

DES LOGICIELS POUR L'INITIATION À L'INFORMATIQUE

R. MORE

1. DIFFICULTÉS PÉDAGOGIQUES DE L'INITIATION À L'INFORMATIQUE

L'informatique est une discipline qui est enseignée à un public de plus en plus vaste.

Ce public est extrêmement disparate et manifeste des besoins très divers : les élèves des lycées et collèges, les étudiants mais aussi les professionnels en formation continue sui vent des enseignements d'informatique. Que l'on forme de futurs professionnels ou de simples utilisateurs de l'informatique, tous ces publics doivent passer par une étape obligatoire celle de l'initiation à l'informatique. Par initiation, on entend cette phase dans laquelle on ne s'en tient pas à un simple discours culturel sur l'informatique mais où l'on s'efforce de transmettre les idées de base qui permettent la compréhension de la technique informatique.

L'expérience montre que cette phase est très délicate pour les débutants à cause principalement de deux difficultés :

- l'ordinateur est un appareil qui ne ressemble à aucun autre, qui fait intervenir des notions tout à fait nouvelles pour le débutant (notions de mémoire, de calcul automatique, etc.) Celui-ci a beaucoup de difficultés à se représenter un système aussi différent de ceux qu'il tonnait déjà.
- lorsqu'on réalise les premiers programmes, on est placé devant une certaine "opacité" de la machine : l'utilisateur introduit son programme, l'ordinateur demande les données, puis affiche les résultats. La phase intermédiaire de calcul est entièrement cachée.

Pour apporter une réponse à ces deux difficultés et promouvoir une méthode active d'initiation, nous avons conçu et réalisé des logiciels

favorisant l'activité des élèves et faisant un large appel à la représentation graphique.

Nous présentons dans cet article les caractéristiques principales de notre démarche et aussi les deux derniers logiciels . que nous venons de réaliser : les systèmes MAP et BAP dur APPLE II.

2. MÉTHODE PÉDAGOGIQUE UTILISÉE

Le contenu de l'initiation

Après avoir exposé quelques généralités (définition de l'informatique, domaine d'application), il est intéressant d'enseigner des éléments de numération binaire : ceci permet aux débutants d'avoir une première idée de la représentation et du traitement des informations.

La deuxième étape apporte une description fonctionnelle de l'ordinateur puis l'étude des schémas fondamentaux de la programmation : les algorithmes séquentiels, conditionnels, répétitifs.

La dernière étape concerne la notion de fichier : les fichiers à accès séquentiel ou à accès direct sont abordés. Les opérations de base (création, consultation, mise à jour) sont présentées, et des applications simples peuvent être expliquées. L'expérience pédagogique montre que c'est la deuxième étape, celle des algorithmes, qui pose aux débutants le plus gros problème de compréhension. Par ailleurs, les notions exposées, les algorithmes, sont les notions-clé de l'informatique : si elles ne sont pas comprises, il n'est pas possible d'aller plus loin avec un réel profit.

Méthode pédagogique

Notre enseignement repose sur les principes suivants - simplicité et progressivité des notions exposées, – activité des élèves pour la maîtrise des notions de base.

simplicité-et-progressivité

Plutôt que de présenter un langage puissant mais complexe (tel Pascal), nous avons choisi de commencer avec une machine élémentaire, MAP ou un langage très simple, BAP. Les algorithmes de base sont présentés et programmés en MAP ou en BAP. Les calculs faits par les programmes portent exclusivement sur les nombres arithmétiques, qui sont les plus concrets et les plus familiers aux élèves. Les notions

introduites ne nécessitent aucune connaissance préalable ni en informatique, ni en mathématique.

L'activité de l'élève

L'activité personnelle des élèves est à la fois indispensable et facile à réaliser en informatique. C'est en réalisant eux-mêmes des algorithmes, même très simples que les débutants comprennent les notions de base.

Les systèmes MAP et BAP étant très simples, les élèves peuvent très vite réaliser, sur le papier, leurs premiers programmes. Leur intérêt s'accroît beaucoup si, de plus, ils peuvent ensuite réellement faire exécuter leurs programmes.

Nous avons donc conçu et réalisé des logiciels permettant :

- l'introduction et l'exécution des programmes
- la représentation graphique de l'exécution des programmes,
- l'évaluation automatique des programmes.

L'activité de l'élève est fortement encouragée : il introduit son programme, il le voit s'exécuter, le système lui fournit une évaluation de sa solution, qu'il peut, s'il le désire, tenter d'améliorer.

3. DESCRIPTION DES LOGICIELS. Premiers logiciels réalisés

Depuis plusieurs années nous utilisons un logiciel appelé MAP (Machine Arithmétique Pédagogique). Le langage MAP est un langage d'assemblage, très simplifié. Il permet aux débutants de commencer très vite à programmer et d'avoir une première représentation de l'ordinateur.

Cette utilisation d'une machine élémentaire n'était pas originale-beaucoup d'enseignants l'ont utilisée en initiation. Elle rend cependant de grands services aux débutants.

Nous y avons apporté deux améliorations :

- en 1982, nous avons mis au point un système d'évaluation automatique des programmes. Notre système permettait alors à un élève d'introduire un programme MAP puis de demander une "validation". Le système lui indiquait si son programme était juste ou faux et, dans le premier cas, lui attribuait une note. Ce, logiciel

était réalisé sur un ordinateur SOLAR 16/65, avec lecteur de cartes et imprimante. Utilisé avec plusieurs centaines d'étudiants, ce logiciel soulageait le travail de correction des enseignants et renforçait la motivation des étudiants.

- en 1983, nous avons réalisé une simulation graphique du système MAP. Sur des écrans Télévidéo connectés à l'ordinateur SOLAR 16-65 de l'Université de Toulouse-Le-Mirail, nous avons reproduit le logiciel MAP : lorsqu'un étudiant a terminé la saisie d'un programme MAP, il peut en demander l'exécution graphique.

Il voit alors s'exécuter de manière globale ou en "pas à pas" l'ensemble de son programme. La totalité du système est représentée sur l'écran et dès qu'un élément change de valeur la modification est représentée. Ce système utilisé en formation initiale ou continue a eu beaucoup de succès auprès de nos étudiants.

Ce sont ces deux idées de base, représentation graphique et évaluation automatique que nous avons utilisées pour la conception et la réalisation de nos logiciels actuels.

Les logiciels actuels

MAP est, comme nous l'avons déjà indiqué, un langage d'assemblage simplifié, tandis que BAP est une version simplifiée du langage BASIC.

Ces deux logiciels sont réalisés sur le micro-ordinateur ,APPLE II et chacun d'eux comprend

- un système de saisie permettant l'entrée des programmes,
- un système d'exécution graphique des programmes,
- un système d'évaluation (en fait de validation puis d'évaluation) des programmes.

Ces logiciels ont des objectifs limités mais importants ils permettent la mise en œuvre des trois schémas fondamentaux de la programmation

- les programmes séquentiels,
- les programmes conditionnels,
- les programmes répétitifs.

Le système MAP

a) présentation des programmes

MAP permet de réaliser les algorithmes de calcul élémentaire. Les programmes fonctionnent seulement sur des données arithmétiques et fournissent des résultats arithmétiques.

MAP impose au programmeur de fixer lui-même l'implantation de son programme en mémoire, de même que de choisir ses mémoires de travail : ainsi le programmeur doit connaître la "géographie" de son programme.

Donnons le programme correspondant à l'énoncé : Lire 2 nombres puis calculer et imprimer leur somme :

- 000 LI 100
- 001 LI 101
- 002 CH 100
- 003 AJ 101
- 004 RA 102
- 005 EC 102
- 006 ST 000
- 100 RS 003

Ce programme réalise les actions suivantes :

- lecture des 2 données dans les mots-mémoire d'adresse 100 et 101.
- chargement dans un accumulateur de la première donnée addition de la deuxième, stockage du résultat dans la mémoire 102 ; - impression du contenu de 102, c'est-à-dire du résultat et arrêt.

b) mise en œuvre du programme

Il y a trois phases principales :

- une phase de saisie : l'utilisateur peut introduire son programme, effacer, lister, etc.
- une phase d'exécution graphique : le programme apparaît dans la mémoire centrale avec ses mémoires de travail, l'utilisateur peut commander une exécution globale, d'une seule traite, ou, pour le voir de manière plus précise et plus détaillée, une exécution en "pas à pas", c'est-à-dire instruction par instruction : chaque action

du programme est représentée, l'utilisateur peut suivre l'évolution des mots-mémoire ou de l'accumulateur. Chaque fois qu'un élément du programme va changer d'état, il est signalé par un clignotement ;

- une phase d'évaluation automatique : lorsque l'utilisateur estime que son programme est correct, après une ou plusieurs phases de saisie et d'exécution, il peut demander une validation. Notre système va rechercher les informations nécessaires sur le programme de l'utilisateur puis va vérifier que ce programme
 - s'exécute normalement,
 - fonctionne de manière correcte au niveau des entrées-sorties (lecture et écriture),
 - fournit des résultats justes.

Si ces résultats sont justes, le système attribue une note sur 20, en fonction de l'efficacité du programme.

Le système BAP

a) présentation des programmes

BAP signifie Basic Arithmétique pédagogique. C'est donc un sous-système du système BASIC destiné à l'initiation à la programmation. Il a 2 caractéristiques principales par rapport au Basic

- les programmes peuvent être écrits avec des mots de base anglais ou français.
- il ne permet d'utiliser que des variables simples (non dimensionnées) et entières.

Donnons le programme correspondant à l'énoncé Lire 10 nombres, calculer et imprimer leur somme

- 10 S = 0
- 20 POUR I =1 A 10
- 30 LIRE N
- 40 S = S + N
- 50 PROCHAIN I
- 60 ÉCRIRE S
- 70 FIN

b) mise en œuvre des programmes

Il y a là aussi, 3 phases principales :

- une phase de saisie,
- une phase d'exécution graphique : le programme apparaît dans la mémoire centrale. Les variables utilisées apparaissent aussi, de même, qu'un emplacement de calcul où l'on pourra suivre l'évaluation des expressions arithmétiques.

L'utilisateur peut commander une exécution globale le programme s'exécute complètement et l'utilisateur peut avoir une première idée de la validité de son programme. Il peut, s'il le désire, commander une exécution en "pas à pas" : dans ce cas, le système exécute chaque fois une instruction ou une opération, il est ainsi possible de suivre en détail l'exécution du programme. Tous les états du programme sont représentés : l'instruction en cours est signalée par un effet de contraste vidéo et dès qu'une variable va changer de contenu, son emplacement se met à clignoter.

- une phase d'évaluation automatique : comme dans le système précédent, lorsque l'utilisateur le juge utile, il peut demander une validation. Le système lui fournira un diagnostic et le cas échéant une note.

Remarque

Chacun des 2 systèmes MAP et BAP permet de travailler de deux façons :

- de manière libre : l'utilisateur introduit un programme quelconque. Dans ce cas, seules les phases de saisie et d'exécution graphique ont lieu. Il n'y a pas de validation.
- d'après une liste d'énoncés de problèmes à programmer l'énoncé est trouvé sur une brochure associée au didacticiel et est ensuite affiché sur l'écran. Les 3 phases doivent ici être réalisées : saisie, exécution, validation.

4. MÉTHODES ET TECHNIQUES UTILISÉES

Les programmes de simulation

Le langage MAP est très simple à simuler. L'écriture d'un simulateur MAP est un sujet maintes fois proposé à des étudiants de deuxième année de D.E.U.G.

Pour le langage BAP nous avons réalisé un interpréteur complet en utilisant les méthodes classiques de la traduction des langages de programmation.

L'exécution graphique

Elle ne pose pas de problèmes particuliers. Elle consiste simplement à insérer dans le simulateur (MAP ou BAP), l'appel de procédures graphiques (gestion du curseur, passage en inverse vidéo commande de "flash") permettant la représentation des actions du programme.

L'évaluation automatique

- Principe de l'évaluation

Un programme est une suite de calculs qui associe à un ensemble de données, un ensemble de résultats : chaque valeur lue par le programme est une donnée, chaque valeur écrite est un résultat.

Nous sommes partis de l'idée suivante : un programme est juste si, quelles que soient les valeurs des données, il calcule correctement des résultats

Nous avons considéré qu'il y avait deux types de données :

- les données régulières : ce sont celles qui ne posent pas de problème. L'application de l'algorithme général donne les bons résultats,
- les données singulières : ce sont les valeurs particulières pour lesquelles l'algorithme général ne fonctionne plus.

Le principe de notre méthode est le suivant

- pour un problème donné, nous disposons sur un fichier de l'ensemble des jeux de données et de résultats. Si pour tous les jeux de données, le programme de l'utilisateur fournit les résultats attendus, il est déclaré juste, sinon il est faux.

Donnons un exemple : pour valider un programme de calcul de factorielle, nous le ferons exécuter sur les valeurs de données 5, 1 et 0. Si les résultats sont respectivement 120, 1 et 1 le programme est déclaré juste.

Attribution d'une note au programme : celle-ci est fonction du produit $P \times Q$, P étant une estimation de la place occupée en mémoire, Q étant le nombre d'instructions (ou d'opérations) exécutées. Lorsque le programme a été reconnu juste, la note attribuée est inversement proportionnelle au produit $P \times Q$ du programme utilisateur, la note 20 correspondant à la valeur minimale du produit $P \times Q$, conservée dans le fichier des problèmes.

Remarque

Ces définitions et ces règles ne sont applicables que dans un contexte réduit et bien précis de la programmation, ce qui est tout à fait le cas de l'initiation à l'informatique.

Caractéristiques techniques et état d'avancement

Les deux systèmes MAP et BAP sont écrits en BASIC Apple-soft sur le micro-ordinateur APPLE II. Nous utilisons des versions compilées de ces deux logiciels. Les deux systèmes dans leur état actuel ont été présentés à Toulouse, au mois de février 1985, au salon SIBSO - 85.

Le système MAP

MAP est un programme d'environ 1200 instructions BASIC. Après compilation il occupe 30.000 octets. Ce système est entièrement opérationnel et nous venons de commencer à l'utiliser en formation permanente.

Le système BAP

Sur les trois phases, simulation, représentation graphique et validation, les deux premières sont opérationnelles, la troisième est en cours de réalisation.

Sans son état actuel, le programme BAP compte environ 1000 instructions.

5. CONCLUSION

Nous n'avons pas encore mis en place une procédure systématique et expérimentale pour mesurer l'efficacité pédagogique de nos logiciels.

Nous avons cependant utilisé de manière courante depuis deux ans le système MAP graphique sur SOLAR 16, et le système MAP avec évaluation automatique.

Le système MAP graphique employé avec des débutants en formation initiale et continue permet dès maintenant les observations suivantes : il est un très bon moyen de mise en oeuvre des premiers algorithmes et suscite l'intérêt et la satisfaction de tous les débutants.

Le logiciel de validation automatique MAP sur SOLAR 16 a été utilisé avec deux promotions d'étudiants de première année de D.E.U.G. MASS. Avec cette population très motivée, nous avons chaque fois observé l'acharnement que mettaient les étudiants pour obtenir la note maximale.

Dans une pédagogie de l'informatique où l'activité de l'élève joue un rôle très important, nous pensons que ces idées de base sont fondamentales : la simulation qui permet l'activité de l'élève, la représentation graphique qui montre et clarifie, l'évaluation automatique qui favorise l'entraînement individuel. Ces idées, qui se sont déjà montrées efficaces dans le contexte de l'Université, devraient, a fortiori, rendre des services lors de l'entrée de l'informatique dans les collèges et les lycées.

par R. MORE (1)

avec la collaboration de

G. LAMOUREUX (2) et M. P. MARTIN (3)) pour la conception

et

P. AURIAC, Th. CABALLE,

J. Ch. DUCASSE

J.L. VIANES pour la réalisation (4)

(1) Maître assistant en informatique à l'Université de Toulouse-Le Mirail

(2) Maître assistant en psychologie à l'Université de Toulouse-Le Mirail

(3) Analyste informatique à l'université de Toulouse-Le Mirail.

(4) Étudiants de 1^{er} et 2^{ème} cycles.