un éditeur ainsi qu'un compilateur n'est pas une opération très difficile mais, même en les guidant, cette étape suffit à décourager de nombreux élèves qui n'ont pas encore acquis la motivation nécessaire. Nous avons donc choisi de mettre en place un éditeur intégré au navigateur, comportant les composants de base d'un bon éditeur (coloration syntaxique, indentation, copier/ coller, chercher/remplacer) sans être trop chargé. Cet éditeur permet donc d'écrire son programme directement depuis le site d'entraînement, de le tester sur les exemples de l'énoncé ou sur ses propres exemples, et enfin de le soumettre pour évaluation. Tous les programmes édités en ligne sont enregistrés dans la base de données et donc modifiables à tout moment depuis n'importe quel ordinateur. Cela permet également de résoudre les exercices depuis des ordinateurs de collèges ou lycées où l'installation de logiciels peut être difficile.

Il peut être difficile pour un débutant de bien comprendre les principes de l'exécution successive des instructions d'un programme, la modification des variables, les appels de fonctions... Il est bien entendu possible d'expliquer tout cela sous forme de texte, d'image animée ou de vidéo mais il est difficile de remplacer la personne qui, aux côtés de l'élève, va pouvoir lui expliquer le fonctionnement du programme. Il existe bien certains débogueurs ou IDE avec des modes d'exécution pas à pas mais ils sont souvent bien difficiles à installer et à utiliser pour un débutant. Nous avons donc développé et mis en place un mode pas à pas utilisable dans le navigateur. Celui-ci permet à l'élève, sur les codes sources présentés dans les cours ou corrections, d'exécuter le programme sur différentes entrées, et de voir au fur et à mesure l'évolution des variables, et de visualiser les appels de fonctions ainsi que le contenu de la pile. L'élève peut ainsi suivre en détail les embranchements, les boucles, les fonctions et les appels récursifs, et tout cela très simplement sans installer quoi que ce soit.

Quelques chiffres

La plate-forme de formation compte actuellement 15 000 inscrits dont 3 000 nouveaux chaque année. Environ 500 élèves sont actifs (ont soumis une solution) chaque mois et résolvent près de 500 exercices chaque jour. L'entraînement est basé sur les quelques 500 exercices créés spécialement par l'association et environ 1000 sujets issus de concours de programmation internationaux, qui sont donc évaluable automatiquement depuis le site internet. Le nombre d'exercices résolus n'est pas le point le plus important, mais les élèves partant aux Olympiades ont, en général, validé automatiquement entre 200 et 300 exercices et en ont résolus 100 à 150 autres sous la forme d'un pseudo-code. Un exercice prend en général entre 1 à 3 heures selon la difficulté et le niveau de l'élève.

L'association compte 13 membres-entraîneurs bénévoles gérant les aspects techniques (site internet, serveur d'évaluation...), créant les outils nécessaires, ajoutant des sujets et cours régulièrement. Quatre stages d'une semaine environ sont organisés, avec environ 8 élèves à chaque fois.

Perspectives

Au cours de ce document nous avons présenté la formation offerte à nos élèves. Celle-ci repose sur trois piliers principaux :

1. *la technique* : évaluation automatique, progression en autonomie...
2. *la pédagogie* : apprentissage ludique par la pratique, importance des corrections qui constituent la plus grande partie des cours ; interactions et entraide ; séances en présentiel...
3. *l'enseignement* : apprentissage progressif par les élèves d'une méthode générale de résolution des problèmes algorithmiques (Charguéraud & Hiron, 2008)

Tout en continuant à améliorer les cours proposés, à ajouter des exercices et à entraîner les meilleurs élèves pour les Olympiades, nous avons un certain nombre de projets qui se mettront en place durant l'année 2011/2012 et permettront de toucher un public encore plus large.

Actuellement, il est possible de résoudre les exercices en C/C++, OCaml, Pascal, Java, Python et nous proposons des cours de C/C++ et Caml. Un nouveau cours d'introduction à la programmation utilisant le langage Python est également en préparation.

En plus des sujets de programmation, nous allons mettre en place des sujets de type QCM pour valider certains acquis qui ne peuvent se vérifier facilement par l'écriture d'un programme. Nous souhaitons également introduire des sujets pour lesquels il faudra décrire de quelle manière on a construit sa solution et inclure le dessin réalisé. Ces sujets pourront être corrigés par d'autres élèves et permettront aux entraîneurs d'avoir un aperçu des méthodes de travail et du raisonnement des élèves, donc de corriger plus en amont certains défauts. Il est également prévu de mettre en place des sujets à sortie graphique où l'objectif n'est plus de calculer un résultat mais de générer une image ou une animation, par exemple le chemin à prendre pour sortir d'un labyrinthe, l'objectif étant de renforcer l'aspect ludique.

Dans le même esprit, nous envisageons de mettre en place un suivi automatique des élèves permettant de leur proposer régulièrement des exercices à résoudre adaptés aux difficultés qu'ils rencontrent, ainsi que de l'aide qu'ils n'auraient pas osé demander par eux-mêmes. Nous avons pu constater qu'un suivi régulier des élèves diminue nettement la probabilité qu'ils abandonnent en cours de route.

Nous sommes persuadés que notre méthode d'enseignement pourrait être adaptée avec succès à d'autres situations, à partir du moment où il est possible d'évaluer automatiquement (au moins partiellement) les solutions des élèves. En particulier, dans le cadre de l'apprentissage de l'informatique à l'université, ce système permettrait de proposer une formation plus individualisée. D'ores et déjà un projet de collaboration avec des enseignants de lycée est en train d'être mis en place afin de proposer sur notre plate-forme des exercices adaptés aux programmes de lycée

et leur permettre de suivre la progression de leurs élèves tout en profitant des outils mis en place et de la communauté actuelle.

Bibliographie

- Charguéraud, A., Hiron, M. (2008). *Teaching Algorithmics for Informatics Olympiads: The French Method*. Olympiads in Informatics 2, 48-63
http://www.mii.lt/olympiads_in_informatics/pdf/INFOL018.pdf
- Clark, D. (2003). *Using crossword puzzles to teach computing students*. Mathematics Competitions 16, no. 2, 41-49
- Combéfis, S., Leroy, D. (2011). *Belgian Olympiads in Informatics: The Story of Launching a National Contest*. Olympiads in Informatics 5, 131-139
- Dicks, K., Kubica, M., Stencel, K. (2008). *Polish Olympiad in Informatics - 14 Years of Experience*. Olympiads in Informatics 1, 50-56
- Forisek, M., (2008). *The International Olympiad in Informatics Syllabus*.
<http://people.ksp.sk/~misof/oi-syllabus/oi-syllabus.pdf>
- Haberman, B., Cohen, A., Dagiene, V. (2011). *The beaver contest: attracting youngsters to study computing*. ITiCSE 2011 : 378
- Idlbi, A. (2009). *Taking kids into programming (contests) with Scratch*. Olympiads in Informatics 3, 17-25
- Italiani, M. (2011). *Italian Olympiad in Informatics: 10 Years of the Selection and Education Process*. Olympiads in Informatics 5, 140-146
- Manev, K., Kelevedjiev, E., Kapralov, S. (2008). *Programming Contests for School Students in Bulgaria*. Olympiads in Informatics 1, 112-123
- Kolstad, R., Piele, D. (2008). *USA Computing Olympiad (USACO)*. Olympiads in Informatics 1
- Revilla, M.A., Manzoor, S., Lui, R. (2008). *Competitive Learning in Informatics : The UVa Online Judge Experience*. Olympiads in Informatics 2, 131-148
- Van der Vegt, W. (2010). *The CodeCup, an annual game programming competition*. Perspectives on computer science competitions for (high school) students
- Verhoeff, T.(2010). *An Enticing Environment for Programming*. Olympiads in Informatics 4, 134-141
- Zur, E., Benaya, T., Ginat, D. (2010). *IOI Israel - Team selection, Training, and Statistics*. Olympiads in Informatics 4, 105-114