

EDUCATION & INFORMATIQUE

n° 14 • Janvier-Février 1983 • bimestriel • 25 F



Evariste Galois

1811-1832

DOSSIER

**LES
MATHÉMATIQUES**

**— E.A.O. — PROGRAMMES —
TECHNIQUE : LA TORTUE —**

**LA MICRO VIENT D'ENTRER A L'ÉCOLE.
MICRO 7 VIENT DE SORTIR.**



Un micro-ordinateur, ça coûte trois fois moins cher qu'une 125 et ça va beaucoup plus vite. En plus, on peut apprendre les maths ou l'anglais. Aujourd'hui, ils sont des milliers à l'avoir compris.

Pour ne pas vous laisser dépasser, lisez Micro 7, le magazine mensuel qui explique la micro-informatique à ceux qui foncent.

Micro 7. La micro-informatique expliquée à ceux qui foncent.

Leçon 9

André Deledicq

STRUCTURER

9-1 Un programme principal court :

Après avoir appris la syntaxe des différentes instructions acceptées par une machine dans un certain langage, on croit souvent que la programmation n'est plus qu'un tricotage patient consistant à aligner des instructions numéro après numéro.

Or, écrire les instructions les unes après les autres de la première à la dernière est une très mauvaise méthode de programmation, on aboutit alors souvent (surtout après quelques améliorations) à un style de programme dénommé "style spaghetti" ; en effet, les sauts, conditionnels ou non, ont une fâcheuse tendance à se chevaucher les uns les autres de sorte que même l'auteur du programme perd le fil de son Ariane dans un inextricable lacis d'instructions.

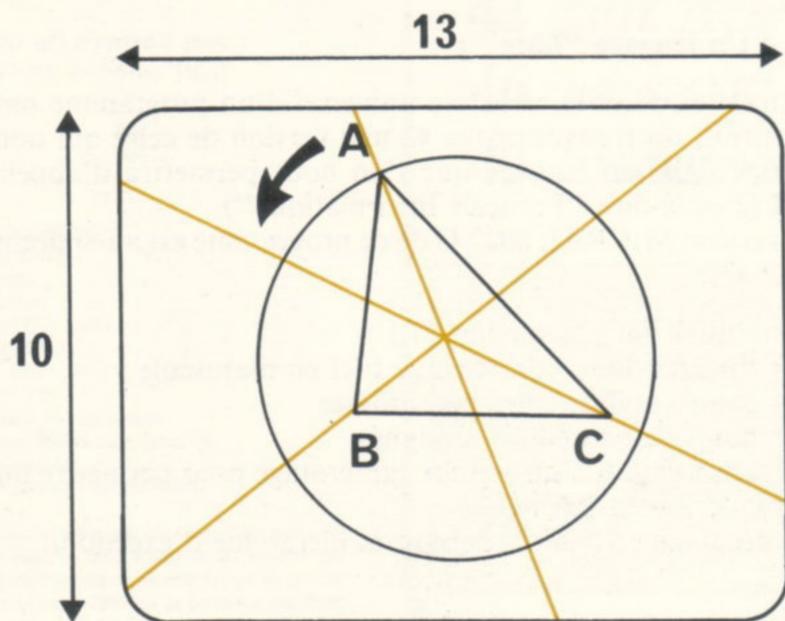
① Une meilleure méthode consiste à ne jamais écrire (en tout cas pour les débutants) de bloc de plus d'une douzaine d'instructions.

Cela n'est évidemment possible que si l'on peut utiliser ce que l'on pourrait appeler des "macro-instructions" dont le détail descriptif est renvoyé en annexe : un programme est alors constitué de quelques macro-instructions, elle-même décrites par une douzaine au plus d'instructions...

On aboutit ainsi à un programme, dont chaque bloc est aisément accessible à l'analyse ; par la même occasion, on réalise ainsi des programmes naturellement transportables sur d'autres machines ou d'autres langages puisque les modifications alors nécessaires sont facilement localisées. L'outil de base de cette méthode de programmation est le sous-programme.

En guise d'exemple, voici un morceau de programme BASIC, réalisant un petit dessin animé géométrique sur un MICRAL 80 22G.

9-2 Exemple :



```

Menu
1 TEXT:'MENU'
2 PRINT'VOUS ALLEZ VOIR SOIT LES HAUTEURS'
3 PRINT'SOIT LES MEDIANES , SOIT LES MEDIATRICES'
4 PRINT'D'UN TRIANGLE VARIABLE'
5 PRINT
6 PRINT'QUE VOULEZ VOUS VOIR ?          FRAPPEZ:'
7 PRINT
8 PRINT'  1  POUR LES HAUTEURS'
9 PRINT'  2  POUR LES MEDIANES
10 PRINT'  3  POUR LES MEDIATRICES'
15 INPUT R
19 '
20 'LES ANGLES SONT EN RADIANES
21 DATA 3.14,0.1
22 READ PI,H
25 'POINTS B ET C
26 DATA -2,-1,3,-1
27 READ XB,YB,XC,YC
29
30 'DESSIN
40 FOR I=PI/2 TO 2*PI+PI/2 STEP H
50 XA=4*COS(I):YA=4*SIN(I)
51 HGR
60 GOSUB100:'DESSIN DU TRIANGLE
70 ON R GOSUB 200,300,400:'DESSIN
80 FOR K=1 TO 3000:NEXT K
90 NEXT I
99 END

Programme principal
100 'DESSIN DU TRIANGLE
101 .....
199 RETURN
200 'DESSIN DES 3 HAUTEURS
201 .....
299 RETURN
300 'DESSIN DES 3 MEDIANES
301 .....
399 RETURN
400 'DESSIN DES 3 MEDIATRICES
401 .....
499 RETURN
    
```

Remarques :

- Il est commode de rentrer en données des constantes classiques ou non classiques (exemple : dans les instructions 22 et 27)
- Le point A, de coordonnées XA, YA tourne sur un cercle de centre (0,0) de rayon 4 (d'environ 6° entre chaque position).
- HGR et TEXT efface les pages "graphique" et "texte"
- GOSUB 100 envoie à l'instruction 100 ; puis le programme reprendra à l'instruction suivant GOSUB 100 dès que l'exécution rencontrera l'instruction RETURN
- L'instruction ON R GOSUB 200, 300, 400 est un "aiguillage", très pratique, envoyant à l'instruction 200 si R vaut 1, à l'instruction 300 si R vaut 2, et à l'instruction 400 si R vaut 3.
- Les remarques "REM" sont évitées dans les BASIC (et plus généralement les langages) acceptant le renvoi à des étiquettes. Par exemple, on écrirait sur une HP 9845 :

60 GOSUB Triangle

.....

100 Triangle :

.....

199 RETURN

- Le remplissage des blocs [100, 199] [200, 299] [300, 399] et [400, 499] peut attendre le paragraphe suivant.

② Vous avez certainement entendu dire quelque part que l'avantage (et par suite la raison d'être) des sous-programmes consistait en la possibilité d'être "appelés" de plusieurs endroits successifs à partir d'un programme principal ; on éviterait ainsi d'écrire plusieurs fois la même suite d'instructions. **RIEN N'EST PLUS GRAVEMENT TROMPEUR !** S'il est vrai que cet avantage existe, il apparaît comme dérisoire par rapport à la simplification de pensée engendrée par la possibilité d'écrire des blocs s'emboîtant les uns dans les autres et s'appelant hiérarchiquement.

9 - 3 Des sous-programmes indépendants

On peut, en effet, considérer que le programme désiré ci-dessus est complètement écrit. Le fait que les blocs comme [100, 199] ne le soient pas encore n'est vraiment qu'un détail. Le programmeur peut aussi bien aller prendre un café ou un week end prolongé : l'écriture de ces blocs est complètement indépendante de celle du problème principal.

Par exemple, pour le bloc [100, 199], on écrira :

```
110 LINE (XA,YA) - (XB,YB), PSET
120 LINE (XB,YB) - (XC,YC), PSET
130 LINE (XC,YC) - (XA,YA), PSET
```

En fait, on voit ici que les valeurs de XA, YA, XB, YB, XC, YC sont "empruntées" au programme principal ; l'indépendance est donc restreinte par la nécessité de **passer la valeur de certains paramètres** du programme principal vers le sous-programme. Il serait bon de pouvoir expliciter ce passage (et ce serait même beaucoup plus pratique !).

Or, les appels de sous-programmes avec passage de paramètres n'existent pas dans les basics standards. Par contre, on les trouve en LSE sous le nom de "procédure". La séquence d'appel et le bloc appelé se présentent alors par exemple ainsi :

```
60 TRIANGLE (XA, YA, XB, YB, XC, YC)
```

```
100 PROCÉDURE & TRIANGLE (U1, V1, U2, V2, U3, V3)
```

```
110 Tracé du segment (U1, V1) - (U2, V2)
```

```
120 Tracé du segment (U2, V2) - (U3, V3)
```

```
130 Tracé du segment (U3, V3)-(U1, V1)
```

```
199 RETOUR
```

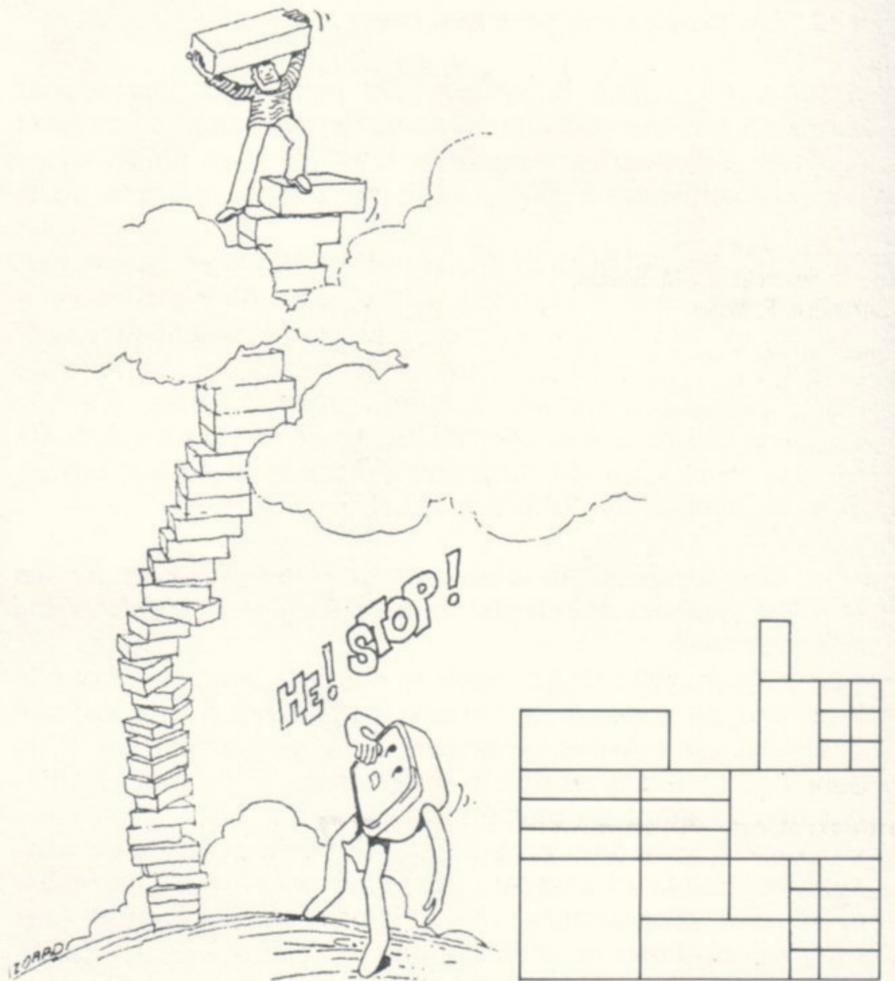
Remarque : Lors de l'appel de la "procédure" & TRIANGLE, U1 prend la valeur XA, V1 prend la valeur YA, ainsi de suite dans l'ordre d'écriture des paramètres, les valeurs "passées" pouvant être des "données" ou des "résultats" pour le sous-programme.

L'intérêt de ce passage de valeurs apparaîtra encore plus manifeste dans les blocs de tracés des hauteurs, des médianes ou des médiatrices : Il sera alors fait systématiquement appel à un sous-programme de tracé d'une droite passant par un point (de coordonnées P et Q) et parallèle à un vecteur (de composantes U et V) ; il suffira, dans chacun des cas, de calculer ce point et ce vecteur avant l'appel du sous-programme.

Il y a enfin une dernière difficulté, déjà signalée dans la leçon 8, et relative au système de coordonnées : la transformation des "coordonnées-utilisateurs" en "coordonnées-écran" justifie encore une fois l'appel à un sous-programme avant chaque tracé. (Et lorsqu'on passera d'un type d'écran à un autre

type, il suffira de modifier ce sous-programme et celui-là seulement !).

On voit donc que l'imbrication et la sériation des blocs les uns dans les autres atteignent rapidement une sophistication qui rend dangereuse toute construction fondée sur l'empilement d'instructions.



A gauche, la programmation suicide, instruction par instruction.
A droite, la programmation structurée, bloc dans/après bloc.

9 - 4 Un langage "libre" :

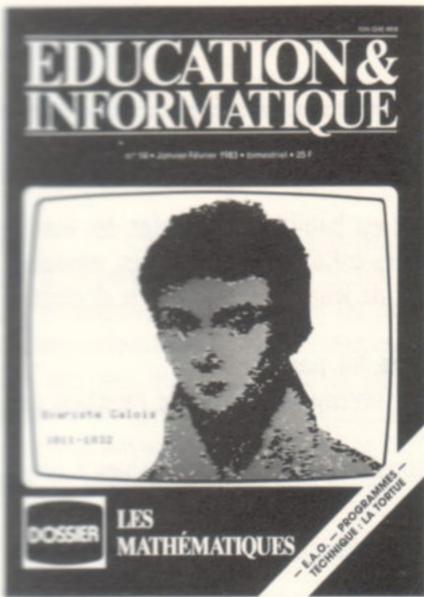
Afin de montrer le caractère universel d'un programme bien construit, on trouvera page 42 une version de celui qui nous occupe dans un langage que l'on nous permettra d'appeler **FRI** (c'est-à-dire "Français Informatique").

La version MICRAL 8022 G de ce programme est aussi donnée page 42.

Remarquez dans la version FRI :

- les "instructions" du langage FRI en majuscule
- les commentaires entre parenthèses
- les noms de procédure soulignés
- la conservation d'un certain numérotage pour permettre une traduction en BASIC
- le décalage vertical traduisant la hiérarchie d'exécution

Sommaire



N° 14 janvier/février
1983

L'illustration de couverture
a été réalisée par
Jean-Baptiste Touchard, sur Apple II

Rédacteur en chef :
Wladimir Mercouroff

Rédacteur en chef adjoint :
Christian Lafond

Comité de rédaction :
J. Arzac, H. Bestougeff, F-M. Blondel,
J.-Cl. Guiraudon, P. Muller

Ont participé à ce numéro :
G.L. Baron, M. Barrier, P. Bouchet, J. Brette,
B. Canale, A. et H. Deledicq, J.L. Frot,
S. Hocquenghem, M. Jacques, J. Cl. Le Touzé,
A. Meyer, P. Muller, M. Nivat, L. Paul,
S. Pouts-Lajus A. Rouchier, J.P. Roy,
C. Steyaert, P. Tixeuil, J.B. Touchard, J. Vicard.

Comité scientifique :
J. Beghin, A. Danzin, H. Fady, J. Hebenstreit,
C. Pair, J. Perriault, J.-C. Simon

Secrétariat d'édition :
Hélène Deledicq

Maquette :
Patrick Deiller

Administration, Abonnement :
9, rue Méchain, 75676 Paris Cedex 14

Abonnements :
1 an (5 numéros)
110 F (France) 140 F (Étranger)

Rédaction :
32 bd St-Germain,
75005 Paris - Tél. : 326.42.71

Publicité :
Média Régie
29 rue Violet - 75015 Paris
Tél. : 575.05.60.

Composition, photogravure :
SEDAG, 5 rue de Pontoise
75005 PARIS

Imprimé en France par :
Vaudrey à Lyon en février 1983

Directeur de la publication :
Jean-Jacques Nathan
Directeur délégué :
François Robineau
Dépôt légal : 1983, février
N°ISSN 0245-4858
Commission paritaire : 62 701
N° d'éditeur : 1-83-1
© 1983 Librairie Fernand Nathan
9, rue Méchain, 75676 Paris Cedex 14
© 1983 Éditions Cédic
32, bd Saint Germain - 75005 PARIS

Toute reproduction même partielle et par tous procédés,
actuels ou à venir, de cette publication est interdite. Elle
constituerait une contrefaçon passible des peines prévues
par la loi du 11 mars 1957 sur la protection des droits
d'auteurs.

ABONNEMENT PAGE 12

6 SAISIE DIRECTE

La Fondation Jean-Marc Salomon
L'Association des Informaticiens de Langue
Française, les stages lourds et les rubriques
habituelles

15 ÉDITORIAL

L'illusion d'une panacée

16 ABC de l'E.A.O.

E.D.A.O. **Bruno Canale et Pierre Tixeuil**

19

Présentation du dossier

20

Où en est l'E.A.O. en maths ?
Georges-Louis Baron

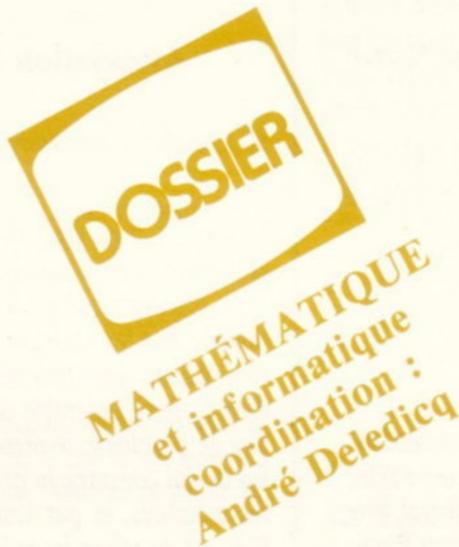
23

Les Mathématiques Illustrées par Ordina-
teur. Quelques exemples.
Serge Hocquenghem

26

Kijouakoua? **André Rouchier**

27



La mathématique, science expérimentale :
• le bon vieux plumeau de Taylor
André Deledicq
• la triste histoire d'Arthur Bécassin
Andrée Meyer - Colette Steyaert

29

Il était une fois un didacticiel...
Georges-Louis Baron - Jean-Louis Frot

31

Informatique et Conjectures et résultats
mathématiques. **Jean Brette**

34

Ordinateurs et calcul symbolique.
Jean-Paul Roy

37

Être ou ne pas être... Mathématicien et/ou
informaticien. **Pierre Muller, Andrée
Meyer, Maurice Nivat.**

39 VOS PROGRAMMES

1. Petite Annonce. **Serge Pouts-Lajus**
2. Dessiner un polyèdre.
Jean Baptiste Touchard
3. L'aire d'un triangle. **Serge Hocquenghem**
4. Droites Remarquables d'un triangle.
André Deledicq

43 CÔTÉ TECHNIQUE

La tortue maternelle. **Jean Claude Le Touzé**

46 VOUS AVEZ DIT... ?

Informatique. p.c.c. **Pierre Muller**

47 EXPÉRIENCES ÉTRANGÈRES

Faut-il avoir peur des didacticiels américains ?
Michel Barrier, Michel Jacques

SUPPLÉMENTS DÉTACHABLES

3 et 4

Leçon n° 9 : Structurer
André Deledicq

49 et 50

Cobol d'un coup d'œil
Pierre Bouchet et Jean Vicard

FONDATION

Nos lecteurs se souviennent de **Jean-Marc Salomon** disparu tragiquement l'an dernier : au sein de notre Comité de Rédaction il animait la rubrique « la vie des clubs »

A l'initiative de sa famille sous l'égide de la Fondation de France et avec le concours de l'ANSTJ, la **Fondation Jean-Marc Salomon** a été créée en 1982 ; Elle a pour objet d'aider les jeunes de toute nationalité s'intéressant aux sciences et aux techniques (informatique et astronomie en particulier).

Le Comité, présidé par M. **Gérard Salomon**, attribue chaque année des prix après l'étude des dossiers. Il est constitué de personnalités de l'industrie et de la recherche qui ont bien connu et apprécié Jean-Marc Salomon :

Michel Bignier, Directeur à l'Agence Spatiale Européenne

Jean-Claude Guiraudon, Chef de la Section Science Jeunesse au Palais de la Découverte

Jacques Perriault, Directeur de programme à l'Institut National de Recherche Pédagogique

Olivier Las Vergnas, Responsable Formation à l'Association Nationale Science Technique Jeunesse au Palais de la Découverte

Pierre Quéward, Directeur des activités civiles, MATRA

Jean-Jacques Salomon, Professeur au Conservatoire National des Arts et Métiers, Chef de la Division des politiques de la science et de la technologie à l'O.C.D.E.

Jean-Paul Zahn, Directeur de l'Observatoire du Pic du Midi

auxquels sont associés, pour les délibérations, comme membres de droit, les lauréats de l'année précédente.

A qui s'adresse la fondation ?

Chaque année, des jeunes que passionnent les sciences et les techniques échaffaudent des projets : créer une association, constituer une entreprise ou une société, effectuer un voyage d'études, étudier et développer un projet, améliorer une machine, organiser des sessions d'information ou de vulgarisation scientifique, constituer un conservatoire d'objets techniques...

Malheureusement, trop souvent isolés ou en butte à l'incrédulité de leur entourage, ils doivent renoncer, faute de conseils judicieux et d'une aide financière ou matérielle.

C'est à eux, jeunes de moins de 21 ans, quelle que soit leur nationalité, que s'adresse la **FONDATION JEAN-MARC SALOMON**.

Quelle aide attendre de la fondation ?

Chaque année, la Fondation distribue des prix dont elle fixe le nombre et le montant en fonction de l'intérêt des projets présentés. Outre la somme d'argent correspondant au prix proprement dit, la Fondation offre, pour ceux qui ont le goût d'entreprendre :

- des conseils, des suggestions, des recommandations pour une meilleure formulation du projet
- une introduction auprès des équipes d'amateurs ou de professionnels poursuivant des objectifs similaires.

Comment présenter son dossier ?

Le règlement intérieur est à la disposition des candidats. Ses lignes principales sont les suivantes :

Les dossiers sont à adresser entre le 15 novembre et le 30 avril de l'année suivante pour être examinés par le Comité, qui fait connaître ses décisions avant le 30 juin.

Chaque dossier doit comprendre :

- une fiche de présentation du candidat ou des candidats associés à un projet collectif
- la description du projet en mettant en particulier en valeur ses objectifs et ses motivations
- le calendrier de sa réalisation avec les principales échéances et les moyens nécessaires pour parvenir au succès
- éventuellement, une lettre d'accompagnement d'une personne aînée, soutenant son initiative.

Ces documents sont à adresser à la :

FONDATION JEAN-MARC SALOMON

FONDATION DE FRANCE

40, avenue Hoche

75008 PARIS

Tél. : 563.66.66

Comment encourager les activités de la fondation ?

La FONDATION JEAN-MARC SALOMON est habilitée à collecter des fonds auprès de ceux (particuliers, entreprises, centres culturels et techniques, associations) qui souhaitent contribuer à l'expression de jeunes talents dans le domaine des sciences et des techniques.

Elle offre des avantages fiscaux prévus par la loi pour :

- les particuliers qui peuvent déduire de leur revenu imposable le montant de leurs dons à concurrence de 3 % de celui-ci
- les entreprises qui peuvent déduire leurs dons des bénéfices imposables à concurrence de 3 % de leur chiffre d'affaires.

Pour tout renseignement complémentaire, s'adresser à la FONDATION DE FRANCE

ASSOCIATIONS

Association des Informaticiens de Langue Française

Rôle de l'A.I.L.F.

L'A.I.L.F. s'est créée dans le but de promouvoir une informatique en français et par conséquent, de réagir contre l'invasion de l'anglais dans et par l'informatique.

Pour atteindre ce but, il faut susciter une prise de conscience de l'enjeu chez les informaticiens et les utilisateurs de l'informatique ; il faut les mobiliser pour qu'ils réagissent chacun dans leur secteur, et si possible tous ensemble. Mais il faut aussi recueillir et traiter l'ensemble de l'information relative à la langue dans l'informatique, étudier le problème et prendre dans certains cas des mesures d'ordre gouvernemental.

En ce qui concerne la prise de conscience, l'A.I.L.F. peut agir directement auprès de ses membres, et par leur intermédiaire, toucher un public beaucoup plus large.

Il en est de même pour la mobilisation : regroupant de plus en plus d'informaticiens et d'utilisateurs de l'informatique, elle peut créer et propager un certain esprit offensif.

En ce qui concerne l'information, l'A.I.L.F. peut l'obtenir par ses propres moyens ou en organisant des enquêtes, et l'exploiter elle-même.

Elle peut de ce fait étudier nombre de problèmes et proposer ensuite des mesures à prendre aux organismes compétents.

Ni syndicat, ni société savante, l'A.I.L.F. doit donc être à la fois :

- un **forum** ouvert à tous les informaticiens et utilisateurs de l'informatique,
- un **animateur** de la réaction contre l'hégémonie anglo-saxonne dans l'informatique et un **promoteur** de l'informatique en français,
- un **pôle d'information** destiné à rassembler les données des différents problèmes ainsi posés et à les fournir à la demande.
- un **centre d'études** pour tout ce qui concerne l'informatique en français.
- enfin, un conseil à la disposition des instances gouvernementales, des élus et plus généralement des responsables de l'informatique française.

Relations avec les organismes intéressés et les autres associations

Le rôle multiple de l'A.I.L.F. la situe au carrefour de l'intervention des divers organismes ou associations concernés par l'informatique et la langue française.

Elle n'entre en compétition avec aucun d'eux et se propose au contraire, compte tenu de son but, de constituer un trait d'union entre tous ceux qui partagent le propos "d'entrer en français dans l'ère informationnelle".

Déjà patronnée par le Haut Comité de la Langue Française et l'Agence de l'Informatique, l'A.I.L.F. peut coopérer directement avec tout autre organisme intéressé.

Elle doit d'autre part entretenir des relations étroites avec toutes les associations scientifiques directement ou indirectement concernées telles que l'AFCEP ou l'ANSULF, les associations d'informaticiens telles que la SFI et l'ensemble des associations de promotion ou de défense de la langue française.

Il importe enfin qu'elle opère en liaison constante avec les associations internationales qui se préoccupent d'informatique et de linguistique, ainsi qu'avec les associations nationales qui se proposent de défendre la langue et la culture de leur peuple à travers l'aventure informatique.

Afin de faciliter l'entente et la coopération, il est fait appel à toutes ces associations pour qu'elles adhèrent à l'A.I.L.F. ou s'y fassent représenter.

Inversement, l'A.I.L.F. se propose d'adhérer en tant que "personne morale" à toute association qu'elle estimerait devoir soutenir.

Fonctionnement

L'A.I.L.F. fonctionne d'ores et déjà comme "forum". En effet, plusieurs réunions ont déjà permis à un certain nombre de membres d'échanger des idées, de débattre des principaux problèmes qui les préoccupent, et d'établir des plans d'action. Une correspondance assez importante est allée dans le même sens.

Ces activités qui constituent la base du fonctionnement et sont assurées par les moyens propres de l'Association, s'amplifieront avec le nombre des adhérents et pourront comporter l'organisation de colloques, la parution d'un Bulletin de Liaison, et ultérieurement d'une Revue.

Premières actions envisagées

Trois actions concrètes sont prévues dans un premier temps :

- La constitution et l'édition d'un Annuaire des Informaticiens de langue française.
- La constitution et l'édition d'un Annuaire des institutions et des diverses initiatives officielles et privées dispensant une formation en informatique.
- Une enquête sur "l'invasion" de l'anglais dans l'informatique et sur la prise de conscience de cette invasion par les différentes catégories d'informaticiens et d'utilisateurs de l'informatique.

La commission de terminologie se réunit environ toutes les 3 semaines à l'Agence de l'Informatique.

Elle examine les termes informatiques qui lui sont soumis par diverses instances ; cherche la meilleure façon de traduire les termes anglais (ou autres) ; discute les termes français proposés et en crée éventuellement d'autres ; éclaircit les concepts ; met au point les définitions et propose des termes français assortis de définitions précises, susceptibles d'être publiés dans des décrets imposant leur usage.

La collaboration de l'A.I.L.F. doit consister à :

- signaler à la commission de terminologie tous les termes anglais ou autres qui lui semblent devoir être traduits ; les termes français à préciser, et, d'une façon générale tous les problèmes de terminologie qui lui paraissent se poser dans la pratique de l'informatique ;
- proposer des solutions (traductions, définitions, etc.).

Chaque membre de l'Association est invité à adresser au siège de l'A.I.L.F. les listes de termes qui lui posent problème ainsi que ses propositions de traduction ou d'éclaircissement.

L'A.I.L.F. exploitera le courrier ainsi reçu, en fera la synthèse et la confiera à celui de ses membres qui la représente à la Commission de terminologie pour qu'il la lui soumette.

Étant donné que la Commission de terminologie a déjà mis au point un certain nombre de termes et de définitions qui ont désormais "force de loi"* et qu'elle étudie actuellement nombre d'autres termes et concepts, il est indispensable que les membres de l'A.I.L.F. désireux de collaborer sachent ce qui est déjà fait et ce qui est en cours d'examen afin de ne pas faire un travail inutile.

C'est pourquoi nous proposons d'adresser à tous les adhérents intéressés qui en feront la demande une documentation complète sur les travaux passés et actuels de la Commission de terminologie.

Pour tout renseignement complémentaire sur A.I.L.F. et ses actions écrire à :

Général Jean BECAM

Président de l'A.I.L.F.

61, rue de Vaugirard

75006 Paris.

* La première liste de termes et de définitions est parue dans E & I n° 10. Nous rappelons que l'utilisation de ces termes a été rendue obligatoire au 22 juin 82, par un arrêté du 22 décembre 1981, dans les décrets, circulaires, instructions et directives des ministres et fonctionnaires de l'État, et dans les correspondances, documents et productions de quelque nature que ce soit qui émanent des administrations, services et établissements publics de l'état (ou qui leur sont adressés).

TERMINOLOGIE

micro-ordinateur

micro ; *orth.* les composés ne prennent un trait d'union que lorsque le second élément commence par une voyelle : micro-ampère, micro-analyse, micro-onde etc.

Le pluriel est marqué par un s final :

des micro-ordinateurs, des microglossaires.

Sens : "Petit" ; dans les mesures d'une grandeur marque la division au millionième de l'unité (micro-ampère, 10⁻⁶ A.)

EPI

Association Enseignement Public et Informatique

Résolution de l'Assemblée générale du 6 novembre 1982

Partout l'informatique progresse ; mais la réussite du développement de la filière électronique et des réseaux de communication dépend de l'aptitude de notre pays à réaliser pour ces moyens techniques les programmes qui occuperont demain une grande place dans notre espace culturel. C'est pourquoi le gouvernement appelle à "un effort vigoureux en faveur du développement des industries culturelles françaises".

L'Association Enseignement Public et Informatique (E.P.I.) considère qu'il est de la responsabilité de l'Éducation Nationale de participer à cette vaste entreprise qui ne saurait être le monopole des seules industries culturelles.

La France dispose d'un grand service public de l'éducation et de la formation continue qui a souvent, depuis plus de dix ans, engagé des actions originales d'introduction de l'informatique dans les divers enseignements. Il comprend aujourd'hui des milliers d'enseignants de toutes disciplines et de tous niveaux qui utilisent l'ordinateur comme outil d'amélioration et de transformation de leur enseignement.

Pour répondre aux besoins en rapide croissance, ce potentiel doit être valorisé et accru. Actuellement, le dispositif d'introduction de l'informatique dans l'Éducation Nationale se déploie et s'enrichit mais l'effort de recherche et de formation reste très insuffisant et le pragmatisme l'emporte là où il faudrait un grand dessein.

La priorité donnée à la dimension technologique et aux équipements risque de nous engager dans une impasse. Il est de la mission de l'Éducation Nationale d'exiger une définition claire des objectifs et des contenus culturels.

Ces préoccupations conduisent l'EPI à demander, qu'à l'initiative du Ministère de l'Éducation Nationale, s'engage d'urgence un large débat démocratique.

Liste mise à jour au 12 novembre 82 des "RÉGIONALES EPI" (une par Académie)

Aix Marseille E.J. SEYVOZ. *Chemin des Figeons 13510 ÉQUILLES*

Amiens B. CARDON. *2, rue P. Curie 80240 ROISEL*

Besançon M. Hillenweck. *1 bis, Ch. Des Monts de Bréguille 25000 BESANÇON*

Bordeaux R. RAMIS. *Lycée Borda 40106 DAX*

Caen C. BAIL. *Lycée J. Rostand 14034 CAEN Cedex*

Clermont Fer. JB. CHOUERY. *Route de Saligny 03290 DOMPIERRE/BESBRE*

Corse F. ROSSI. *Poggio-di-Lozzi 20224 CALACUCCIA*

Créteil C. FACCA. *112, rue Caron 91200 ATHIS MONS*

Dijon M. MASSARD. *33, Bd St Quentin 71200 LE CREUSOT*

Grenoble J. GAUCHE. *30, rue H. Berlioz 38420 DOMÈNE*

Lille M. LOONIS. *59, rue Ch. Gounod 59130 LAMBERSART*

Limoges F. LABROUE. *14, rue des Genêts 19300 ÉGLETONS*

Lyon R. BYK. *21, rue des Hautinières 69260 CHARBONNIÈRES*

Montpellier R. WALD. *12, rue des Avants Monts (prolongée) 34100 MONTPELLIER*

Nancy-Metz M. SCHWOB. *31, rue St Pierre 57000 METZ*

Nantes J. BARRAULT. *7, av. du Dr Peton 49400 SAUMUR*

Nice P. THEBAULT. *155, av. Ste Marguerite 06200 NICE*

Orléans-Tours JY. DUPONT. *20, rue Rabelais 37230 LUYNES*

Paris E. PÉLISSSET. *11, rue Morinet 94270 KREMLIN BICÈTRE*

Poitiers C. LEBON. *13, rue du Moulin de la Grille 17100 SAINTES*

Reims G. AMAND. *22, rue Duruy 51100 REIMS*

Rennes A. NICOLAS. *18, rue de Lettonie 35100 RENNES*

Rouen P. GAILLARDON. *Rés. Valger-Pourville 76550 OFFRANVILLE*

Strasbourg J. SCHNEPF. *5, rue Bergmann-Mittelhausbergen 67200 STRASBOURG*

Toulouse C. BARONET. *24, bd Bon Repos 31000 TOULOUSE*

Versailles J. BAUDE. *Lycée P. Corneille 78170 LA CELLE SAINT CLOUD*

Quelques productions EPI

A paraître début 83 :

Disquette EPI n° 3 : Nous préparons actuellement la disquette EPI n° 3 à paraître début 83. Elle comprendra notamment :

- un logiciel de traitement de texte sous LSE et un logiciel permettant de mieux comprendre le fonctionnement de LSE et complétant le dossier EPI à paraître sur ce sujet (voir souscription) ; un logiciel de lecture rapide.

Adhérents de l'EPI : Préparez vos 4 timbres à 1,80 F... et attendez le bon de commande à paraître !

Des exemplaires encore disponibles :

L'informatique à l'École : Travaux des lauréats du Concours de programmes AFCET-81 présentés par l'EPI.

En vente : 35 F (port compris)

Écrire à **M. CANAL** - Trésorier EPI

Lycée P. Corneille 78170 LA CELLE SAINT CLOUD

DOSSIER EPI : souscription ouverte aux adhérents de l'EPI :

LE SYSTÈME LSE

Mieux connaître son fonctionnement sur micro.

A paraître dans le courant du 2^e trimestre 82-83.

Le dossier PBI étant épuisé nous préparons une deuxième édition plus accessible, plus complète et surtout plus générale, puisque les procédures binaires seront intégrées dans une explication générale du système LSE. Il s'agira d'un dossier d'environ 120 pages (15 x 21) offset. Prix de souscription = 35 F port compris.

Pour nous permettre d'estimer le nombre d'exemplaires à imprimer nous demandons aux collègues intéressés de nous renvoyer rapidement un bon de commande.

- Extraits du Sommaire du Bulletin n° 28 (Déc.82) :

Document

- Extrait d'une enquête programmée par l'ANSULF sur la situation du français dans l'ensemble des sciences.

Rubrique pédagogique

- Exemple d'utilisation de banques de données dans l'enseignement des Sciences naturelles (classe de 1^{re}), **S. Dupouy**

- Intelligence artificielle et EAO - les systèmes experts - le langage naturel - les stratégies pédagogiques, **Alain Bonnet**

- Archéologie et informatique, **Michel Devaud**

- Le logiciel KAIB. Simulation de la dynamique d'un écosystème, **Christian Bravard**

- Un logiciel d'auto-apprentissage du Basic, **Hubert Nadin**

- Pour LOGO, **J. Louis Jourdan**

Rubrique technique

- Écrire des programmes en utilisant des procédures standards, **J. Lucy et J.J. Equoy**

- Video "OR" imprimante, **Émile François**

- Le graphique (suite) **collectif**.

etc.

Permanence E.P.I.

Comme l'année passée, Michel Devaud assure une permanence EPI au Centre de Formation de l'ENS de Saint-Cloud. 6, grande rue 92310 SÈVRES. Les micro-ordinateurs du centre sont à votre disposition, ainsi que la bibliothèque de logiciels. Membres de l'EPI, apportez vos disquettes vierges pour duplication (LX, R2E, SIL'Z)

Permanence EPI également le mercredi après-midi de 14 à 16 h au CRDP de Paris rue Jacob 75006 Paris.

Adhésion et abonnement au bulletin trimestriel de l'Association EPI.

S'adresser à :

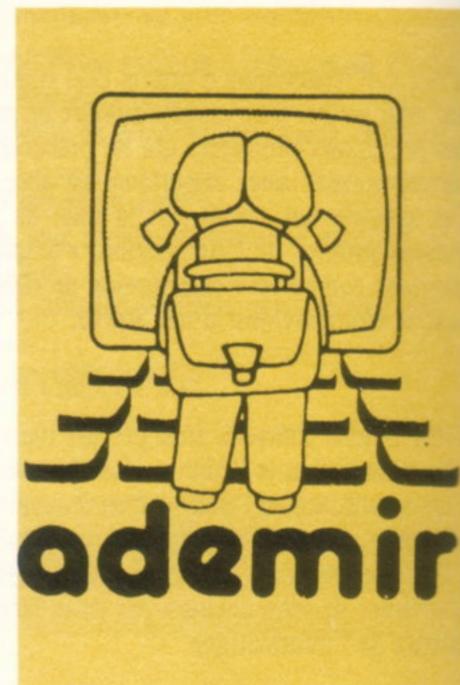
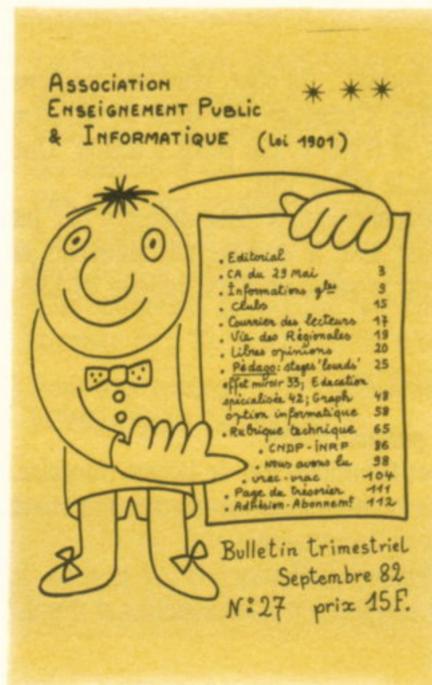
Michel Canal

Trésorier EPI

Lycée P. Corneille

78170 LA CELLE ST CLOUD

ou voir le modèle de bulletin d'adhésion dans E & I n° 12 (tarifs inchangés pour 1983).



ADEMIR

Le point sur les clubs ADEMIR

Rentrée scolaire 1982-83.

On compte à ce jour : 66 clubs ADEMIR

12 clubs créés cette rentrée ;

- Lyon Vaulx-en-Velin (école publique)
- Maison des Sciences de l'Homme Paris
- Montlignon (95) (école primaire publique)
- Champigny (collège)
- Denain (Nord) (LEP)
- Valence d'Agen (LEP)
- Grenoble (La Rampe/Maison des Jeunes)
- Troarn (Normandie) (collège)
- La Varenne St Hilaire (lycée)
- Marmande (lycée)
- Beauvais (LEP)
- St Juery (collège)

Bibliothèque de logiciels

Sur la base de 3 programmes éducatifs ou ludiques par année scolaire, les clubs ADEMIR ont mis en commun et échangé leurs réalisations. Au total : une douzaine de disquettes 5 pouces remplies de programmes divers (exercices d'enseignement et jeux).

Diffusion des logiciels professionnels dans les clubs ADEMIR

(Pascal - LSE - Voltaire - Pen - SBasic)

L'ADEMIR a négocié des prix (achats groupés/association) et demande ensuite aux clubs une participation de l'ordre d'un tiers du prix de vente normal du logiciel.

Ademir formation

Pour la première fois, l'ADEMIR organise des stages de formation spécifique à l'enseignement assisté par ordinateur, à partir d'un langage auteur PEN.

• 4 stages sont prévus pendant les vacances scolaires et sont agréés par l'UNAPEC (Union Nationale pour la Promotion Pédagogique dans l'Enseignement Catholique). Stages prévus à Paris :

7 février au 11 février 1983

28 mars au 1^{er} avril

4 juillet au 8 juillet

29 août au 2 septembre

• 1 stage sur l'initiation à l'utilisation de la télématique appliquée à l'enseignement est prévu (soutien de la DGT).

Club ADEMIR/télématique

Suite à un accord passé avec la DGT, 12 clubs seront équipés cette année de Minitel,

de cartes Vidéotex et de lignes téléphoniques pour :

- Organiser des journées "portes ouvertes" ; démonstration
- Interroger des serveurs ;
- Composer des pages Vidéotex, à usage éducatif et d'informations locales

Club ADEMIR/Maison des sciences de l'homme

54, Bd Raspail 75006 Paris

Un club ADEMIR et MICROTEL au centre de Paris vient d'être créé avec le soutien du CNRS, il est ouvert de 8 heures à 20 heures, et le samedi matin de 9 h à 12 h, et promet de bonnes activités. Bien équipé, bien encadré, des conférences et des démonstrations seront proposées aux nouveaux adhérents.

En projet

1^{er} colloque ADEMIR/EAO/TÉLÉMATIQUE à Beaubourg (avril ou mai, avec le soutien de l'ADI et de la DGT).

Expérience ADEMIR/EAO/TÉLÉMATIQUE

Dans une classe de perfectionnement d'une école publique de St Denis l'ADEMIR organise en accord avec une équipe d'enseignants une expérimentation EAO et télématique en mettant à disposition du matériel (1 micro-ordinateur avec graphique couleur Goupil et des logiciels Ademir (exercices et jeux), 1 Thomson T07... et des didacticiels pour les plus jeunes. Un enseignant Ademir se chargera d'initier les enseignants et participera aussi à l'animation E.A.O.

Un terminal TELETEL permettra aussi d'accéder aux programmes disponibles sur le serveur de la Villette et de Vélizy ou de composer des pages videotex spécifiques à l'école et aux préoccupations des élèves, à partir du micro-ordinateur devenu "serveur", voire de communiquer aussi avec les autres clubs Ademir ainsi équipés.

Durée : 1 mois.

Quelques exemples de programmes réalisés dans les clubs

Par des élèves

CLUB ADEMIR PARIS ÉCOLE ALSACIENNE

Une banque de données avec graphique couleur nous donnera des renseignements historiques, géographiques et touristiques sur la région du Quercy.

Les élèves ont aussi révisé leurs conjugaisons des verbes allemands en réalisant un didacticiel pour leurs camarades.

Ils ont aussi inventé un jeu "Pilote" en couleur pour amuser les plus petits et les plus grands, mais pour eux le jeu : c'était leur travail de programmation !

CLUB ADEMIR PARIS LYCÉE HÉLÈNE BOUCHER

Un élève a travaillé sérieusement ses verbes irréguliers anglais en réalisant un didacticiel de révision et d'exercices avec notation finale.

Par les enseignants

Des programmes d'enseignement divers ont été réalisés : exercices d'orthographe, révisions des déclinaisons latines et des conjugaisons des verbes irréguliers français, anglais, allemands ; entraînement au calcul rapide, études de fonctions, décomposition en produits de facteurs premiers (5^e), trigonométrie (3^e), étude de similitudes planes directes, recherche et construction de barycentre (2^{de}) ; chimie, électricité ; géographie (reconnaissance des fleuves et des villes) ; démographie, diététique, biorythme, horoscope, étude des thèmes ; rédaction d'un conte pour enfants ; des jeux ; la gymnastique au brevet ; petit vocabulaire du dialecte alsacien et lexique français, anglais, allemand par thèmes ; calendrier perpétuel (grégorien) ; des projets : gestion des conseils de classe, dictionnaire des citations et des proverbes...

CLUB ADEMIR ALES

Un professeur de français fait parler l'ordinateur GOUPIL à partir d'une étude phonétique et linguistique très élaborée. Le principe : une phrase tapée sur le clavier est analysée puis prononcée, deux secondes après, d'une voix grave un peu étrange...

Il parvient même à prononcer correctement des phrases pièges (phonétiquement) : "Les poules du couvent couvent" !

CLUB ADEMIR AVRANCHES

Un professeur de musique fait chanter le MICRAL (R2E) et compose des morceaux de musique sur GOUPIL 2.

CLUB ADEMIR PARIS STANISLAS

Un professeur a pensé aux néophytes et a réalisé un programme d'auto-apprentissage du BASIC. On peut ainsi apprendre tout seul, mais bien sûr les adhérents du club sont là aussi pour aider et conseiller les débutants...

Pour tout renseignement : ADEMIR

Association pour le Développement dans l'Enseignement de la Micro-Informatique et des Réseaux

9, rue Huysmans

75006 Paris

Tél. : 544.70.73.

GAGS

Université d'Été Informatique

Composition d'Algorithmique - Problème

Le TGV 615 part de Paris à 7 h 15 mm.

De Paris à Saint Florentin, il roule, sur cent vingt kilomètres, à une vitesse moyenne de cent quarante kilomètres heure.

Quittant l'Impériale après avoir parcouru ces cent vingt kilomètres, il s'engage sur la voie à grande vitesse PSE.

Négligeant les temps d'accélération et de décélération de Sathonay à Lyon-Brotteaux, nous admettons que le TGV 615 roule à une vitesse constante de deux cent quatre vingt kilomètres heure sur les trois cent dix kilomètres que représente ce trajet.

Le TGV 616 quitte Lyon-Brotteaux à 7 heures en direction de Paris.

Considérant qu'il emprunte en sens inverse le même parcours que le TGV 615 aux mêmes vitesses, répondre aux questions suivantes :

Question n° 1 :

A quelle heure se croisent-ils ?

Question n° 2 :

Sachant que chacune des rames, en UM, mesure quatre cent trente mètres, quelle sera la durée du croisement ?

Remarque : Ne pas écrire le programme qui calcule le résultat.

Marc Bergman, Bernard Méot

Composition d'Histoire de l'Enseignement Assisté par Ordinateur

Voici deux textes dont l'un est un logiciel d'Enseignement Assisté par Ordinateur, destiné à des élèves de 6^e/5^e et l'autre un exercice extrait du :

"**Livre Unique de Français - Cours Élémentaire**"

Hachette - 1929.

Texte 1 :

Mettez au féminin les mots suivants :

Ex. : fumeux, fumeuse.

fumeux, soigneux, jaloux, moqueur, frileux, honteux, heureux, nuageux, boudeur, pluvieux.

Texte 2 :

Donnez le féminin des adjectifs suivants :

sage, net, joli, simplet, marron, grenat, exigu, épais, original.

Question :

Lequel est le logiciel d'Enseignement Assisté par Ordinateur ?

Hélène David

MINISTÈRE DE L'E.N.

Voici la liste mise à jour des Centres de stages de formation, à l'informatique, des professeurs.

Pour la rentrée 83-84, quelques nouveaux centres seront peut-être créés : la décision sera prise début février ; le cas échéant, nous vous en informerons dans notre prochain numéro.

Nous vous faisons part à nouveau de la marche à suivre pour postuler un stage "lourd" :

Préparez un dossier de candidature. Envoyez-le par la voie hiérarchique :
1 — à la Direction dont vous dépendez (Lycées, Collèges, Écoles...)
2 — à l'I.P.R. "délégué aux technologies nouvelles", au Rectorat de votre académie

3 — Au Chef de la mission académique de la formation des personnels de l'Éducation Nationale (un par académie ; adresse : Rectorat ; liste nominative, cf. E & I n° 12)

Les décisions étant décentralisées, il ne semble par indispensable d'envoyer un double du dossier à la MITIF.

MARSEILLE

C.R.E.F.I.
Faculté de Luminy
10, route Léon Lachamp
MARSEILLE CEDEX 5
(91) 41.39.40

GRENOBLE

Stage Informatique d'Applications Pédagogiques
Bât TP de chimie-Université
GRENOBLE 1 B.P. 53X
38041 GRENOBLE CEDEX
(76) 44.82.76 poste 297

LILLE

Responsable : J.P. Steen
C.U.R.F.I.P. - I.E.E.A.
Université de Lille 1
59655 VILLENEUVE D'ASCQ
CEDEX
(20) 91.92.22. poste 28-70

LYON

C.R.E.F.I.
Université Claude Bernard - Lyon 1
43, avenue du 11 Novembre
69100 VILLEURBANNE
(7) 889.81.24 poste 37-43

NANCY

I.U.T.
2 bis, bd Charlemagne
54000 NANCY
(8) 327.30.85

NANTES

Centre de Formation à l'informatique et à ses applications pédagogiques ; Ensemble "Sciences"
44072 NANTES CEDEX

(provisoirement 40 74.50.70
poste 398

NICE

C.F.P.I.
Lycée Audibert
Boulevard Wilson
06602 ANTIBES
(93) 61.81.46

ORLÉANS

C.F.P.I. - I.R.E.M.
45046 ORLÉANS-CEDEX
(38) 63.22.16 poste 638

JUSSIEU - (PARIS)

C.F.A.P.I.
Tour 25
Couloir 25-15 (2^e étage)
2, Place Jussieu
75251 PARIS CEDEX 05
(1) 336.25.25
poste 45-84 ou 54-06

SAINT-CLOUD

E.N.S. de Saint-Cloud
Grille d'Honneur
2, av. du Palais
92211 SAINT-CLOUD
(1) 507.11.58

PARIS-ULM

Centre de Formation à l'informatique et à ses applications pédagogiques
5, rue Censier 75005 PARIS
(1) 535.41.35.

VILLETANEUSE

I.U.T.
Centre de Formation Continue
C.F.I.E.
Avenue Jean-Baptiste-Clément
93430 VILLETANEUSE
(1) 821.61.70
poste 49-65 (ou 48-40)

POITIERS

C.F.I.A.P.
U.E.R. Sciences -
Bât. Sciences Naturelles
40, avenue du Recteur Pineau
86022 POITIERS CEDEX
(49) 46.26.30. poste 688

RENNES

C.R.E.F.F.I.B.
Campus de Beaulieu
35042 RENNES CEDEX
36.48.15 poste 10-42
ou 36.20.00 poste 428

TOULOUSE

Centre de Stage de Formation à l'Informatique et à ses applications pédagogiques
Bâtiment des mathématiques
Université Paul Sabatier
118, route de Narbonne
31062 TOULOUSE CEDEX
(61) 25.56.71

PARIS VI - Prof E.N.

Institut de Programmation
Université PARIS VI
75230 PARIS CEDEX 05
(1) 336.25.25. poste 53-90.

LU DANS LA PRESSE

La revue trimestrielle *Vecteurs* a consacré son numéro 4 (oct. nov. déc. 1982) à "la communication aujourd'hui"

Au sommaire notamment
• un article de Jacques Hebens-treit, présenté par Wladimir Mercoureff, "l'Enseignement Assisté par Ordinateur, mode d'emploi", reprenant un article de E & I n° 11.

• un article de Lucien Boyer et Pierre Nolot "Ordinateur, jeu d'échecs et quelques autres jeux".

adresse : VECTEURS
BP 180
31014 Toulouse Cedex

Lue dans le *Supplément n° 2 au Bulletin National de l'A.P.M.E.P. n° 335, déc. 82* cette information qui intéressera aussi les collègues non matheux :

MISE EN GARDE A PROPOS DES CRÉDITS DE MAINTENANCE DES ÉTABLISSEMENTS ÉQUIPÉS

1 — Des établissements déjà équipés ont reçu un crédit de maintenance correspondant au centime près au contrat d'entretien du parc machines. Or dans l'esprit des responsables de la cellule informatique de la D.L., il devait y avoir une petite somme supplémentaire pour l'achat de disquettes ou autres besoins analogues. Vérifiez auprès de votre proviseur ou de votre intendant que votre lycée a bien reçu un crédit supérieur au coût de la maintenance. Faites-nous part de vos problèmes.

Nous vous mettons en garde sur le fait que l'année prochaine, ces crédits seront globalisés avec le reste du budget de l'établissement. Le bureau émet déjà quelques craintes, connaissant les pratiques antérieures.

N'oubliez pas de faire mentionner dans le projet de budget ce crédit supplémentaire.

2 — Les établissements nouvellement dotés vont "bénéficier" d'une garantie modifiée, quel que soit le type de matériel dont ils sont équipés :

• Pour 8 micro-ordinateurs, ils auront droit à un maximum de dix interventions durant l'année de garantie (au delà, les frais de dépannage seront à la charge de l'établissement !). Il faudra donc se montrer prudent dans les demandes d'intervention...

• Par ailleurs, la procédure de dépannage est modifiée : le technicien vient, emballe et emporte la (ou les) machine(s) défectueuse(s). Le retour du matériel réparé doit se faire dans les 10 jours ouvrables... ce qui en fera bien 15 !

• Enfin, à l'échéance de l'année de garantie, le parc machines sera soumis à une visite systématique d'au moins 2 h.

• **Rappel** : à la livraison de la dotation, il est important de signer le Procès-verbal de réception définitive (dit "de service régulier") le plus tôt possible, ces P.V. étant regroupés par tranches de 10 établissements à la fois.

petites annonces

VIFI-NATHAN poursuit son développement de logiciels éducatifs originaux :

Ses microdidacts

Vous avez des idées nouvelles et distrayantes de programmes éducatifs sur micro-ordinateurs. Vous êtes capable de les réaliser. Rejoignez les équipes d'auteurs VIFI-NATHAN !

Écrire à
VIFI-NATHAN
Service Développement
32 bd St Germain
75005 Paris

Petite machine à écrire électronique EP-20 cherche à savoir si elle peut être modifiée pour servir d'imprimante à un micro-ordinateur. Écrire au journal qui transmettra.
BROTHER

Le Centre d'Études et Systèmes des Technologies Avancées (CESTA) — 5, rue Descartes — PARIS 5^e, organise les 24 et 25 mars 1983, 2 Journées Internationales sur les « Applications des nouvelles technologies à l'éducation multimédia ».

Renseignements :
Anne-Françoise TESTER
— Tél. (1) 634.32.95
ou 634.32.90
Participation :
1 250 F.

DES REVUES

L'informatique professionnelle.
n° 6 — octobre 1982, 162 p., 60 F

Ed. : 99, bd Jean-Jaurès 92100 Boulogne

Faut-il avoir peur des informaticiens ?

C'est à cette question fort pertinente que tente de répondre la revue « l'informatique professionnelle » qui y consacre la totalité de son n° 6 (octobre 82), un dossier dont nous conseillons vivement la lecture, tant il est bien conçu. Observons le « monsieur » tout d'abord, bien sous tous rapports : 25-40 ans, « logique », bons revenus (100 à 130 000 Francs annuels en moyenne), en majorité des ingénieurs, mais aussi « individualiste », tel est grossièrement le portrait qu'en dresse J.C. Tallineau.

De quoi se nourrit-il?... de papier ; l'informatique suit la presse comme gros consommateur de papier... et destructeurs de forêts (« point de vue du papetier » de L. Floccia). Ceci n'empêche pas Brice Lalonde de porter un regard bienveillant... et réaliste « Rien ne porte en soi tout le bien et tout le mal ». Y aurait-il, dans la profession, de bons et de mauvais informaticiens ?

Au demeurant, l'espèce n'est pas en voie de disparition (300 à 500 000 en France et entre 5 et 10 millions dans le monde). Mais, Philippe Lemoine pose la question du « manque d'informaticiens » relevant que l'évolution de la technologie et des langages conduisent « l'utilisateur à se passer du technicien pour faire marcher l'outil ». Pour Ph. Lemoine, l'harmonie s'installe, l'informaticien trouve sa place et « on

ne veut plus sa mort ». Fort heureusement, car paradoxalement, il est protégé, notamment dans son activité de créateur de logiciels (« le point de vue du juriste » — A. Lucas). Tout ceci laisserait penser qu'il n'y a plus à avoir peur des informaticiens. Conclusion peut-être un peu hâtive, comme en témoignent les « quelques bavures de l'informatique » rapportées par J. Bounine ou les difficultés du dirigeant de PME avec les ordinateurs et les informaticiens (J. Edelman).

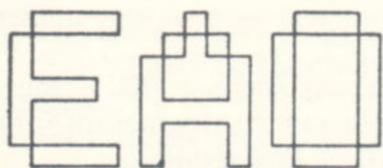
Le véritable enjeu, dit Alain Tournain, c'est « l'information », et il y a lieu pour la Société de la contrôler, c'est le rôle de la Commission Nationale Informatique et Liberté. C'est aussi celui de l'éducation comme le souligne J.P. Fourcade ; ainsi initié, on cessera de s'interroger sur « l'étrange regard de l'informaticien » (conte de Lucas de Malzan), ni drogué, ni obsessionnel, mais tout simplement passionné... Mais, attention, la passion n'est pas sans danger.

L.P.

Naissance de la revue « EAO, Mensuel d'actualité de l'Enseignement Assisté par Ordinateur »

adresse : A. Jour
11, rue du Marché St-Honoré,
75001 Paris
Tél. (1) 261.45.17

abonnement : 12 n°s + hors série,
800 F
n° 1 : janvier 83.



Chaque mois, EAO propose une information productive sur le monde

de l'Enseignement Assisté par Ordinateur :

- Stratégies des acteurs
- Marchés de l'EAO
- Nouveaux didacticiels
- Enjeux économiques, politiques et culturels
- Actualités internationales
- Technologies de l'EAO
- Congrès, colloques, séminaires, études, ouvrages

L'Établissement Public du Parc de la Villette (E.P.P.L.V.) diffuse un journal d'informations de 8 pages : « le courrier de la Villette »



Ce numéro 1 informe sur l'avancement des travaux :
- du musée des sciences, des techniques et des industries,
- de la cité de la Musique,
- et du « Parc dans la ville » (30 hectares).

On trouve aussi un bref historique sur la transformation progressive de ce qui a été pendant un siècle un marché de la viande

parisienne, en Établissement Public à vocation culturelle. La Villette possède déjà un serveur (ordinateur relié à une ligne téléphonique), Sevil, qui propose notamment des programmes de vulgarisation scientifique et technique (énergie, radioactivité, informatique...) - quarante accès simultanés - plus de 900 appels par semaine. Nous vous parlons de Sevil dans le prochain numéro de E & I consacré à la télématique.

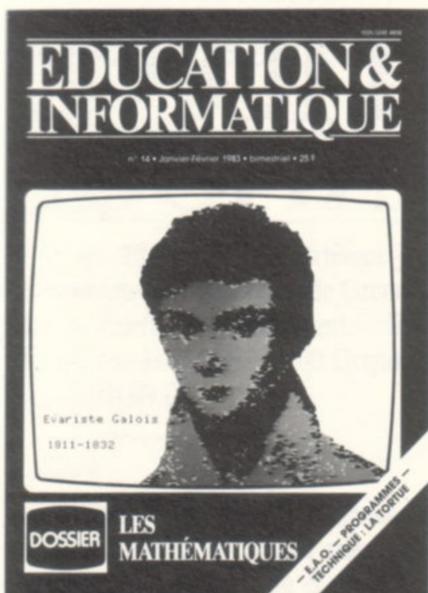
Le numéro 2 du *Courrier de la Villette* paraîtra en février 83

adresse : Le courrier de la Villette
E.P.P.L.V.
291, avenue Jean-Jaurès
75019 Paris

NEWTON

Si vous vous intéressez à la science en général, à l'informatique et à l'italien de surcroît, nos confrères transalpins sortent une revue vivante et bien documentée qui s'adresse aux jeunes à partir de 12 ans.

Renseignements :
NEWTON
Pigreco s.r.l.
via volta 35
22100 Como
Italie.



EDUCATION & INFORMATIQUE

La revue des enseignants et des formateurs

Dans chaque numéro, un dossier d'une quinzaine de pages traite les domaines qui vous intéressent. Parmi les dossiers déjà parus :

- L'informatique chez les littéraires
- Les jeux électroniques
- La formation des techniciens
- Les langages de programmation
- L'informatique à l'école élémentaire



ABONNEMENT

NOUVELLES TECHNIQUES

Les maths sans peine

Software Arts qui a développé le célèbre logiciel VISICALC (250 000 exemplaires vendus) annonce un nouveau logiciel miracle : TK ! Solver (ne pas oublier le point d'exclamation !) qui devrait permettre de faire des calculs compliqués à base de formules complexes et de tableaux sans avoir à programmer dans un quelconque langage de programmation. Disponible sur APPLE et sur IBM.

16 bits pour Goupil

SMT annonce le Goupil SM68000 autour du microprocesseur du même nom. Comportant 256/512 K ou 68/128 K de mémoire et acceptant disque dur, disque souple (8 et 5 pouces) et modem 1200 bauds ; Goupil SM68000, qui se situe en haut de gamme, vise à faire concurrence au Micro méga de Thomson. Commercialisation au début de 1983.

Si le cube de Rubik vous angoisse...

La "cubomanie" est enfin vaincue. Si vous en avez assez de tourner et de retourner le célèbre cube sans jamais arriver à la solution il existe un moyen simple de ne pas perdre vos soirées et de retrouver le sommeil : confiez votre cube au robot AKR300 fabriqué en France par AOIP. Formé d'un simple bras articulé et d'une caméra couleur de télévision (avec un ordinateur entre les deux) le robot accepte le cube dans n'importe quelle configuration et après une seconde de réflexion

(500 instructions FORTRAN) se met à l'ouvrage à raison d'une centaine de rotations en une demie-heure, pour reconstituer le cube initial. Pour retrouver le sommeil il ne vous coûtera que la modique somme de 600 000 F (ht !) mais sachez que pour le même prix ce robot sait aussi souder, peindre, déplacer des charges, etc.

Une caisse enregistreuse qui parle

Hugin Retail System offre aux points de vente des supermarchés un système qui lit les "codes-barres" et énonce à haute voix pour chaque article, le prix de celui-ci ainsi que le total à payer. Il ajoute même "Merci" après le paiement.

Un satellite à large bande

Intelsat VI qui sera lancé en 1986 pourra transmettre simultanément, 33 000 conversations téléphoniques et 4 chaînes de télévision pour un poids total de 4 tonnes.

Tant l'on crie au loup...

Le TI99/4A de Texas Instrument est enfin disponible. Microprocesseur 16 bits (9900 de chez Texas), 16 koctets de RAM (extensible à 48 koctets) générateur de sons, carte couleur (16 couleurs) avec une définition de 192x256 pts, interface pour 2 magnétophones à cassettes, clavier QWERTY de 48 touches, avec en option : disquettes, imprimante,

synthétiseur de parole. Il accepte les langages BASIC, TILOGO et assembleur. Il est connectable à tout télévi-

seur en couleur ayant une prise Peritel et coûte aux environs de 3500 F en version de base. JH/VC

FORMATION

Formation pour les entreprises

Parmi tous les stages et les séminaires proposés par différentes sociétés, nous sélectionnons pour ce numéro ceux de la société F.D.S.

LES OUTILS DE LA BUREAUTIQUE, deux jours, les 17 et 18 février / 21 et 22 mars 1983.

Un séminaire destiné aux cadres, pour les familiariser avec les principales applications de la bureautique : traitement de texte, communication au bureau, gestion des documents.

Une approche concrète et pratique du sujet.

LE LANGAGE "C", deux jours, les 16 et 17 février 1983.

Les principales caractéristiques du langage "C" qui a permis d'écrire UNIX. Un séminaire inédit en France.

Séminaires pour utilisateurs non-spécialistes

DÉCOUVERTE ET CHOIX D'UN MICRO-ORDINATEUR, deux jours, 14 et 15 février 1983

Le choix du matériel, le choix du logiciel, la démarche à suivre.

VISICALC, deux jours, 24 et 25 février / 17 et 18 mars 1983.

Le progiciel des cadres gestionnaires : principes de mise en œuvre, travaux pratiques sur micro-ordinateurs.

LES OUTILS DE LA BUREAUTIQUE, deux jours, 17 et 18 février / 21 et 22 mars 1983.

Séminaires pour programmeurs débutants

PREMIERS PAS EN BASIC, trois jours, du 21 au 23 février / du 9 au 11 mars 1983.

Programmation sur machine par petits groupes : premières notions et concepts du Basic. L'apprentissage par l'action.

FICHIERS EN BASIC, deux jours, 28 février, 1^{er} mars / 14 et 15 mars.

Pour réaliser des programmes Basic mettant en œuvre les fichiers.

Prix des stages : 1700 F/jour

Pour tout renseignement (programme détaillé, inscription etc.) :

F.D.S. Formation

10, rue Henri Pape

75013 Paris

Tél. : (1) 588.76.53.

ABONNEZ-VOUS VITE A :

EDUCATION & INFORMATIQUE

Veillez m'adresser Education & Informatique

- Pendant un an, au prix spécial de 95 F (140 F hors métropole)
- Pendant deux ans, au prix spécial de 180 F
- Pendant trois ans, au prix spécial de 250 F

Ci-joint un chèque de F ...

Je désire une facture

Nom, prénom : Profession :

Adresse : Signature :

Bon à retourner accompagné de votre règlement à : Librairie Fernand Nathan
Service abonnements : 9, rue Méchain - 75676 PARIS CEDEX 14

COLLOQUES

— passés

**JOURNÉES LOGO
CLERMONT-FERRAND
9-10-11 décembre 1982**

Plus d'une centaine de personnes, essentiellement des enseignants, se sont retrouvées à Clermont-Ferrand les 9-10-11 décembre 1982 pour un premier colloque à propos de LOGO.

Il faut dire que LOGO est depuis quelques années un instrument de travail essentiel pour plusieurs équipes de recherche dans lesquelles se retrouvent des Instituteurs, des Professeurs de Collèges, des Enseignants du Supérieur, etc. Il leur paraissait essentiel de confronter leurs expériences à un moment où il semble que l'intérêt officiel se fixe sur le langage, la tortue et leurs applications pédagogiques.

Le travail des participants à la rencontre a été organisé en trois thèmes :

• LOGO et les contenus d'enseignement et de formation.

• Évolution du langage et des matériels.

• L'organisation des apprentissages et les problèmes posés par l'introduction d'un système LOGO dans un cadre scolaire et/ou dans une perspective de formation ou d'animation.

... sans oublier un « thème » important dans ce genre de rencontre : assister à quelques présentations de LOGOS présents et à venir !

Des actes seront publiés, vraisemblablement dans le courant du mois de mars 1983. Ils seront co-édités par l'INRP et les IREM de Clermont-Ferrand et d'Orléans au prix approximatif de 30 F. Les personnes intéressées peuvent se renseigner auprès de Rouchier André - IREM - Université - 45046 ORLÉANS CEDEX.

... à venir

Grenoble

• 16 au 18 février : quatrièmes journées micro-informatiques de Grenoble, par le Cuefa. Renseignements : BB & A, 14, rue Jacquard, 38100 Grenoble ; tél. 76/40.09.04.

Cannes

• 22 au 25 février : Sécurocom'83, congrès mondial et premier salon international de la protection et de la sécurité informatique et des communications, à Cannes. Renseignements : Sedep expositions, 8, rue de la Michodière, 75002 Paris ; tél. 1/742.41.00.

LES CRIMINELS A COL BLANC

ou

LA FRAUDE INFORMATIQUE

LE PROFIL DU FRAUDEUR INFORMATIQUE

Selon une étude américaine, le fraudeur est mâle, blanc, jeune (20 à 30 ans), son casier judiciaire est vierge. Il est brillant, créatif, énergique, il a confiance en lui, il accepte volontiers le défi et l'aventure. Ses motivations peuvent être de plusieurs ordres :

- goût du luxe.
- problèmes familiaux.
- dettes de jeu.
- il peut obéir au sentiment de l'exploitation par un employeur froid, indifférent.
- il peut être anti-social.
- il n'est peut-être pas promu à l'échelon qu'il croit mériter.

Dans la plupart des cas il ne se considère pas comme un voleur mais comme un emprunteur.

LES DÉLITS LES PLUS COURANTS

- Les détournements de fichiers.
- Les invalidations de fichiers.
- Le vol ou le piratage.
- La prise en otage de bandes magnétiques.

COMMENT COMBATTRE LA FRAUDE

Pour combattre les fraudeurs, il faut connaître leurs techniques. Ce n'est pas une tâche facile. Chaque fraudeur est un inventeur particulier, bien que commençant à naître des groupes de fraudeurs (notamment au niveau des comptables). Jusqu'à présent, c'est la chance, le hasard qui ont permis les détections de fraudes.

On peut méditer ce conseil qui vient des analystes américains : « Vos employés, payez-les bien, traitez-les bien, écoutez leurs problèmes, mais malgré cela n'avez jamais totalement confiance en eux ».

Attention, lorsqu'un informaticien ne fait plus de suggestion de changement de technique ou de formule de programmation, c'est peut-être qu'il a intérêt, pour une exploitation occulte, à ce que rien ne bouge.

LES CLÉS ESSENTIELLES DE LA PROTECTION

— Les plans de sécurité des entreprises doivent constamment être remis à jour.
— Il ne doit pas y avoir libre accès à l'ordinateur. Un contrôle d'accès rigoureux s'impose.

— Les programmeurs ne devraient pas être opérateurs en même temps, tandis que les opérateurs ne doivent pas avoir accès aux documents de programmation de base.

— Des audits informatiques rigoureux doivent avoir lieu régulièrement.

— Les pays Européens doivent réviser leur code pénal pour une meilleure couverture du vol d'information.

— Leurs codes pénaux doivent être harmonisés dans le domaine de la fraude informatique, afin que les fraudeurs ne se protègent pas à travers les transferts des données informatiques par-delà les frontières.

— Les problèmes de la fraude informatique doivent être traités en priorité lors de la planification de l'avenir des structures de notre société.

— La police doit se doter d'un personnel spécialisé dans l'informatique. (Le F.B.I. a déjà créé un département spécialisé).

Les différents aspects de la FRAUDE INFORMATIQUE seront traités en profondeur par SECURICOM'83.

Toulouse

• 22 au 25 février : quatrième Sibso Midi-Pyrénées, au parc des expositions de Toulouse. Renseignements : « Bulletin économique du Midi », 35, bd des Récollets, BP 4074, 31029 Toulouse cedex ; tél. 61/25.02.61.

INFODIAL

Semaine internationale sur les bases et

les banques de données.

lieu : Palais des Congrès, Paris
date : 24-27 mai 83.

Renseignements :

INFODIAL

4, Place de Valois
75001 Paris
Tél. : (1) 261.52.42

**IFIP'83
19-23 Sept 1983
Paris**

PARIS ACCUEILLE
EN 1983
LE CONGRÈS IFIP...

Le SICOB, la CONVENTION INFORMATIQUE et l'AFCEC organisent IFIP'83, le 9^e Congrès Mondial d'Informatique de la Fédération Internationale du Traitement de l'Information (IFIP).

Cet événement exceptionnel qui se tient tous les 3 ans se déroulera à Paris du 19 au 23 septembre 1983. Il rassemblera l'élite internationale de la recherche, de l'industrie et des utilisateurs. L'IFIP renoue ainsi avec ses origines, puisque Paris a vu sa création en 1959 sous l'égide de l'Unesco.

Le SICOB se tiendra du 21 au 30 septembre, sa section informatique constituant l'exposition du congrès.

IFIP'83 est considéré comme la plus importante manifestation mondiale en informatique.

Le programme du congrès comporte 10 thèmes : matériels informatiques et architectures, logiciels, bases théoriques du traitement de l'information, réseaux d'ordinateurs et communications, bases de données et systèmes d'information, systèmes appliqués, systèmes bureautiques, applications des microprocesseurs, implications socio-économiques, les ordinateurs dans la vie courante.

En complément des conférences, 33 tables rondes sur ces mêmes thèmes seront pour les participants l'occasion de dégager les aspects essentiels de l'informatique de demain.

Parmi les nombreux conférenciers attendus, on suivra plus particulièrement les exposés des 41 orateurs d'audience internationale spécialement invités par le Comité de Programme.

La participation au congrès a été fixée à 2 250 F (pour les inscriptions reçues avant le 28 février). Les inscriptions sont à adresser au secrétariat d'IFIP'83 :

6, place de Valois, 75001 Paris
Tél. : (1) 261.52.42

Qu'est-ce que l'IFIP ?

La Fédération Internationale pour le Traitement de l'Information fut créée à Paris en 1960 par décision de 8 Associations Nationales d'Informatique qui avaient travaillé ensemble à la préparation du 1^{er} Congrès Mondial d'Informatique. (1)

Aujourd'hui 42 pays ont rejoint l'IFIP. 9 Comités Techniques et 33 Groupes de Travail couvrent la majorité des domaines de la Science de

l'Information et bénéficie d'une audience internationale.

Les objectifs d'IFIP sont de :

- promouvoir la science et la technologie de l'information ;
- faire progresser la coopération internationale ;
- stimuler la recherche, le développement et les applications du traitement de l'information dans les sciences et les activités humaines ;
- amplifier les échanges dans le domaine de l'information ;
- encourager l'enseignement et la formation.

L'IFIP organise diverses conférences et symposiums, sa manifestation la plus importante étant le Congrès Mondial d'Informatique qui a lieu tous les trois ans dans un pays différent.

L'AFCEC est l'Association Française membre de l'IFIP. Outre l'Informatique, elle couvre les domaines de l'Automatique, de la Bureautique et des Sciences de l'Organisation.

(1) Tenu à Paris en 1959, sous le patronage de l'UNESCO.

DES LIVRES

D. Le Verrand

Le langage ADA Manuel d'évaluation

Éd : Dunod Informatique
- phase spécialité -
DUNOD, 1982, 268 p., 95 F

Nouveau langage de programmation, ADA a été conçu, à l'instigation du Ministère de la Défense des États-Unis, en sorte d'être adapté à une grande variété d'applications : calcul scientifique, applications de gestion, logiciels de base, systèmes en temps réel, logiciels à grande diffusion, etc. (cf. E & I n° 7 Sept. oct. 81).

La conception de ce langage a nécessité un important travail collectif faisant appel à de nombreux experts internationaux du monde de la recherche et de l'industrie.

Certains affirment qu'ADA sera le langage des décennies 80-90 et qu'il supplantera tous les autres. Pour d'autres, ADA est en deça de l'état de l'art actuel. L'objectif principal de cet ouvrage est d'aider les personnes intéressées par le langage ADA à se forger elles-mêmes une opinion reposant sur une connaissance technique et critique de ses différents aspects. Chaque concept du langage est expliqué à l'aide d'exemples et de contre-exemples originaux, et, pour les concepts nouveaux, certains points sont mis en évidence : tâches, "généricité", exceptions...

le langage ADA manuel d'évaluation

Dominique LE VERRAND

Publié sous l'égide de l'AFCEC
avec le concours de
l'Agence de l'Informatique

phase spécialité
DUNOD
Informatique

Tout utilisateur d'un langage de programmation, qu'il ait été contraint de l'utiliser ou qu'il l'ait choisi de son plein gré, est un consommateur et se comporte comme tel. C'est en quelque sorte en tant que « groupement de consommateurs », que le groupe ADA de l'AFCEC, rassemblant plus de vingt membres sous le nom collectif de **Dominique Le Verrand**, propose ici aux informaticiens cette évaluation d'ADA.

Table des matières

1 — Introduction. 2 — Déclarations et types. 3 — Types numériques. 4 — Noms et expressions. 5 — Structures de contrôle séquentiel. 6 — La modularité. 7 — Visibilité. 8 — Les tâches. 9 — Les exceptions. 10 — La généricité. 11 — La compilation séparée. 12 — Moyens d'adaptation. 13 — Les entrées-sorties. 14 — Éléments syntaxiques, lexicaux et textuels. 15 — Conclusion. Annexes : bibliographie et index.

LE VIDÉOTEX GUIDE L'index des services accessibles sur les terminaux vidéotex

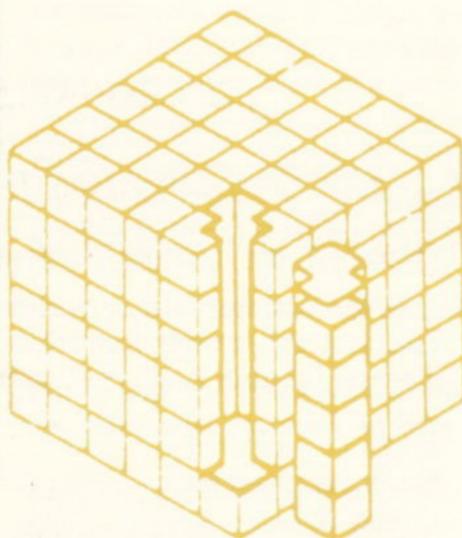
CENTRE D'INFORMATION
DES BANQUES DE DONNÉES
ET DU VIDÉOTEX

11, rue du Marché Saint-Honoré,
75001 Paris. Tél. : 261.45.17/27

Plusieurs dizaines de milliers de terminaux vidéotex seront diffusés en France d'ici la fin de l'année.

Le Vidéotex Guide a recensé pour les utilisateurs l'ensemble des services accessibles par terminal vidéotex. Le guide décrit le contenu de chaque service, et indique les moyens (n° de téléphone) et les conditions d'accès.

Le Vidéotex Guide est aussi accessible en version électronique sur le centre serveur Énergie Vidéotex où il sera mis à jour en permanence. A partir de leur terminal, les utilisateurs pourront connaître les services correspondant à leurs centres d'intérêt.



Le Vidéotex Guide imprimé (qui paraîtra 3 fois par an) et l'accès au service électronique font l'objet d'un abonnement couplé (200 F par an). Rappelons que le Centre d'Information des Banques de Données et du Vidéotex diffuse en France une série de publications et de services spécialisés sur l'évolution

Roland Gersperrin

Apports possibles de l'Informatique à l'École Élémentaire.

Éd. CNDP - CRDP Marseille. 35 F

Commande (accompagnée d'un chèque bancaire ou postal. 3 volets)
- soit à Monsieur l'Agent Comptable du Centre Régional de Documentation Pédagogique. 31, bd d'Athènes. 13232 Marseille Cedex 01.
- soit à Monsieur l'Agent Comptable de l'un des Centres Départementaux de Documentation Pédagogique suivants :
- C.D.D.P. - 8, rue Frédéric Mistral - 84000 AVIGNON
- C.D.D.P. - École Maternelle "Les Ferréols" avenue Maréchal Juin - 04000 DIGNE
- C.D.D.P. - 14, avenue Maréchal Foch - 05000 GAP.

L'auteur, Inspecteur Départemental de l'Éducation Nationale, participe depuis une dizaine d'années à la réflexion sur l'introduction de l'informatique dans l'enseignement.

Il brosse un panorama des différentes pistes sur lesquelles s'engage actuellement la recherche dans le domaine de l'école élémentaire. Ce document s'adressant aux personnes curieuses d'information sur le sujet, sa lecture ne nécessite pas une compétence en informatique. La réflexion s'appuie constamment sur des comptes rendus d'expériences et des documents de réflexion qui occupent les trois quarts de la pagination.

L'informatique est la somme d'une technologie et d'une démarche de raisonnement, l'algorithmique. L'auteur cherche à montrer ce que l'école peut attendre de l'une et de l'autre, en illustrant son propos par des exemples d'utilisation. Il insiste sur les rapports possibles de l'algorithmique à la pédagogie, à un moment où l'on identifie peut-être trop rapidement l'informatique à l'ordinateur.

Au sommaire :

1) Préambule méthodologique

2) L'école face au défi informatique

- I. L'enfant au cœur de l'informatisation de la société
- II. La "seconde génération" de l'école parallèle
- III. "Éducation et informatique : le mariage du siècle"

3) Les objectifs - pour une "Informatique libératrice"

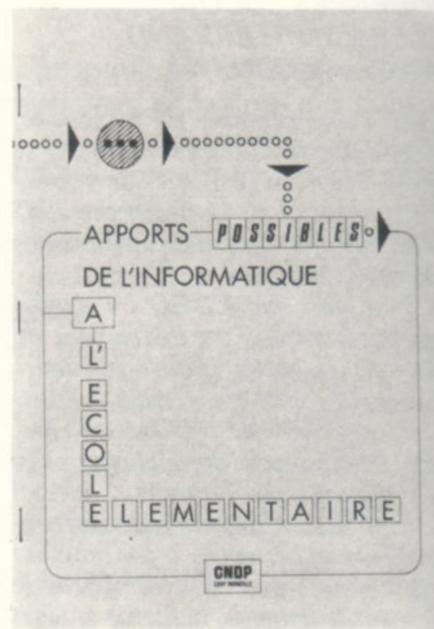
4) les méthodes — les approches possibles

5) questions pour agir

En Annexe, comptes-rendus d'observation

Documents, bibliographie

des banques de données. Il répond aux demandes d'information du public. Énergie Vidéotex est le premier réseau français de publications et services électroniques.



L'ILLUSION D'UNE PANACÉE

La revue américaine ELECTRONICS a publié dans son numéro du 20 octobre dernier, un éditorial sur l'utilisation des ordinateurs dans l'enseignement. Compte tenu de l'audience considérable de cette revue tant aux États-Unis, qu'à l'étranger, il nous a paru intéressant d'en faire part à nos lecteurs. Le texte qui s'intitule "l'illusion d'une panacée", évoque d'abord le choc subi pas les États-Unis, quand a été lancé le premier Spoutnik, et la volonté qu'il a fait naître de battre l'Union Soviétique sur les plans scientifique et technologique. Cette volonté se traduit dans les faits par une amélioration spectaculaire des enseignements scientifiques et techniques.

Alors, le Japon vint. L'opinion s'est de nouveau émue, et s'est penchée sur les écoles. Récemment, D. Packard (l'un des fondateurs de la firme Hewlett-Packard) a demandé que les normes d'éducation américaines se rapprochent de celles du Japon. Et un rapport du Massachusetts Institute of Technology sur la formation permanente des ingénieurs, insiste également sur la nécessité pour les États-Unis de maintenir un "leadership" dans ce domaine de formation.

Les États-Unis étant ce qu'ils sont, on a, comme d'habitude, cherché un remède rapide et qui puisse guérir tous les maux. Et certains pensent l'avoir trouvé : l'ordinateur. Un "show" télévisé a popularisé l'idée que les ordinateurs dans une salle de classe peuvent éduquer les enfants et leur donner un niveau au moins égal à celui des japonais. De toute évidence, ce "show" appelé "FAST FORWARD" ("En avant, toute !"), ignore délibérément le problème du logiciel et du stockage de données.

Les ordinateurs ont un potentiel important en éducation, et peut-être plus encore en formation professionnelle. Convenablement programmés et utilisés sagement, ils peuvent être, après l'imprimerie, la seconde aide à l'éducation qu'on ait jamais inventée. Mais programmés pauvrement et mal utilisés, les ordinateurs (comme la télévision qui fut la panacée précédente), peuvent contribuer à appauvrir, plutôt qu'à enrichir, l'esprit de nos enfants. Comme l'avoue l'éditeur d'une revue de jeux vidéo : "j'ai des craintes pour mon entreprise, car je ne suis pas sûr que les gens qui jouent sachent seulement lire" !

Les quatre professeurs du MIT qui ont étudié la formation permanente des ingénieurs ont reconnu que la technologie en général, et les ordinateurs en particulier, ne peuvent remplacer les relations humaines dans une classe. Si leur rapport recommande l'usage de vidéocassettes, il précise cependant que des enseignants devront toujours être présents.

Aussi évidente qu'elle puisse paraître, cette conclusion mérite d'être rappelée à tous ceux qui développent de la technologie et à ceux qui se préoccupent d'éducation.

La rédaction.



Enseignement Documentation Assistés par Ordinateur

L'E.A.O. est utilisé depuis plus de 10 ans à AIR FRANCE pour la formation du personnel navigant technique (pilotes,...), mais la conception et la réalisation des cours étaient réservées à des spécialistes du système. En 1980, le service de formation de l'informatique a conçu un langage simple d'utilisation, E.D.A.O. — Enseignement et Documentation Assistés par Ordinateur —, permettant à tout auteur de créer des cours rapidement, et dont la structure correspond à une méthode pédagogique, issue de l'analyse comportementale, enseignée depuis plusieurs années aux formateurs utilisant l'E.A.O. en mode tutoriel.

Pierre Tixeuil, maître d'œuvre du système, se consacra d'abord à une analyse des conditions de réussite ou d'échec des systèmes existants. Deux causes d'échecs ont surtout retenu son attention :

- En premier le manque d'écoute, de dialogue et de coopération entre les informaticiens et les utilisateurs. Trop souvent les informaticiens imposent leur point de vue en fonction de considération théorique informatique, et conçoivent un produit technique parfait mais ne répondant pas aux besoins des utilisateurs ; à eux de s'adapter !

- En second le manque de formation pédagogique de l'utilisateur ; on lui fournit une brochure trop technique ou, au mieux, un concepteur présente le système de façon plus technique que pédagogique (aucune information sur l'utilisation pédagogique). Deux points qui amènent l'utilisateur soit à utiliser le système à 50 % de ses potentialités, soit à l'abandonner.

Afin d'éviter ces écueils, nous avons constitué un groupe de travail composé d'un informaticien, Pierre Tixeuil, d'un utilisateur, Nessim Amiaich, et d'un pédagogue Bruno Canale.

Notre analyse a porté essentiellement sur :

- Que demande l'élève, de quoi a-t-il besoin ?
- Que demande l'auteur, de quoi a-t-il besoin ?
- Qu'apporte la télématique à l'E.A.O. ?

De nombreux articles d'Éducation et Informatique ont déjà traité de ces problèmes et nous ne reviendrons pas sur de nombreux points admis par la majorité des spécialistes de l'E.A.O., mais chercherons plutôt à développer ce qui nous a paru important pour E.D.A.O.

L'ÉLÈVE

Notre expérience de dix années d'E.A.O. avec le personnel navigant technique nous a montré que l'élève veut :

- Une réponse rapide à son dialogue, avec commentaire de sa réponse.
- Sentir qu'il progresse au fur et à mesure que le cours se déroule afin que l'enchaînement des articles fasse croître sa motivation.



— Constaté, par la réussite à un exercice final, qu'il a atteint son but.

— Enfin, ne pas avoir à lire de longues pages d'écran, avec comme unique action d'appuyer sur une touche pour tourner ces pages, mais au contraire avoir un dialogue rapide et vivant avec des réponses construites et personnalisées.

Pour ceci, il est nécessaire qu'il dispose de didacticiels de qualité, et c'est là qu'intervient l'auteur.

L'AUTEUR

Pour créer des didacticiels, les auteurs demandent un éditeur de didacticiel simple d'utilisation, facilitant la conception et la mise au point, utilisant un langage naturel exempt de symbole informatique ou mathématique.

Nous avons pensé aussi qu'il était nécessaire de leur enseigner une formation pédagogique aux techniques de l'E.A.O., et afin qu'ils puissent créer, dès les premiers essais, un didacticiel exploitable, et afin d'éviter les longs tâtonnements et parfois l'échec d'une méthode empirique.

ENFIN, LA TÉLÉMATIQUE

Grâce à son réseau de télécommunication de plus de 40 000 km sur lequel sont reliés près de 6 000 termi-

naux, AIR FRANCE peut assurer une formation sur le poste de travail, 24 h sur 24 h, et ceci dans 85 pays.

La logique du système de réservation électronique qui utilise ce matériel permet un dialogue rapide, adapté à l'E.A.O., mais aussi la possibilité pour l'élève de poser des questions aux auteurs. En fonction des fuseaux horaires, les auteurs ne peuvent répondre en temps réel, mais l'élève a sa réponse personnalisée dans les 24 heures.

E.D.A.O. :
Trois parties

Ces réflexions nous ont amené à créer :

- Un cours spécifique E.D.A.O. sur la méthode d'analyse et d'ordonnement propre à ce système
- E.D.A.O. auteur
- E.D.A.O. élève.

MÉTHODE

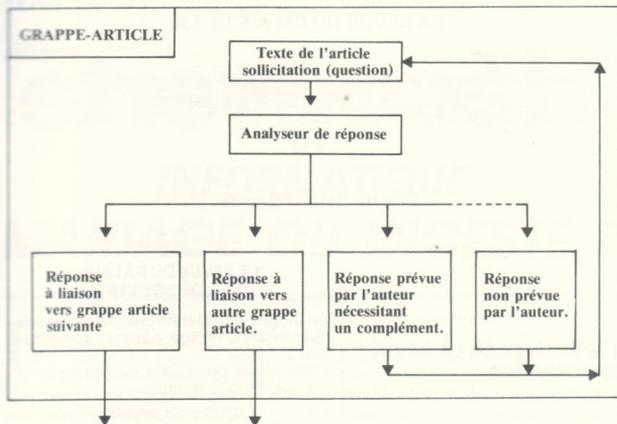
Cette méthode reprend les principes de base de l'E.A.O. développés dans les numéros 1 et 2 d'Éducation et Informatique en les complétant par une méthodologie qui permet de réaliser du premier coup un produit acceptable qu'il suffira ensuite d'améliorer en tenant compte du dialogue avec les apprenants et des statistiques fournis par E.D.A.O.

Elle comporte :

- une étude de l'objectif avec constitution d'un test final,
- une étude de la population avec création d'un test initial,
- une méthode d'analyse décomposant le contenu en notions élémentaires.
- La construction d'un graphe qui permet :
 - de visualiser les cheminement potentiels,
 - de faire ressortir les points de débranchements à partir desquels l'analyse du dialogue orientera l'élève.



ABC de l'E.A.O



— d'indiquer l'emplacement des articles d'information; de découverte, d'apport de connaissance, de récapitulation, d'exercices, de synthèse et de liaison. L'élément de base de ce graphe est l'unité de liaison que nous appelons « grappe article » et qui comprend :

- La partie texte de l'article } Item
- La question
- L'analyse des différentes réponses prévues et non prévues
- Les commentaires attribués à chaque réponse.



L'emploi de cette méthodologie a montré qu'elle était l'élément primordial de la réussite d'un didacticiel. En effet les statistiques faites sur les cours ont montré que seuls ceux respectant les principes de cette méthode sont suivis avec succès par les élèves. Les autres ont dû être enlevés du système, car ils n'étaient jamais utilisés jusqu'au bout.



A partir du graphe l'auteur rédige son didacticiel, il dispose pour cela d'un éditeur de didacticiel ayant la même structure que le graphe et conçu à base de dialogues en langage naturel. Il a aussi la possibilité de piloter un projecteur de diapositive.

Exemple :

Quelle langue désirez-vous utiliser

1. FRANÇAIS
2. ANGLAIS

Votre choix 1

Que désirez-vous faire

1. CRÉER
2. MODIFIER

Votre choix 1

Frappez votre texte - Article 1

Le code de début d'un enregistrement de passage est constitué par les deux lettres de la compagnie assurant le vol. Frappez les deux premières lettres d'un enregistrement sur Air France.

Frappez la réponse n° 1 à l'article n° 1 AF.

Est-ce une réponse à liaison ?

oui
Entrez votre commentaire : Bravo !

Désirez-vous utiliser un dispositif ZAV ?

L'auteur peut ensuite tester son produit, avec l'aide des agents utilisant les 6 000 terminaux. Leurs remarques et les statistiques E.D.A.O. serviront à améliorer le cours. Lorsque l'analyse des résultats montre que le didacticiel est suivi normalement et que près de 90 % des élèves répondent correctement au test final, le cours est mis en libre-service sur le réseau.

BILAN APRÈS UNE ANNÉE D'UTILISATION

Mis en place à la Direction commerciale en 1980, un groupe de 5 formateurs à créé en un an près de 200 cours en français, 95 en Anglais, et 40 en Espagnol, permettant d'assurer une partie de la formation de base des agents commerciaux. Un de leur souci a été d'humaniser le plus possible leur produit en allant présenter eux-mêmes leurs réalisations ou en se servant au maximum des brochures de la compagnie pour présenter les auteurs. Ils mettent aussi un point d'honneur à répondre rapidement aux questions des élèves. Après un an d'utilisation cet ensemble de formation rencontre toujours autant de succès. Les statistiques fournies par l'ordinateur ont montré qu'en un an, 34 180 didacticiels ont été étudiés, soit une moyenne de 94 cours par jour. Plus de cent didacticiels sont actuellement en cours de réalisation.

Cette réussite nous semble provenir de l'étude préalable, mais aussi des constantes rencontres, et des échanges que nous organisons entre responsables, auteurs et élèves et de l'efficacité pédagogique ressentie par les apprenants.

E.D.A.O. DEMAÏN

Un système ne peut continuer à vivre que s'il évolue, c'est pourquoi, nous cherchons à améliorer les performances techniques en étudiant d'autres périphériques (synthétiseurs de voix, images animées, télétraitement pour relier le système à des micro-ordinateurs,...) ou en cherchant à diversifier les méthodes pédagogiques (méthode de la découverte, jeux, amélioration de l'analyseur de réponses,...).

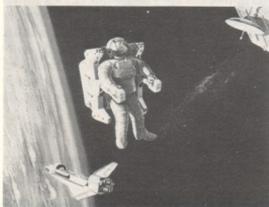
Bruno Canale

Service de Formation de la Direction du Personnel et des Affaires Sociales - AIR FRANCE.

Pierre Tixeuil

Service de Formation de la Direction de l'Informatique - AIR FRANCE.
Pour tout renseignement complémentaire, écrire aux auteurs DP-GF 91300 MASSY.





QUELQUES-UNS DES NUMÉROS SPÉCIAUX DE LA REVUE DU PALAIS DE LA DÉCOUVERTE

- **A la découverte de la terre**, 1974 : publié à l'occasion de photographies spatiales — réalisé par Kodak-Pathé avec Albert Ducroca et le concours de la NASA — prix : 6,00 F Franco.
- **Les courbes mathématiques**, 1976 : courbes algébriques ornementales transcendentes, autant d'illustrations, par J. Brette — prix : en cours de réimpression.
- **Découverte de l'univers**, 1978 : le système du monde de l'antiquité à Copernic. Le système solaire... par G. Oudenot — prix : 23,00 F Franco.
- **La conquête de l'espace**, 1980 : historique, médecine, biologie. Applications scientifiques. Texte des Actes du congrès de l'Association Française pour l'Avancement des Sciences — prix : 39,00 F Franco.
- **Les bases scientifiques de l'amélioration des ressources alimentaires**, 1981 : une mise à jour des problèmes les plus importants avec la participation de nombreux spécialistes de l'Université et de l'industrie. Les Actes du congrès de l'Association Française pour l'Avancement des Sciences — prix : 50,00 F Franco.
- **Laënnec**, 1981 : Laënnec professeur, médecin, humaniste parisien. La diffusion de son œuvre à l'étranger, avec la participation de nombreux spécialistes dont J. Dausset, Y. Laporte, J. Bernard. Actes du colloque organisé au Collège de France à l'occasion du bicentenaire de la naissance de Laënnec — prix : 49,50 F Franco.

LA REVUE DU PALAIS DE LA

découverte

vous intéresse !

LA REVUE DU PALAIS DE LA DÉCOUVERTE

Un magazine scientifique pour des non spécialistes, rédigé par des spécialistes.

Cette revue mensuelle est entièrement consacrée à la science.

VOUS Y TROUVEZ
CHAQUE MOIS

- l'actualité scientifique
- le commentaire des expositions temporaires.
- le texte des conférences du samedi : parmi nos titres :
 - **Jean Rostand, auteur satirique** par A. Tetry.
 - **Einstein : La science à la recherche d'une éthique** par M.A. Tonnelat.
 - **L. de Broglie et la coexistence des ondes et corpuscules** par G. Lochak.
 - **Des robots intelligents** par J.P. Hermann.
 - **Zootéchnie et santé de l'homme** par le Pr R. Ferrando.
 - **Microbiologie et hygiène des denrées alimentaires** par le Pr M. Castaras.
 - **La bale du Mont-Saint-Michel** par C. Larssonneur.
 - **Les vols spatiaux** par A. Ducroca, le Pr G. Meyniel, J. Collet, A. Dupas.

BULLETIN D'ABONNEMENT

E et I n° 14

REVUE DU PALAIS DE LA DÉCOUVERTE

NOM _____ PRENOM _____
(Lettres capitales)

ADRESSE _____

PROFESSION _____

MATHÉMATIQUE ET INFORMATIQUE MARIAGE OU INCESTE ?

Les mathématiques et l'informatique entretiennent depuis leur rencontre des rapports ambigus que de nombreux exégètes ont essayé d'analyser dans l'espoir d'en préciser la nature incestueuse.

On trouvera les traces de ce débat racien dans les trois articles concluant le dossier : « Mathématicien et/ou informaticien » où le professeur NIVAT nous précise ce que cache pour lui et aujourd'hui le mot « informatique ». Au delà des mots, on sait bien, dans les faits, que l'apparition des matériels et des méthodes informatiques apportent quelques changements à la discipline mathématique et à son enseignement. Ces changements sont au moins de trois ordres :

— L'ordinateur gère l'apprentissage d'un savoir donné ; c'est un outil de présentation qui n'interfère pas de façon significative avec le contenu de ce savoir.

— L'ordinateur permet une présentation nouvelle, et modifie le rapport entre l'apprenant et la discipline ; par exemple, la mathématique devient illustrée et expérimentale.

— L'ordinateur modifie la nature même du savoir enseigné : certaines parties de ce savoir (par exemple tout ce qui touche les algorithmes opératoires ou les fonctions et les suites) prennent une importance nouvelle.

Cette tripartition est évidemment caricaturale et l'on sait bien qu'il est difficile d'isoler tel ou tel aspect dans chaque interaction particulière. En parcourant ce dossier « maths », le lecteur reconnaîtra certainement les échappées transversales à cette classification...

Le premier aspect regroupe les productions de l'EAO plus ou moins classique et de son cortège de didacticiens (voir articles p. 20 « Où en est l'EAO en maths » et p. 29 « il était une fois un didacticiel »).

Le deuxième aspect semble porteur d'un germe de changement plus radical

en mathématiques que dans les autres disciplines : une majorité (bien mal informée) voit, en effet, les mathématiques comme la manipulation d'un ensemble d'objets détachés du concret et sans espoir de représentation ; peut être la production massive d'images, d'actions et de calculs par ordinateurs permettra-t-elle de rendre manifeste ce qui n'est aujourd'hui qu'une conviction minoritaire : les objets, les structures et les algorithmes mathématiques ne sont pas des irréalités éthérées mais des objets d'expériences et d'exploration ! (voir les articles p. 23 « quelques exemples d'enseignement des maths illustrés par ordinateur », p. 26 « KIJOUAKOUA » et p. 27 « La mathématique, science expérimentale »).

Ainsi les nombres sont-ils devenus

beaucoup plus présents et familiers aux enfants depuis qu'ils peuvent manipuler des calculatrices de poche ; bien loin de les éloigner du calcul, il est maintenant prouvé, au contraire, que l'intérêt pour les activités numériques augmente avec leur manipulation et donc aussi avec la volonté d'être efficace et d'apprendre les règles.

Finalement, les gosses qui disposent régulièrement d'une calculatrice de poche connaissent un peu mieux (statistiquement) leur table de multiplication et calculent un peu mieux mentalement que ceux qui n'en ont pas. En sera-t-il de même avec les nouvelles fonctions qui apparaissent sur de petites calculatrices — « solutions » d'une équation, intégrale d'une fonction sur un intervalle —, indépendamment de la confiance que l'on peut avoir dans les résultats ? Que va donc devenir la pratique du calcul algébrique avec des logiciels comme « MU-MATHS » (voir article p. 34 « Ordinateurs et calcul symbolique ») ?

Tout dépend évidemment de la facilité d'utilisation des futurs matériels et de l'importance de leur diffusion. Gageons cependant qu'il faudra être attentif et vigilant et préparons nous à rester à la fois critiques et optimistes : tant qu'il y aura des problèmes et des individus pour chercher à les résoudre (voir article p. 31 « Conjectures et résultats mathématiques »), tant qu'il y aura le désir de mettre en œuvre et en concurrence les outils et les techniques du passé et du présent, tant qu'il y aura des hommes...

André Deledicq

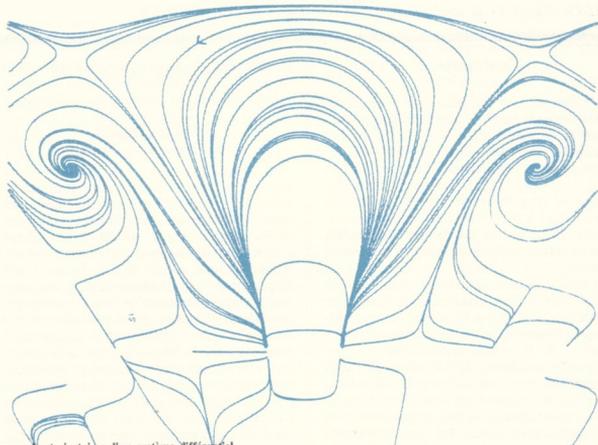


Image des trajectoires d'un système différentiel.

Extrait du livre « Systèmes différentiels - Étude graphique »

Véronique Gautheron et Michèle Artigue - éd. CEDIC - 1^{er} trim. 83

$$\begin{aligned}x' &= \sin(x) \sin(y) \\ y' &= \cos(xy)\end{aligned}$$

OÙ EN EST L'E.A.O. EN MATH ?



Il y a plus de 10 ans que l'on recherche, expérimente et pratique l'informatique dans l'enseignement. Au tout début sans ordinateur, puis avec des mini-ordinateurs (expérience des 58 lycées) et, depuis 1980 des micro-ordinateurs.

Les observations se sont accumulées, et il existe déjà une "histoire de France" de l'E.A.O. - avec des dates importantes (1970, 76, 80, ...) - de nombreuses publications et une bibliothèque conséquente de logiciels "Éducation Nationale". Rappelons que cette bibliothèque, gérée d'abord dans la phase expérimentale des 58 lycées par l'I.N.R.P., l'est maintenant par le C.N.D.P. qui la diffuse gratuitement, modulo le prix du support*. Les logiciels composant cette bibliothèque, rédigés en L.S.E. sont l'œuvre d'enseignants (1)

En mathématiques, le travail a été important et a dépassé l'écriture de didacticiels : des recherches ont été menées par exemple sur l'algorithme et la programmation de problèmes divers par les élèves. Un bilan, dressé en 1980 (2), a montré entre autres que les mathématiques venaient en tête pour l'utilisation de logiciels pédagogiques. Avant de faire un point rapide sur la situation actuelle, il est peut-être utile de rappeler un certain nombre de faits liés au contexte de fabrication et d'utilisation de logiciels pédagogiques.

Confluence...

La fabrication d'un didacticiel de mathématiques se trouve au confluent (pour ainsi dire) de plusieurs disciplines : Les mathématiques, leur didactique, l'informatique. Expression d'une volonté pédagogique, il doit, une fois réalisé "tourner" avec des élèves, et de préférence leur être utile. Les savoir-faire à maîtriser sont donc très nombreux. Dans la plupart des cas, le produit fini ne reflète qu'imparfaitement la volonté initiale.

On doit rappeler également que, jusqu'à cette année, les matériels "Éducation Nationale" n'ont pas bénéficié de possibilités graphiques. Il en va différemment maintenant, et il existe depuis mai 82 une maquette graphique L.S.E. dont l'expérimentation doit conduire à la définition d'un L.S.E. traitant des objets graphiques. Enfin mentionnons le fait que l'utilisation de didacticiels avec les configurations actuelles (une salle avec 8 micro-

ordinateurs par lycée équipé) est forcément occasionnelle : 10 % du temps scolaire-élève sont dans la pratique un maximum.

Du traitement au dialogue interactif

Historiquement, une des premières tendances a été d'utiliser l'ordinateur comme un outil de traitement de données numériques (45% des logiciels de la bibliothèque I.N.R.P.), sans analyse de réponses-élèves. Effectivement, c'était le plus facile. Cette approche est certes intéressante : elle peut par exemple permettre de "voir" converger des suites, d'étudier expérimentalement des fonctions, de résoudre des équations... (voir encadré 1) Mais elle peut, à la limite, être réalisée par des machines programmables, et ne tient pas compte des possibilités de "dialogue interactif" qu'autorisent les ordinateurs. (Ce qui n'est pas une raison pour ne pas l'employer chaque fois qu'elle est utile).

Une autre approche, parallèle, a été basée sur l'aide que l'on peut apporter aux élèves

en faisant simuler par logiciel un dialogue maître-élève sur certains points de cours. Il est bien connu en effet qu'en mathématiques, les élèves ont dans l'ensemble de grandes difficultés à maîtriser des savoirs opératoires simples. Certaines erreurs sont des "classiques" et reviennent constamment chez certains élèves qui sont, dans ces conditions, très handicapés (3).

On a donc pensé que des programmes posant des exercices sur des points particuliers, renvoyant à l'élève des indications sur ses résultats, le félicitant, ou le blâmant, étaient susceptibles de le faire progresser.

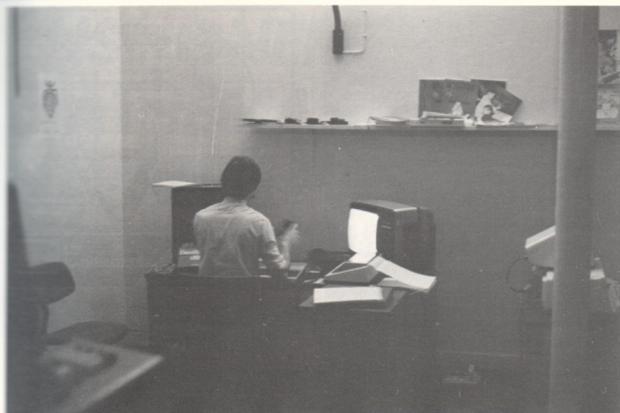
Ainsi 45 % des logiciels de l'ancienne bibliothèque I.N.R.P. (et plus de 3/4 des utilisations effectives avant 1980) ressortissent à cette catégorie. Ce sont très souvent des logiciels répétitifs qui posent des exercices de types répertoriés, à données numériques aléatoirement générées.

Bien entendu, certains sujets sont très abondamment représentés (calculs numériques divers, résolution d'équations de degré 1 et 2...) et d'autres très peu : (dérivation, continuité.)

Vers une adaptation individualisée

Cette orientation, qui vise à aider les élèves en difficulté en leur permettant de s'entraîner sur les points où ils sont faibles, est plus actuelle que jamais. Le logiciel et la machine y prennent en charge la fonction de renvoyer à l'élève une certaine image de l'état de ses connaissances sur un sujet donné, ainsi que des indications pour progresser ; c'est ce que nous avons appelé par ailleurs un "effet miroir"(5). Il est donc important que le logiciel garde une certaine crédibilité en ne comptant pas fausses des réponses justes, en ne diagnostiquant pas

* dans les établissements publics équipés.



de travers, en ne rabâchant pas à l'élève des points de cours qu'il sait déjà, et bien sûr en ne l'humiliant pas par des jugements hâtifs ou de l'humour perfide auquel il n'a pas la possibilité de répliquer. (Tous travers déjà effectivement rencontrés, heureusement rarement).

Cependant il faut bien reconnaître que subsistent encore des difficultés pour reconnaître des réponses données en langue naturelle (c'est pourquoi on a souvent tendance à les « discrétiser ») et surtout pour porter des diagnostics pertinents sur les erreurs et les savoirs de l'élève.

Pour cela en effet, il faut disposer d'un "bon" modèle de l'élève, avec des procédures d'évaluation permettant de juger sur quels points il est faible; de plus, il faut avoir défini des procédures qui, à partir de ce "profil" de l'élève, pourront proposer une action adaptée : ne rien dire, aider, adapter le type et le niveau de l'exercice... cela est loin d'être évident.

Il s'ensuit que l'on ne dispose pas de logiciels assez souples pour s'adapter à chaque élève en particulier : s'il a tout bien compris, l'élève risque de s'ennuyer; s'il est très faible, le logiciel ne peut rien pour lui. Classiquement alors, il s'énervé, tape des obscénités, éteint la machine... L'enseignant doit alors intervenir pour apporter une explication adaptée.

Rappelons que les études menées de 1977 à 1979 par l'INRP ("classes expérimentales") ont montré

1) que se sont les élèves "moyens-faibles" qui tirent le plus de profit du soutien individualisé, revenant parfois travailler en libre-service, surtout si le logiciel leur fournit une certaine évaluation de leurs travaux (un score).

2) bien que les élèves travaillent à leur rythme et individuellement, on n'a pas réussi à les rendre complètement autonomes.

Notons que la taille des logiciels a une tendance à croître. Mais de nombreux enseignants déclarent se satisfaire de petits logiciels assez élémentaires, relatifs à des points très précis et organisés en ligne, ou en gamme coordonnée. (cf. encadré 2)

Les langages d'auteurs

Les langages d'auteurs déchargent l'enseignant d'une partie de la programmation et lui fournissent des primitives puissantes d'analyse de réponse. Ils lui permettent de se concentrer sur l'analyse didactique, de changer facilement les formulations des questions et des commentaires, de gérer sans peine la page-écran. Ils sont utilisés depuis longtemps; le système ENSPI par exemple a fait l'objet de travaux dès 1975, et l'on peut citer — entre autres — Arlequin, mis au point par des enseignants, et dont la première version a été expérimentée par l'I.N.R.P.

Un point pour l'instant un peu délicat de ces systèmes est leur relative lenteur d'exécution (l'analyse de réponse prend du temps). Par ailleurs, et c'est peut être plus fondamental, un graphe des possibilités a été établi a priori, et on ne peut que l'explorer. On touche ici une des limites des types d'EAO mentionnés plus haut.

Des logiciels intelligents ?

Cela fait des années que l'on cherche à pallier aux problèmes évoqués plus haut, à modéliser plus finement une relation maître-élève sur un point précis et à conférer au logiciel une certaine capacité de suivi et de guidage de l'élève. Des recherches ont été menées dans le cadre de l'intelligence artificielle et ont conduit à des réalisations impressionnantes et... inaccessibles pour l'instant aux enseignants du secondaire.

Par exemple BUGGY (4) est capable de synthétiser automatiquement un modèle des "savoirs déviés" d'un élève sur les opérations de base, et à partir de ce modèle peut, non seulement reconnaître les erreurs "de surface" (les symptômes) mais encore en inférer pourquoi il fait telle erreur. Il fait appel à des représentations sophistiquées des savoirs, en jeu en arithmétique élémentaire.

L'idée qui s'est ainsi dégagée est que le logiciel, pour pouvoir diagnostiquer des erreurs et suivre correctement l'élève, doit être lui-même un expert du domaine considéré :

savoir résoudre comme un expert humain et connaître les "savoirs déviés" les plus fréquents et leur hiérarchie...

Il existe en France des équipes universitaires travaillant dans cette direction; on peut citer par exemple les travaux de M.O. Cordier, D. Kayser, A. Bonnet (Orsay) qui reprennent l'idée d'expertise nécessaire du système dans lequel ils distinguent deux parties : "l'expert" lui-même, et "le pédagogue", module chargé de gérer les interactions avec l'élève et de prendre les décisions pédagogique. Ce module tient à jour un profil de l'élève.

Un point très important est que dans les différentes parties du système on prend soin de séparer les connaissances nécessaires (la base de faits) du contrôle de l'interprèteur qui les utilise. Ces connaissances (règles de production, règles de réécriture) sont donc des données du programme.

Évidemment, il faut avoir préalablement exprimé de façon explicite les savoirs en jeu pour pouvoir faire fonctionner le système (ce n'est pas facile !)

Une autre utilisation : Le retour au collectif

Un domaine encore expérimental est celui de l'utilisation collective d'un micro-ordinateur à possibilités graphiques qui intervient à la demande comme ressource supplémentaire dans le cours pour fournir à l'ensemble des élèves des images dynamiquement modifiables de situations mathématiques.

C'est ce que A. Deledicq appelle l'approche "imagicielle". Une recherche est en cours à l'INRP en liaison avec le CNAM et l'IREM de Paris Sud sur ce sujet.

L'utilisation est entièrement à inventer; les premières observations montrent que si les élèves ne sont pas simplement spectateurs mais peuvent intervenir, cela « marche »; si en revanche on les laisse assister à un spectacle, l'intérêt faiblit assez vite.

Une conclusion ?

En 1982, l'expérience le montre, l'utilisation de l'ordinateur a droit de cité dans l'enseignement des mathématiques. L'intervention informatique, utilisée en complément du cours "classique" peut assurer très localement une aide, et favoriser l'appropriation de certains savoirs (notamment opératoires).

Mais beaucoup reste à faire; le rôle de la recherche, en particulier, est considérable. Il est difficile de se hasarder à des paris sur ce que sera l'avenir, trop de paramètres sont en jeu, du matériel à la formation des enseignants en passant par la structure des classes et le contenu des programmes scolaires.

Il est souhaitable de pouvoir disposer, en utilisation individuelle, de logiciels « intelligents » ; et on peut rêver de systèmes généraux acceptant en données des règles pédagogiques, des savoirs mathématiques, des faits sur l'élève, et le faisant progresser, pendant de longues périodes de temps. Peut être...

Une chose est certaine, pourtant. On ne pourra pas faire d'économies sur la réflexion pédagogique.

Georges-Louis Baron
Département des Actions Pédagogiques
en Développement
DAPED — cellule Informatique
Direction des lycées

(1) Éducation et Informatique numéro 13 a donné la liste de ces didacticiels

(2) "Dix ans d'informatique dans l'enseignement secondaire 1970-1980", Recherches Pédagogiques. N° 113 - Paris INRP

(3) Voir par exemple, p. 29

(4) *Diagnostic Models for procedural bugs in basic Mathematical skills*: John Seely Brown, Richard R. Burton, *Cognition Science*, 2 1978 pp. 155-192

(5) Effet miroir en E.A.O. - Communication française au WCCE Lausanne - 81 - North-Holland éd.

Encadré 2

Un exemple de mini-didacticiel : AFF1

Ce programme, mis au point par une équipe d'enseignants du lycée Balzac à Paris*, traite de la représentation de fonctions affines au niveau Seconde, avec mise en évidence de l'alignement des points de la représentation graphique. Il est utilisé en liaison avec d'autres logiciels sur la représentation graphique d'une fonction en général (intérêts et dangers du tracé d'une courbe point par point.)

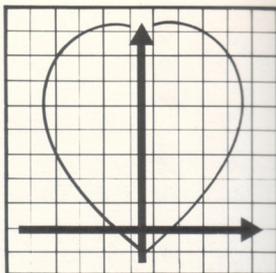
Son utilisation prend une heure-élève.

Le programme génère aléatoirement une fonction affine, pour laquelle il demande :

- 5 calculs d'images de réels.
- 5 tests d'appartenance au graphe de la fonction de couples (x,y)
- 5 calculs d'antécédents.

On demande à l'élève de marquer en rouge sur une feuille à part les points appartenant à la représentation graphique de la fonction, en bleu les autres. Il remarque que les points rouges sont alignés. L'élève a la possibilité de refaire un exercice du même type.

*M. Collomer, F. Pettré, N. Rodriguez



*Si votre passion
d'Euclide n'a
d'égale que celle
de Gauss...*

LA PETITE ENCYCLOPÉDIE DES MATHÉMATIQUES

Sous la direction du
Professeur J.L. LIONS
du Collège de France

Un ouvrage de référence destiné à tous les publics, pour comprendre en trois niveaux progressifs l'évolution des théories mathématiques :

- 1- Mathématiques élémentaires.
- 2- Mathématiques supérieures.
- 3- Mathématiques appliquées aux sciences et aux technologies modernes.

- 900 pages, illustrations couleurs (graphiques, dessins...)
- Prix de lancement : 250F

En vente chez

 **Didier**

Nouvelle adresse

13, rue de l'Odéon 75006 Paris.

Documentation détaillée sur simple demande tél: 329.95.50

On sait que la suite $U_n = (1 + \frac{1}{n})^n$ converge vers e si n tend vers +∞

Le petit programme suivant illustre bien le fait que si on arrive à une «bonne» approximation de e pour n grand, mais voisin de 1000 environ, en revanche pour n très grand, la machine en vient à négliger $\frac{1}{n}$ devant 1.

```

programme
  1 AFFICHER 'N':LIRE N
  3 AFFICHER 'PAS D'AFFICHAGE ';LIRE P
  4 AFFICHER 'A PARTIR DE ';LIRE NB
  5 FAIRE 7 POUR I=NB PAS P TANT QUE I<=N
  6 U=(1+1/I)^I
  7 AFFICHER 'U','I','=',U
  10 TERMINER
  EXECUTER A PARTIR DE 1
  N=1000
  PAS D'AFFICHAGE 100
  A PARTIR DE 100
  U100 =2.70481
  U200 =2.711575
  U300 =2.713754
  U400 =2.714945
  U500 =2.715686
  U600 =2.716032
  U700 =2.716396
  U800 =2.716638
  U900 =2.716568
  U1000 =2.716718
  TERMINE EN LIGNE 010
  
```

On donne ci-dessous l'exécution pour $n = 10^7$ qui fournit des résultats aberrants.

```

EXECUTER A PARTIR DE 1
N=1E7
PAS D'AFFICHAGE 1E6
A PARTIR DE 1E6
U1000000=2.595218
U2000000=2.595219
U3000000=2.044704
U4000000=2.595219
U5000000=3.293954
U6000000=4.180816
U7000000=5.306455
U8000000=6.735161
U9000000=1
U10000000=1
TERMINE EN LIGNE 010
  
```

ENSEIGNEMENT DES MATHÉMATIQUES ILLUSTRÉ PAR ORDINATEUR

quelques exemples

DOSSIER

Le terme d'«Enseignement Assisté par Ordinateur» (abrégié EAO) s'est trouvé spécialisé dans l'utilisation des ordinateurs de manière individuelle par l'élève, en général utilisant un logiciel du type questions-réponses. Il est possible cependant d'utiliser un ordinateur en classe comme source d'informations à usage collectif et ceci s'est fait depuis de nombreuses années. Depuis que les ordinateurs graphiques se sont généralisés à des prix «abordables», on peut leur demander d'être une source d'images ou de suites d'images destinées à illustrer le cours ou les exercices, en particulier en Mathématiques. C'est ainsi qu'on pourrait parler d'«Enseignement Illustré par Ordinateur» (en abrégé EIO). Pour ne pas alourdir cet article nous n'aborderons ni une étude comparative des moyens d'illustration (dessins, diapositives, films, magnétoscopes, tableau noir, ordinateur, etc.), ni les problèmes de technique informatique (matériel, programmation...), ni les problèmes d'évaluation de l'efficacité pédagogique de l'EIO. Nous nous limiterons à quelques exemples simples pris parmi beaucoup d'autres qui font partie d'un ensemble d'études et de recherches menées au C.R.E.E.M.* et à l'I.R.E.M.* de Paris VII grâce à l'A.D.I.* et l'I.N.R.P.*.

Tracé de courbes

Un domaine privilégié de l'utilisation d'un ordinateur graphique est naturellement celui du tracé des courbes. Nous nous limiterons ici à quelques commentaires sur cette application dans le cadre de l'enseignement des Mathématiques illustrées par ordinateur.

A titre d'exemple, nous allons évoquer très sommairement quelques questions soulevées par la réalisation d'un «bon» programme de tracé de courbes représentatives de fonctions destinée à être utilisée en classe. Ici, comme souvent, le matériel a une grande importance ; il est évident que la qualité de la représentation est fonction croissante de la définition graphique (nombre de points affichables). De même la présence de la couleur est un atout quand on

veut représenter simultanément plusieurs fonctions. Il faut cependant signaler que le temps de construction de la courbe est grosso-modo proportionnel au nombre de points utilisés en horizontal ; ceci peut être gênant en utilisation en temps réel en classe, quand on veut tracer toute une famille de fonctions.

Le matériel étant fixé, plus ou moins performant, le logiciel aura besoin d'un certain nombre de données pour fonctionner et son exécution dépendra d'un ensemble de choix :

- la ou les fonctions à représenter. Il faut les entrer dans la machine, d'où problèmes d'écriture et de syntaxe.
- des options de représentations du type : cadrage (qui peut être automatique, assisté ou manuel), représentation des axes de coordonnées, graduations de ces axes, choix des couleurs s'il y a lieu, jonctions des points, traitement des sauts et des points hors écran, cas de non définition de la fonction, possibilités de tracer plusieurs courbes, de changer de cadrage, etc.

En fait il nous semble que, pour l'application qui nous intéresse ici, il n'y a pas de choix miracle. Ce sera le contexte d'utilisation qui commandera le type de logiciel adapté. Nous avons simplement remarqué qu'un programme très sophistiqué, offrant

tout un menu d'options, est d'un usage malcommode en classe par le nombre de réponses à fournir (sans se tromper !) avant d'obtenir la courbe à l'écran. Ainsi nous semble-t-il préférable, malgré le travail supplémentaire de préparation que cela demande, d'utiliser, pour une séance précise, une ou des versions spécialisées du même programme avec des options bloquées à l'avance (par exemple cadrage, couleurs, axes) afin de ne garder que la possibilité de faire varier les paramètres dont nous voulons montrer l'influence.

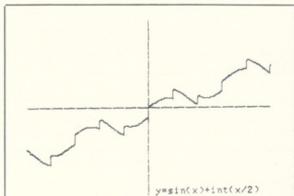


Fig. : 1

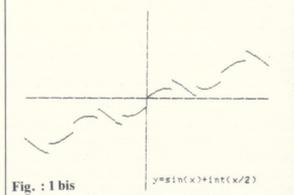


Fig. : 1 bis

La même fonction avec deux options différentes : points joints et points non joints.

Pour revenir au tracé de courbes en classe de façon plus générale, il est souvent nécessaire d'écrire des logiciels particuliers quand le sujet est concentré sur un type de courbes particulières.

Par exemple, pour illustrer une leçon ou des exercices sur les équations de droites en classe de 3^e, on n'utilisera pas un programme général de tracé de courbes, qui est trop lourd et trop lent pour montrer clairement l'influence de la variation des coefficients de l'équation sur la position de la droite. C'est ainsi que nous avons écrit des programmes destinés à n'illustrer que ce point et donc où l'utilisateur ne commande que cette variation des coefficients (auteurs : MM. Petresco et Simon). Ces programmes sont très rapides à l'exécution et permettent ainsi une sorte d'animation qui peut souligner le phénomène à montrer (exemple : pivotement de la droite $ax + by + c = 0$ quand b seul varie).

Les mêmes idées ont été reprises pour la

*CREEM : Centre de Recherches et d'Expérimentation sur l'Enseignement des Mathématiques (C.N.A.M. Paris)

*IREM : Institut de Recherches sur l'Enseignement des Mathématiques (Université de Paris VII)

*ADI : Agence de l'Informatique (Ministère de l'Industrie et de la Recherche)

*INRP : Institut National de Recherche Pédagogique (Ministère de l'Éducation Nationale)

représentation des trinômes (programme écrit par M. Simon), des coniques, de la résolution graphique d'une équation etc.

Fig. : 2

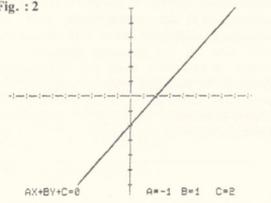
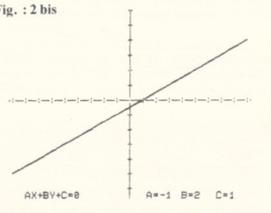


Fig. : 2 bis

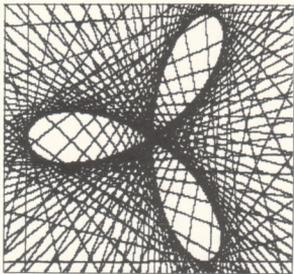


Un cinquième de seconde sépare les deux images (programme de J.M. Simon).

Quand il s'agit de tracer une famille de courbes, la sortie graphique sur écran est quelquefois trop confuse et il faut alors utiliser une table traçante. Il est cependant possible de représenter certaines familles simples sur un écran : par exemple programme de tracé de courbes tangentielles où l'on choisit les coefficients d'une droite dépendant d'un paramètre (utilisable dès la classe de troisième).

Enfin les tracés de surfaces sont possibles, mais les programmes dont nous disposons sont assez lents, donc peu utilisables en classe.

Fig. : 3



Enveloppe d'une famille de droites

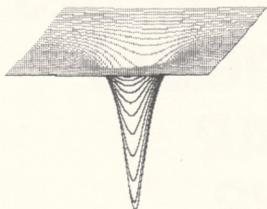


Fig. : 4

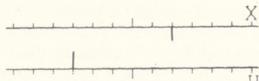
Tracé de surface (temps de tracé environ 7 minutes)

Représentations des fonctions

A côté de la représentation graphique classique des fonctions numériques d'une variable numérique, l'ordinateur graphique permet certaines représentations moins habituelles. Il s'agit tout simplement de faire "fonctionner" la fonction ou les fonctions à représenter en profitant de la rapidité de calcul de l'ordinateur qui peut par exemple montrer la variable et son image sur deux axes parallèles.

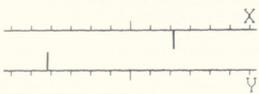
$$Y = -2X^2 + 2X + 1$$

Fig. : 5



$$Y = -2X^2 + 2X + 1$$

Fig. : 5 bis



Entre ces deux figures, l'utilisateur a appuyé deux fois sur la flèche droite du clavier : x a faiblement augmenté, y a fortement diminué (temps écoulé : deux dixièmes de seconde)

Ce type de représentation est beaucoup plus "rustique" que la courbe habituelle mais peut permettre de souligner autrement certains aspects de la fonction (extremum, saut, dérivée...). Elle demande un ordinateur rapide dans ses calculs et ses affichages car elle présente un côté cinématique : il peut être nécessaire d'avoir des déplacements continus (à l'œil) de x et de y, au moins quand la fonction est continue (ne serait-ce que pour souligner les sauts).

Ce type de représentation est applicable à toutes sortes de fonctions numériques et permet quelquefois de "montrer" la fonction et ses variations sans trop alourdir le dessin, en combinant éventuellement avec un affichage numérique (1).

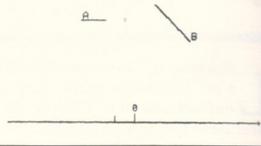
Fig. : 6



AIRE = 9.6

Représentation de l'aire d'un triangle : le côté horizontal est fixe, la variable est le dernier sommet pilotable par des commandes simples (ici on a combiné les deux types d'affichage : numérique et sur un axe vertical)

Fig. : 7



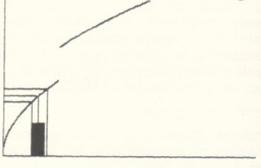
Représentation du produit scalaire sur un axe : ici la variable est un couple de vecteurs dont les directions et les longueurs peuvent varier.

Sur la continuité

Nous voyons déjà, à propos des fonctions, que l'ordinateur peut aider à représenter des objets mathématiques non seulement géométriques (comme des courbes, des surfaces etc.) mais encore "abstraites" comme des applications. Un exemple typique, qui correspond à un réel problème d'enseignement, est la notion de continuité (et de limite) en analyse. Que peut l'ordinateur sur ce point ? Nous ne pouvons pas donner de réponse complète mais indiquer quelques tentatives sous forme de programmes. Bien sûr, le seul fait de montrer aux élèves des représentations de fonctions continues et discontinues est indispensable mais ne suffit pas à faire comprendre, sinon la nécessité de la formulation mathématique, du moins sa signification.

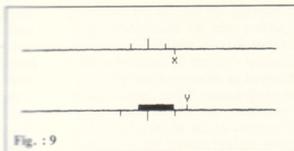
Ceci est l'objet de nombreux logiciels, tant en France qu'à l'étranger qui jouent sur les "epsilons" et les "alphas". C'est dans cette ligne classique que certains logiciels ont été réalisés en utilisant cette fois le graphique (ceux écrits par L. PUEL, C. BARBÉ, J.H. SAÏAC par exemple)

Fig. : 8

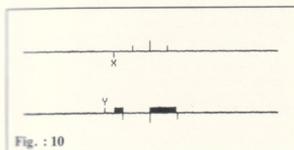


L'utilisateur choisit epsilon et la machine montre un alpha (la partie noire correspond à un remplissage progressif rappelant l'intervalle en alpha sur la courbe).

(programme : J.H. Saïac)



En gras, sur l'axe du bas, l'image de l'intervalle en alpha par la fonction. Ici cet alpha convient pour cet epsilon.

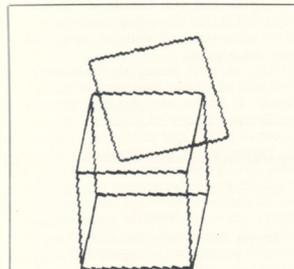


Ici l'alpha choisi ne convient pas (et aucun ne pourra convenir pour ce choix d'épsilon)

Représentation d'objets de l'espace

Mouvement

Il est bien connu des professeurs de Mathématiques que leurs élèves ne "voient" pas dans l'espace, c'est-à-dire sans doute ne comprennent pas les représentations planes d'objets à trois dimensions. L'ordinateur peut apporter ici une aide substantielle par sa faculté de dessiner rapidement plusieurs représentations du même objet, par exemple lors d'une rotation. L'expérience montre que dans ce cas, à condition que la rotation ne soit ni trop lente ni trop rapide, l'effet d'animation contribue à faire "voir" l'objet et non plus sa projection. De plus le fait de mouvoir l'objet permet d'en choisir des positions particulières donnant des projections intéressantes. Des objets composés de plusieurs éléments (deux segments, un rectangle et un segment, les faces d'un cube ou d'un tétraèdre...) peuvent être aussi animés, chaque élément étant mobile. Le mouvement ici améliore nettement la compréhension de la représentation.

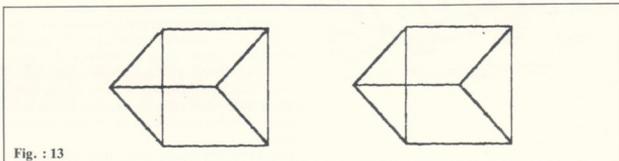


Cube dont une face a été déplacée : noter l'ambiguïté de la figure !
La face est mobile ainsi que l'ensemble, ce qui permet de s'y retrouver en tournant.

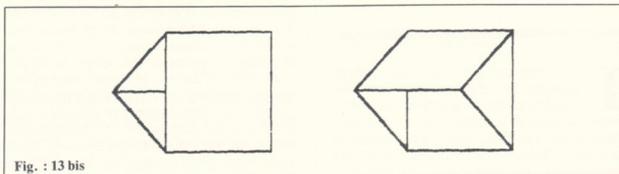
Transparence, ambiguïté des représentations

Donnons enfin un exemple très simple, mais soulignant la commodité de l'ordinateur. La transparence des faces d'un objet

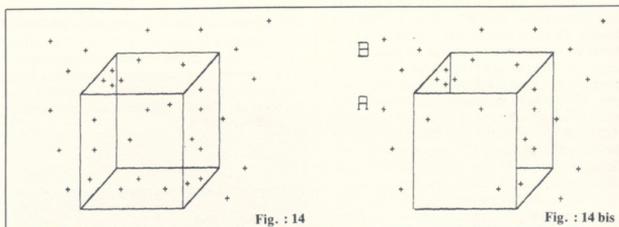
est une cause d'incompréhension de sa projection (d'où l'utilité des pointillés !). En jouant sur ce phénomène on peut réaliser facilement des logiciels combinant des problèmes de déduction logique et de géométrie (idée de J. Petresco)



Toutes les faces sont transparentes sur les deux prismes

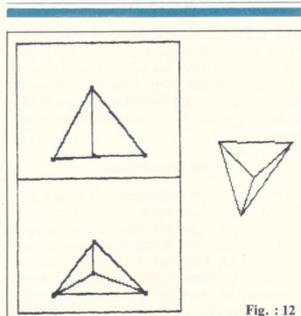


Certaines faces ont été opacifiées : l'ambiguïté est levée.



Jeu logico-géométrique ; fig. 14 : toutes les faces sont transparentes. Il est possible d'opacifier chacune des faces.
Exemple : fig. 14 bis, la face antérieure (A) et

celle d'en bas (B) sont opaques, les autres transparentes. Problème (insoluble ici car il manque des informations) : combien de points sont à l'intérieur ? (Programme J. Petresco)

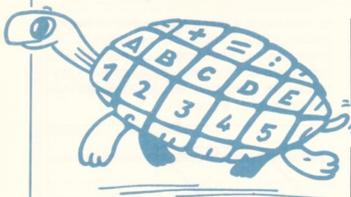


Ici l'objet est construit à partir d'une épreuve située à gauche de l'écran.
Les points sont entrés par leurs deux projections avec possibilité de joindre au précédent.
A chaque étape, il est possible de faire tourner l'objet par des commandes simples (vue de droite)
(dessin fait par un élève de 4^e)

Nous n'avons voulu ici que donner quelques exemples en espérant simplement éveiller l'intérêt pour ce moyen très souple et très puissant qu'est l'illustration de l'enseignement des Mathématiques par ordinateur graphique. Son expérimentation débute à peine, et il est trop tôt pour en cerner tout l'intérêt, s'il en a. Le fait cependant que de plus en plus de personnes et d'équipes travaillent sur ce sujet, en France comme à l'étranger, doit nous inciter à l'optimisme dans ce domaine.

Serge Hocquenghem
CREEM
(CNAM-Paris)

(1) Voir la rubrique « Vos programmes », p. 41



KIJOUAKOUA ?

Il y a vingt ans (ou trente ans même) lorsque les changements dans les conditions de production des objets manufacturés et les nécessités de développement des marchés, joints à la hausse du pouvoir d'achat, ont commencé à rendre envisageable par beaucoup l'achat d'un réfrigérateur, d'une machine à laver, d'une automobile, on ne s'est pas posé beaucoup de problèmes au sujet du sens, de l'intérêt, de l'importance que pouvaient revêtir ces achats. Ils ne faisaient pas problème. Nous en sommes maintenant au magnétoscope, au disque laser et enfin (du moins par le moment) à l'ordinateur domestique. Déjà, dans de nombreux bureaux on archive les disquettes, à côté des cassettes vidéos et des boîtiers classiques en carton destinés aux papiers. On peut rêver (et certains ne s'en font pas faute), de la classe informatique, à distance ou par procuration, de la disquette sur laquelle figurent leçons, problèmes, réponses et corrections, bref d'un système d'enseignement où ne circule plus que cette matière éphémère, éternellement vierge et pure de tout ce qu'elle supporte et qui s'appelle information.

Or cette information ne vaut que par le processus qui l'a produite, que par les essais, les tentatives, les échecs, les reprises qui ont jalonné l'histoire de sa production. A ce titre, l'ordinateur est certainement une machine très rusée. Bien sûr, ce n'est pas la machine elle-même qui est rusée, mais la situation dans laquelle elle va placer celui ou celle qui veut en faire usage. Tous, lecteurs de cette revue, ont plus ou moins entendu parler de LOGO qui reste à ce jour, un des meilleurs exemples d'un emploi intelligent de l'ordinateur à des fins de formation et d'enseignement. Avec le projet déjà ancien de Dynabook, et malgré des échecs et des retards, il représente un prototype bien sympathique d'une machine à apprendre d'un nouveau genre.

Essays de discerner quelques uns des idées essentielles qui pourraient être utiles et significatives pour l'enseignement, en particulier celui des mathématiques : On dit souvent que le savoir, s'acquiert, se développe et se teste dans la résolution de problèmes. Il s'agit là, d'une vieille vérité pédagogique, d'un manteau bien ancien que doivent revêtir toutes les opinions et toutes les doctrines.

Mais, nous avons appris qu'il y a problème et problème, qu'il y a résolution de problème et résolution de problème, que les activités n'ont pas la même signification selon qu'il s'agit de bouger ou de manipuler des pions, d'écrire comment on le fait et d'inventer un système qui nous évite le recours à la pelleteuse pour connaître et calculer sur les grandes collections (les grands nombres). Bref, nous commençons à mettre en évidence et à connaître quelques éléments de la maîtrise didactique des apprentissages, notamment par une meilleure connaissance des conditions de ces apprentissages.

Les mathématiques sont une science où s'élaborent des instruments particuliers de construction et de vérification d'énoncés et de propriétés qui permettent de garantir (autrement dit d'être sûrs qu'ils sont vrais) des résultats (théorèmes). Pour beaucoup, savoir des mathématiques, c'est savoir ces énoncés, ces propriétés et savoir les appliquer. Pour d'autres, le savoir mathématique ne saurait être acquis sans une participation, sans une expérience effective à ce processus de production, d'authentification et de vérification de ces énoncés. Pour faire bref, disons que c'est la qualité de l'expérience mathématique qui importe ! Une orientation qui paraît prometteuse est celle qui consiste à chercher à fabriquer des situations didactiques dans lesquelles les élèves aient à vivre effectivement les étapes nécessaires à la création et à l'identification d'un objet ou d'une théorie mathématique. On commence à disposer d'exemples bien identifiés de ces processus, surtout au niveau de l'école élémentaire ; par exemple : la course à 20.

La Course à vingt
On joue à deux - (deux élèves, les deux moitiés de la classe, etc.)
Chacun énonce un nombre à son tour en ajoutant 1 ou 2 au nombre énoncé par son adversaire. L'un des deux joueurs commence (par 1 ou 2). Celui qui dit 20 a gagné. On peut apprendre à jouer, apprendre des résultats comme "le premier qui a dit 17 a gagné", apprendre à les prouver, etc.

Les contenus traditionnels en mathématiques de l'école secondaire ne se projettent pas toujours aussi bien dans des "jeux" et il faut chercher bien d'autres "mises en scène" qui permettent l'engagement des connaissances et l'identification d'une question.

Qui joue à quoi ?

Programmer un ordinateur, c'est bien pratiquer une forme de jeu, avec un partenaire assez frustré à maints égards (il n'accepte pas qu'on modifie les règles du jeu !) et qui exige qu'on s'adresse à lui par l'intermédiaire d'un code bien spécifique. Ainsi, l'ordinateur impose une médiation entre l'opérateur (le programmeur) et le problème posé. Cette médiation repose sur les opérations nécessaires à l'adaptation et à la transformation des données et de leurs relations selon le code prescrit par la programmation. On ne fait pas forcément des mathématiques, mais on structure son action à travers notamment une distanciation par rapport au résultat (celui qui doit

s'inscrire sur l'écran ou sur l'imprimante). Quelles sont les opérations qu'il faut faire exécuter pour obtenir la production de ce résultat ? C'est la succession de ces opérations et sa structuration qui vont constituer le programme. Elles doivent être définies en fonction de choix de représentations d'entrées et de sorties, avec les modifications qu'apportent les contraintes du traitement algorithmique aux écritures algébriques usuelles.

Somme, Boucle et Indices

La moyenne de n nombres a_i ($i=1, 2, \dots, n$) est définie par l'écriture

$$(1) \quad n \times M = \sum_{i=1}^n a_i = S;$$

c'est une écriture "globalisante". Pour faire exécuter ce calcul par une machine séquentielle, il faut expliciter la succession des opérations qui fabriquent M autrement dit calculer les n sommes partielles (SP_i ($i=1, \dots, n$)) selon le schéma

$$\begin{array}{l} \text{faire} \\ \text{n fois} \end{array} \quad SP_i = SP_{i-1} + a_i$$

Traduit par

$$(2) \quad SP \leftarrow SP + a[i] \\ \text{pour } i \text{ variant de } 1 \text{ à } n$$

Il n'y a pas, dans ces opérations, qu'un simple jeu d'écritures. Pour les effectuer il faut faire connaître (apprendre à connaître) les rôles respectifs de (1) et de (2).

Dans le cas de notre encadré, la suite de nombres (a_i) doit être prise comme un tableau, ses éléments doivent être soigneusement identifiés, rentrer dans la définition d'opérations écrites sous une forme évaluable et calculable. Il y a un changement de "point de vue".

Que l'on travaille avec un système comme LOGO en Géométrie de Tortue ou avec des langages standards assortis des problèmes et exercices déjà classiques, il faut être en mesure de conduire cette analyse pour comprendre le genre d'opérations (de pensée) que l'élève doit effectuer, donc les difficultés qu'il peut rencontrer et les expériences d'utilisation et de manipulation du savoir qui ont pu être les siennes. Au "penser sur sa propre pensée" qui a été le slogan de certains promoteurs de LOGO, on peut substituer, pour être plus réaliste le "penser au sujet de l'action elle-même", agir pour produire un système qui doit, lui-même, effectuer des actions.

Certes, on peut penser légitimement, que tout cela est encore bien loin des mathématiques. Il faudra bien d'autres opérations intellectuelles pour en produire, mais nous pouvons penser que cette mise en situation de l'élève (du sujet connaissant) par rapport aux opérations qu'il effectue est une étape importante et décisive dans le processus de mathématisation. C'est ce processus même qui nous intéresse pour l'enseignement des mathématiques. Il faudra certainement passer par la conception d'autres ensembles "mathématiques" comme la Géométrie de Tortue, mais nous sommes bien convaincus que c'est une voie dont l'exploration et l'étude est de première nécessité pour que l'usage scolaire de l'informatique et de l'ordinateur soit fécond pour l'enseignement des mathématiques.

**André Rouchier
IREM d'Orléans
(Université — La Source)**

La MATHÉMATIQUE...

SCIENCE EXPÉRIMENTALE ?

La possibilité de faire rapidement et mécaniquement des calculs et des dessins rend à la mathématique son caractère expérimental. On peut alors faire des hypothèses, et des expériences pour les confirmer ou les infirmer ; et on débouche, soit sur la recherche de démonstrations, soit sur la justification de définitions ; les trois encadrés ci-dessous en sont des exemples manifestes...

André DELEDICQ

COMME VOUS LE VOYEZ,
EXPERIMENTALEMENT "2"
EST PLUS PETIT QUE "3"...



Développements

Quels sont les liens qui tissent entre 27 et 37 une aussi curieuse rencontre ?

$$1/27 = 0,037037037037\dots$$

$$1/37 = 0,027027027027\dots$$

D'où vient la régularité, puis la rupture dans les deux développements ci-dessous ?

$$1/81 = 0,01234567901234\dots$$

$$1/98 = 0,01020408163263\dots$$

Dérivée

De simples calculs au voisinage de 4 font apparaître, sans peine, la différentielle de $x \rightarrow \sqrt{x}$

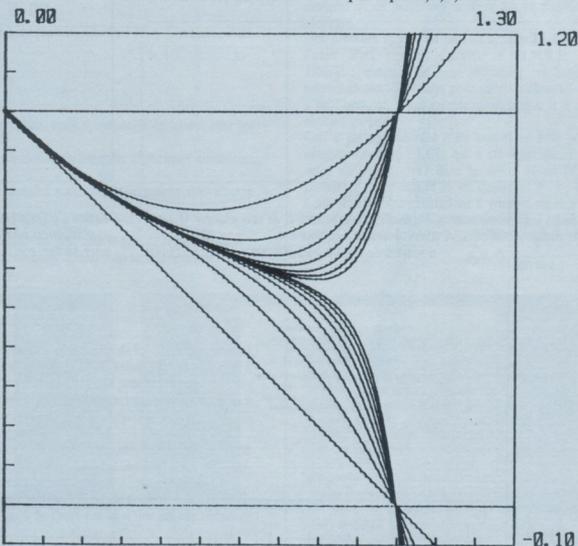
x	\sqrt{x}
4	2
4,1	2,024845
4,01	2,002498
4,001	2,000249
4,0001	2,000025

(Lorsqu'on divise l'accroissement de x par 10, il en est de même de l'accroissement de la fonction. Le coefficient de ce lien linéaire semble approcher 0,25).

Le bon vieux « plumeau de Taylor »

$$y = 1 - x + x^2 - x^3 + \dots + x^p$$

pour $p = 1, 2, 3, \dots, 15$.



Parmi les thèmes mathématiques pouvant être illustrés avec profit, les suites figurent en bonne place. Une brochure de l'IREM de PARIS 7 (Place Jussieu 75005 - 10 F) est ainsi entièrement consacrée aux remarquables suites définies par

$$u_{n+1} = f(u_n)$$

où $f(x) = mx(1-x)$, m étant compris entre 0 et 1.

Cet exemple sera développé dans la rubrique « vos programmes » du prochain numéro de E & I.

Cependant, un enseignant ne peut exploiter de telles expériences que s'il a auparavant fait une étude mathématique sérieuse et s'il s'est assuré que le matériel dont il dispose donne une représentation fidèle des phénomènes à observer. On peut citer de nombreux exemples où des phénomènes aberrants sont observés à cause des limites de la machine : il

suffit de calculer $\frac{x+y-x}{y}$

de calculer les sommes partielles de la série harmonique, ou...

de calculer les termes successifs de la suite

$u_n = (1 + 1/n)^n$, $n > 0$, comme le montre la mésaventure suivante.

La triste histoire d'Arthur Bécassin

André Meyer
Colette Steyaert
Professeurs au lycée
Victor Duruy — Paris

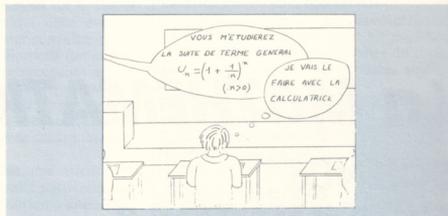


Fig. 1

Le professeur : « Faites attention aux risques d'erreurs si vous utilisez une calculatrice. »

Arthur, déçu, se dit qu'il pourra plutôt résoudre son problème sur l'ordinateur du lycée. N'aimant pas trop les nombres, il représente sa suite sur la console graphique haute définition et, pour que ce soit plus joli, il utilise une représentation « polaire ». $\theta = \frac{n}{50}$, $\rho = u_n$ $1 \leq n \leq 1000$



Fig. 3

Arthur : « Oh ciel ! j'ai toujours rêvé de faire de l'astronomie ! mais ces calculs de puissance me paraissent fantaisistes ! »

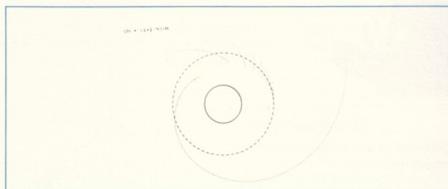


Fig. 4

Arthur : « Qu'est-ce que c'est que ces machins en spirales ? Les mathématiques sont vraiment pluridisciplinaires. Je vais montrer mes dessins à mon ami Stéphane Mathoux ».



Fig. 6

(Arthur a travaillé sur la console graphique reliée au T 1600 Télémechanique de son lycée).

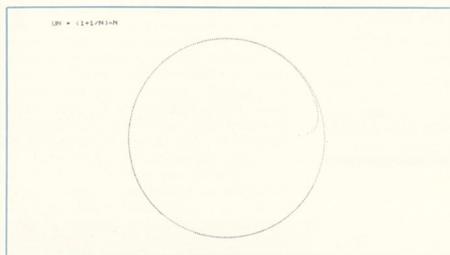


Fig. 2

Arthur : « D'après ce que je vois elle semble converger vers 2,72 mais je vais m'en assurer en regardant plus loin. »

$\theta = \frac{n}{50}$, $\rho = u_n$ n variant de 25 000 à 15 000 000 par pas de 25 000.



Fig. 3 bis

Arthur : « Et maintenant voilà qu'elle converge vers 1. Je vais changer θ pour faire bouger les étoiles, et tracer le cercle de rayon e . »

$\theta = \frac{n}{1\ 250\ 000}$, $\rho = u_n$ n variant de 25 000 à 15 000 000 par pas de 25 000



Fig. 5

IL ÉTAIT UNE FOIS... UN DIDACTICIEL.

Cet article, extrait du compte rendu d'une recherche menée dans le cadre de l'INRP, conte l'histoire de la mise au point de l'expérimentation d'un logiciel de maths : DUC2.

UN CONSTAT DIDACTIQUE

Nous avons constaté que nombre d'élèves de 3^e (voire de 2^e) sont incapables de résoudre correctement des équations dans \mathbb{R} , et sont complètement perdus dès que celles-ci ne sont pas du type le plus simple (1^{er} degré, du genre $2x - 3 = 0$). Corrélativement, les techniques de factorisation sont très mal maîtrisées ; la notion de solution d'une équation n'est pas toujours bien comprise.

Voici quelques exemples de réponses fausses trouvées en 3^e, qui sont assez typiques :

$$\begin{aligned} 1. \quad 3x^2 - 6x = 0 &\Leftrightarrow 3x^2 = 6x \\ &\Leftrightarrow 3x = 6 \\ &\Leftrightarrow x = 2 \end{aligned}$$

Dans ce cas l'élève a abusivement simplifié par x et ne trouve plus la solution évidente $x = 0$.

$$\begin{aligned} 2. \quad x^2 + 4 = 0 &\text{ est vrai si } x = -2 \\ (\text{Il y a ici confusion entre } -(2)^2 \text{ et } (-2)^2) \end{aligned}$$

$$3. \quad x^2 - 50x + 25 = 0 \Rightarrow x = 25 \text{ car } 0^2 - 50 \cdot 0 + 25 = 25$$

Dans ce cas l'élève n'a pas compris ce qu'on lui demandait et a simplement substitué 0 à x . C'est une erreur récurrente chez certains élèves bloqués en mathématiques.

$$\begin{aligned} 4. \quad f(x) &= 3x^2 - 12 \\ f(2) &= 6^2 - 12 \neq 0 \text{ donc } 2 \text{ n'est pas solution de } 3x^2 - 12 = 0 \\ \text{Il y a confusion entre } 3(2^2) &= 12 \text{ et } (3 \cdot 2)^2 = 36. \end{aligned}$$

Ce genre d'erreurs est très fréquent, surtout parmi les élèves faibles, mais se rencontre également chez les élèves considérés comme "bons".

Il révèle la grande difficulté pour les élèves actuels de la manipulation de formules algébriques abstraites. Les points délicats rencontrés sont de plusieurs types :

- Incompréhension de la notion de fonction ; les élèves ont du mal à substituer correctement une valeur à une variable.
- Manque de maîtrise de calcul sur les puissances : (confusion x^2 , $x \times 2$..).
- Grande difficulté à utiliser les fameuses "identités remarquables" surtout dans le sens de la factorisation ; ainsi $x^2 + 6x + 9$ n'est pas toujours reconnu comme $(x+3)^2$ et beaucoup d'élèves ont tendance à développer $(a+b)^2 = a^2 + b^2$ (tentation linéaire).
- Confusion factorisation/développement.

UN SUJET QUI POSE PROBLÈME...

D'une façon générale, il existe une grande méfiance des élèves envers les exercices de factorisation qu'ils ne trouvent pas comment aborder. Et en effet, il n'y a pas en la matière de règles automatiques sécurisantes à appliquer les yeux fermés et il faut déjà avoir une certaine intelligence du sujet pour y réussir.

Ajoutons en outre que le problème de la factorisation polynomiale est en lui-même très compliqué. Ainsi il n'existe pas de méthode générale pour factoriser des polynômes sur \mathbb{R} de degré supérieur ou égal à 5 ni de méthode simple dans le cas où leur degré est supérieur ou égal à 3.

En pratique, les exercices de factorisation effectivement rencontrés dans l'enseignement secondaire sont assez artificiels et construits pour donner des solutions "simples" (racines souvent entières) grâce à l'application plus ou moins immédiate de quelques règles de réécriture.

Dans ce domaine l'élève travaille plutôt par imitation d'exemples déjà traités par le professeur qu'en appliquant des méthodes générales parfaitement maîtrisées ; c'est qu'il ne s'agit pas ici d'une activité déductive mais bien inductive et c'est ce qui en fait la difficulté.

L'élève doit donc apprendre à reconnaître la forme de l'exercice, et à chercher les règles qui s'appliquent pour produire un facteur commun.

FAIRE LA SYNTHÈSE

Il est cependant essentiel dans le second cycle de bien maîtriser les règles opératoires du calcul littéral et de savoir factoriser. En effet, une technique élémentaire et importante de résolution d'une équation algébrique est de la mettre sous la forme d'un produit de facteurs égal à 0.

Il était nécessaire de mettre en relation ces deux domaines : factorisation et résolution d'équations ; c'est pourquoi nous avons mis sur pied un logiciel d'entraînement à la résolution d'équations de degré 2 qui fait appel aux techniques de factorisation en facteurs du 1^{er} degré et favorise la pratique du calcul algébrique et numérique. L'élève a aussi la possibilité de résoudre directement (recherche de solutions évidentes). Dans les exercices proposés, on admet implicitement qu'un polynôme de degré 2 a au plus deux racines.

Une première version du produit appelée DUC a fait l'objet d'un rapport d'activité à l'I.N.R.P. en 1980/81 ; conçu pour mini-ordinateur, ce logiciel nécessitait une adaptation pour micro-ordinateur qui a été l'occasion d'une refonte informatique et d'une révision globale.

Ceci a permis l'écriture d'un logiciel en LSE pour micro-ordinateur, DUC1, qui a été expérimenté au cours de l'hiver 81/82 dans les lycées Henri IV et Bergson à Paris, auprès de six classes de 3^e, 2^e, 1^{re}. Cette série d'expérimentations a conduit au logiciel actuel, DUC2, validé auprès d'élèves dans les deux mêmes lycées au printemps 82 et dans les mêmes classes. (Voir encadré).

LE LOGICIEL "DUC2"

Ses caractéristiques essentielles sont les suivantes :
— lien souligné entre solution de l'équation $f(x) = 0$ et valeurs prises par la fonction f .

— mise en œuvre de techniques algébriques (simplifications factorisations, etc.) qui permettent de remplacer une équation $f(x) = 0$ par une autre équivalente et plus simple.

— manipulation des identités remarquables comme règles de réécriture dans le sens de la factorisation,
— dans certains cas plusieurs méthodes de résolution sont possibles,

— système de notation souple et stimulant qui s'adapte aux difficultés de l'élève.

Ce logiciel est suffisamment souple pour servir dans les classes de 3^e, 2^e, 1^{re}. Les élèves de 1^{re} qui connaissent le discriminant peuvent en faire usage dans certains types d'exercices.

Pour obtenir un produit stimulant, nous définissons deux notes qui sont affichées d'après chaque conclusion d'exercice : l'une locale à l'exercice qui vient d'être résolu, et l'autre globale à la séance.

Il est élaboré au cours de chaque séance un "profil" de l'élève basé sur ses performances passées dans différents types d'exercices proposés et qui nécessitent chacun des savoir-faire particuliers.

Nous avons défini 9 types d'exercices selon les savoirs mis en jeu et les méthodes de résolution les plus typiques qu'on peut leur appliquer, à savoir :

Type 1 : $a(x^2 + bx) = 0$ ($b \neq 0$)

Type 2 : $a(x^2 + c) = 0$ ($c \neq 0$)

Type 3 : $a(x^2 - c) = 0$ ($c \neq 0$)

Type 4 : $ax^2 = 0$

Type 5 : $a(x^2 + 2bx + b^2) = 0$ ($1 \leq |b| \leq 9$)

Type 6 : $a(x^2 + 2bx + b^2) = 0$ ($10 \leq |b| \leq 14$)

Type 7 : $a(x-1)(x-x') = 0$ ou $a(x+1)(x-x') = 0$ ($|x'| > 2$)

Type 8 : $a(x-x')(x-x'') = 0$ sans factorisation ni solution évidente.

Type 9 : $ax^2 + bx + c$ avec $\Delta < 0$ et $b \neq 0$

N.B. : a n'est jamais nul. Tous les réels qui interviennent dans les énoncés et les calculs sont des entiers relatifs. Les types 7 et 8 sont donnés dévolés.

Les élèves peuvent résoudre des exercices de 3 niveaux différents :

pour les niveaux 1 et 2, les énoncés sont tirés parmi les types 1 à 5 ; le niveau 3 fait appel à l'ensemble des types en privilégiant les types 7, 8, 9 dans la proportion de 2 sur 3.

Le niveau s'adapte à l'évolution des performances de l'élève et à celui de ses connaissances ; le type d'un nouvel exercice est choisi en fonction du type du précédent et des difficultés rencontrées.

UNE MISE AU POINT PAR ÉTAPE

Du point de vue pédagogique, on a tenté de réduire le plus possible les difficultés constatées dans les logiciels précédents qui étaient liées à des problèmes de langage et de notations ; en voici 3 exemples :

1. Les vocables "racine" et "polynôme" étaient mal compris des élèves, de plus ils ne sont plus guère utilisés dans le 1^{er} cycle : ils ont été remplacés respectivement par "solution" et "fonction".

2. La formulation de la question initiale était assez mal comprise et génératrice d'efforts souvent considérables pour y répondre par l'affirmative. Donc la phrase "le trinôme T(x) est... a-t-il une racine évidente ?" a été remplacée par celle-ci : "Voyez-vous sans calcul un nombre qui soit solution de l'équation T(x) = 0 ?".

3. Les notations pour les solutions : X0 et X1 ont été remplacées par X' et X'', car beaucoup d'élèves voyaient une multiplication X > 0 ou bien encore une égalité X = 0 dans l'écriture X0.

D'autre part dans la mesure où la structure assez rigide du logiciel à adapter le permettait, on a cherché à assouplir le plus possible et à réduire les étapes de résolution de façon à se rapprocher des modes naturels de raisonnement.

Par exemple la résolution de l'équation $x^2 + 22x + 121 = 0$ peut se faire par factorisation si l'élève voit une de façon évidente ou bien par le calcul de Δ .

L'effort a également porté sur les entrées-clavier et les sorties sur écran :

Concernant les entrées : on a cherché à limiter les effets pervers des erreurs de frappe au clavier en filtrant les réponses données par l'élève, qui sont toutes considérées comme des chaînes de caractères. En cas de non conformité du résultat filtré avec le type de réponse attendu, la question est réposée.

En l'absence des procédures binaires, on n'a malheureusement pas pu neutraliser certaines touches, telles celle d'effacement écran ou celle d'interruption.

Concernant les sorties : on a beaucoup prêté attention à la mise en page, à la mise en évidence des parties les plus importantes, et à la structuration dynamique du déroulement, en décomposant en pages-cran les différentes phases de la résolution.

STRUCTURE ET UTILISATION DU LOGICIEL

Du point de vue informatique il se compose d'un module unique faisant appel à 28 procédures internes dont 2 récursives. En fin de séance un fichier est mis à jour que l'on peut dépouiller grâce à l'utilitaire : DEPOU

Les procédures sont de 4 types :

1. Procédures qui assurent le traitement d'une étape de résolution de l'exercice.
2. Procédures de calcul.
3. Utilitaires de mise en page.
4. procédure de gestion du fichier résultats.

Lorsqu'une séance d'utilisation de DUC2 est achevée, on peut garder des indications sur le travail des élèves. C'est le rôle du fichier # RES qui est géré sur la disquette de chaque élève (ou binôme) et dans lequel figurent la date, le niveau de la classe (3, 2 ou 1) ainsi que la trace du travail de l'élève : notes et façons de résoudre.

Il reste alors à dépouiller les différents fichiers # RES et à les centraliser sur une disquette professeur : c'est l'objectif du logiciel DEPOU qui crée sur la disquette du professeur un fichier # UNI constitué de la concaténation de chacun des fichiers # RES et imprime en série les indications relatives à chaque élève ; parallèlement, il remet à zéro le fichier # RES. Lors de son exécution, il faut bien entendu faire lire successivement les disquettes élèves dans le lecteur n°1. Le fichier # UNI regroupe donc les données relatives au passage d'une classe. On peut éditer ultérieurement son contenu grâce au programme utilitaire DEPE2.

**Georges-Louis Barot
Jean-Louis Froton
Professeurs de Mathématiques**

UTILISATION EN CLASSE

PRIS SUR LE VIF

Stéphanie et Pascal sont deux élèves de niveau moyen/faible. Stéphanie a beaucoup progressé en cours d'année et passe en seconde. Pascal, elle, va redoubler. Toutes deux ont encore des difficultés avec les calculs, surtout lorsqu'interviennent des inconnues. Elles sont un peu familiarisées avec le logiciel qu'elles ont déjà fait tourner à deux reprises. Aujourd'hui, le professeur

leur a proposé de l'utiliser au niveau 2^e.

Comme d'habitude, elles commencent par résoudre sur papier, avant de répondre à la console, en discutant passivement sur les réponses à donner. Elles notent aussi leurs scores locaux.

Elles se trouvent devant un exercice de type 3 :

$$3x^2 - 12 = 0, \text{ et essaient des nombres pour voir s'ils sont solutions :}$$

— Ça marche pas pour 2, parce que $6^2 = 36$ (confusion entre $(3 \times 2)^2$ et 3×2). Elles ne sont pas très sûres et essaient 2 quand même. En revanche, elles ne voient pas -2, et ne l'aperçoivent qu'après simplification et factorisation.

Par la suite, elles répondent toujours "non" à la première question, car elles ont remarqué que la moyenne n'en souffrait pas. Puis elles se trouvent devant :

$$-5x^2 - 80 = 0 \text{ (type 2), et après avoir répondu "non" à la}$$

première question se trouvent devant :

$$-5x^2 - 80 \text{ peut-il s'annuler ?}$$

— non ? oui ? ; elles essaient 80/5 = 16 :

— oui !

— ça peut s'annuler pour 16

— non, pour 8, c'est x' ...

— ah non, 4 !

et sont bien étonnées quand on leur indique une erreur.

Devant x' - 9x, elles ne trouvent pas de racine évidente, se trouvent devant le tableau de valeurs, et immédiatement voient que pour 0 cela fait 0. A la remarque du professeur qu'elles auraient pu voir que 0 était solution, elles répondent :

— de toute façon, on aime bien voir le tableau !

puis à la fin

— M'sieur, on 20 et 12,5 ; on en fait un dernier, faut qu'on ait 20 ! (prenant son papier) : j'ai plus de place, je vais écrire là-dessus...

Silence... "on peut mettre 3 en facteur !"

— Alex ! (leur voisine a une calculatrice) Tape 243/3 (!)

Merde ! - 81, attends...

Leur attitude à la console (effervescence et travail soutenu) est très différente de celle qu'elles ont en classe. Elles restent actives au-delà de la fin du cours.

Véronique (3^e) est une élève sérieuse, plus mûre que la majorité de ses camarades, mais qui a des problèmes avec le calcul littéral. Elle est seule à une console, et travaille avec attention. Par contraste avec les autres, elle lit les textes affichés sur écran, réfléchit, répond, se trompe, refait ses calculs, a une nouvelle erreur, reprend ses papiers, hôte la tête, répond à nouveau... Elle ne se décourage pas et ne demande pas d'aide. Son autonomie est à tout faire remarquable. Elle s'arrêtera que lorsqu'elle sera satisfaite de sa moyenne.

Vincent et Alexandre (1^{er} C) sont deux élèves relativement moyens. Ils s'appliquent énormément et cherchent systématiquement si -1, 0, ou 1 sont solutions évidentes. Ils ont de bonnes notes.

Vincent (à Jean-Louis et Pascal)

— Hé ! on vient d'avoir 20 ; et vous ?

— Nous aussi.

— Nous on a 20 et 13,5 ; et vous ?

— 20 et 8,5 ...

Gester d'intense jubilation de Vincent...

Ils terminent après plusieurs exercices notés 20 et une moyenne de 17.

Jean-Louis et Pascal (1^{er} C) sont deux bons élèves, passant sans problème en TC. C'est la première fois qu'ils viennent exécuter un logiciel d'EAO cette année.

Ils commencent de façon désolée, sans bien réfléchir : ils ont l'impression d'être "au-dessus de ça". Ils accumulent erreur sur erreur, et se trouvent devant une note très faible : 5/20.

Piqués au vif, ils saisissent alors papier et crayon et commencent à écrire leurs solutions. Ils commentent des erreurs de calcul (les coefficients des équations proposées sont parfois assez grands) et sont tout à fait indignés de leur deuxième note : 7,5/20 ; "l'enfermire !" (en s'adressant au micro-ordinateur). Ils s'accrochent alors et refusent de terminer avant que leur moyenne ne soit remontée au-dessus de 10. Il faut noter que, quoique bons élèves, ils ont du mal à voir 0 comme racine évidente par exemple dans $3x^2 - 6x = 0$.

INFORMATIQUE ET CONJECTURES et RÉSULTATS MATHÉMATIQUES

L'informatique, tout comme la Physique, la Biologie ou les Sciences sociales, est liée au monde réel. A ce titre elle est à la fois consommatrice de mathématiques et productrice de problèmes : la représentation, la compression éventuelle et la transmission des données posent des problèmes difficiles de codage qui font appel à la théorie des groupes ou à la théorie des corps finis ; la conception des logiciels nécessite la mise au point de méthodes numériques ou non-numériques performantes qui relèvent de ce qu'on appelle aujourd'hui la complexité algorithmique ; la gestion des systèmes en temps partagé utilise la théorie des files d'attente et l'on pourrait sans difficulté étendre la liste des outils mathématiques utilisés lors de la conception des systèmes : logique, théorie des graphes, algèbre, etc. A l'autre extrémité de la chaîne informatique, celle de l'utilisateur, les problèmes posés sont encore plus diversifiés et dépendent uniquement des applications traitées. Il est donc clair qu'il y a, à ces différents niveaux, une forte demande de mathématiques (1).

Réciproquement, l'Informatique peut aider les mathématiciens, et de plusieurs façons, même si l'usage des ordinateurs n'est pas encore très répandu dans ce milieu. Mais avant de voir comment sur quelques exemples, précisons un certain nombre de points.

En premier lieu, les mathématiciens étudient des objets de natures très différentes qui vont des nombres aux espaces fonctionnels en passant par les groupes, les graphes, les variétés différentiables, les matrices, les expressions, etc. ! Il se superpose donc aux problèmes à résoudre, proprement mathématiques, d'une part l'apprentissage d'un langage de programmation, d'autre part la recherche d'une représentation adéquate des objets à étudier, ce qui explique sans doute que la plupart des mathématiciens qui utilisent actuellement l'informatique travaillent sur des objets discrets et non sur des objets continus.

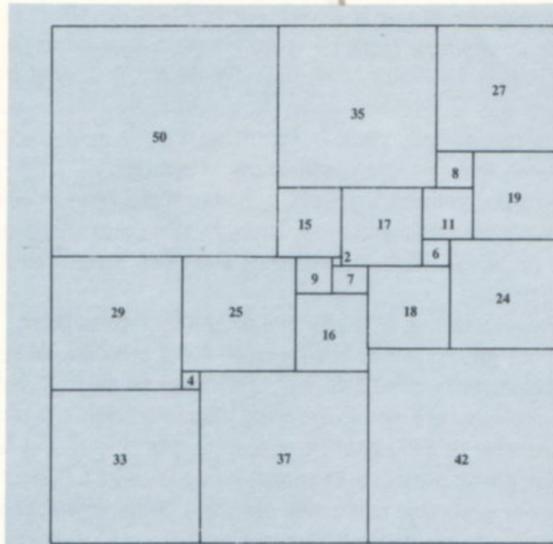
En second lieu, on remarquera que la représentation interne des données impose de travailler, non seulement sur des nombres finis, mais encore rationnels décimaux :

$1/3$ n'est pas égal à $0,3333333$, $\sqrt{2}$ n'est pas égal à $1,4142136$ et e n'est pas égal à $2,7182818$! Si le premier peut être représenté sous la forme $(1,3)$, modulo la programmation convenable des opérations élémentaires, les deux autres nécessitent l'utilisation de langages permettant le calcul formel, lesquels commencent seulement à être largement diffusés, notamment sur certains micro-ordinateurs.

Cela dit, à quoi peut servir un ordinateur en mathématiques ? L'observation de ces vingt ou trente dernières années, si elle n'autorise pas la prospective, permet au moins de faire un tour d'horizon sommaire. On rencontre ainsi :

- 1) La constitution de tables diverses (de nombres, de polynômes, etc.).
- 2) L'assistance à la construction d'objets.
- 3) La démonstration de théorèmes.
- 4) L'établissement de conjectures.
- 5) La réfutation de conjectures.
- 6) La découverte de nouveaux phénomènes.

Le cas 1 est sans aucun doute le plus répandu. On notera comme exemple la recherche des nombres de Mersenne premiers, c'est-à-dire de la forme $2^n - 1$. On en connaît actuellement 26 et le record actuel est détenu par Slovincy avec $n = 44497$, ce qui correspond à un nombre de plus de 13 000 chiffres. Il va de soi qu'il est hors de question, pour des nombres de cette taille, de chercher des diviseurs éventuels : la méthode utilisée est due à Lucas, et date de presque un siècle. Elle est assez simple pour être décrite ici (mais sans démonstration !)(voir encadré col. 3).



Un autre exemple est donné par la constitution de tables de solutions, en nombre rationnels, des équations du type

NOMBRES DE MERSENNE

Soit $M = 2^n - 1$ le nombre étudié. On calcule la suite L_k définie par $L_0 = 4$, $L_{k+1} = (L_k^2 - 2) \pmod{M}$. Le théorème de Lucas dit que M est premier si et seulement si $L_{n-2} = 0$. La vérification de la primalité du nombre de Slovincy demande donc seulement 44495 élévations au carré (sur des nombres assez grands, il est vrai).

Exemple : $n = 5$ donne $M = 31$. On a successivement $L_0 = 4$, $L_1 = 14$, $L_2 = 194 = 8 \pmod{31}$, et $L_3 = 62 = 0 \pmod{31}$ donc 31 est premier, ce qui n'étonnera personne.

Notons que la recherche rapide de très grands nombres premiers, ou la factorisation de très grands nombres, connaît actuellement un regain d'intérêt depuis leur utilisation dans les systèmes de codages à clés publiques.

$Y^2 = X^3 + aX + b$, équations sur lesquelles on possède un assez grand nombre de résultats théoriques importants, mais dont on ignore non seulement une méthode générale de résolution (si elle existe !), mais même dans quelles conditions il existe des solutions.(2)

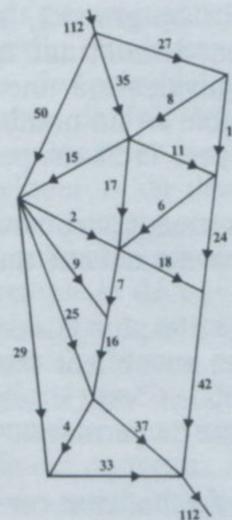
En ce qui concerne le cas 2, on peut évoquer l'usage des ordinateurs dans la détermination, récemment achevée, des groupes simples sporadiques. Ou encore, pour citer un problème d'énoncé accessible à tous, la découverte, également récente, du plus petit carré parfaitement parfait par Duijvestijn.

CARRÉS PARFAITEMENT PARFAITS

Il existe des carrés qui sont somme de deux carrés, par exemple $5^2 = 3^2 + 4^2$, mais il s'agit seulement d'une égalité numérique : un carré de côté 5 ne peut pas être pavé, sans recouvrement ni lacunes, à l'aide de deux carrés de côtés respectifs 3 et 4. Par contre, un carré de côté n peut, bien sûr, être pavé à l'aide de n^2 carrés de côté 1. Le problème a donc été posé, il y a près de cinquante ans, de construire des carrés parfaitement parfaits, c'est-à-dire que l'on peut paver à l'aide de carrés tous distincts. Le premier exemple, donné par Brooks, Smith, Stone et Tutte en 1938, était d'ordre 69. Celui de Duijvestijn, de côté 112 et d'ordre 21, est le plus petit possible en ce qui concerne la taille. Par ailleurs, Duijvestijn a démontré qu'il n'existe aucun carré parfaitement parfait d'ordre inférieur à 21.

Le lecteur trouvera le principe d'exploration utilisé ainsi que ses liens avec l'étude de circuits électriques dont le graphe est planaire, dans :

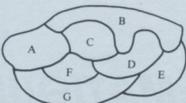
R.F. Churchhouse et J.C. Herz, *Computer in mathematical research* North-Holland, Amsterdam, 1968.



(2) Serge Lang, *Une activité vivante : faire des mathématiques*. Revue du Palais de la Découverte. A paraître en janvier 1983.

LE THÉORÈME
DES QUATRE COULEURS.

Est-il possible de colorier n'importe quelle carte plane dont les pays sont d'un seul tenant, à l'aide de quatre couleurs de façon que deux pays ayant une frontière commune soient de couleurs différentes ?
Posé à A. De Morgan en 1852, ce problème a été résolu en 1976 par Appel, Haken et Koch au terme de 1 200 heures de calcul mené sur un IBM 370-168 (3). Ici, l'ensemble E est celui de toutes les cartes possibles, évidemment en nombre infini. Le passage de l'infini au fini repose sur une idée de Kempe datant de 1879. Nous la formulerons en termes de graphes, ce qui permet déjà de négliger la forme exacte des frontières : il suffit de représenter chaque pays par un sommet du graphe et de relier deux sommets par une arête si les pays correspondants ont une frontière commune. On dit dans ce cas que les deux sommets sont adjacents et l'on notera que le graphe obtenu est planaire c'est-à-dire que les arêtes ne se coupent pas en dehors des sommets.



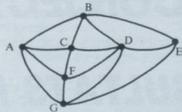
En ce qui concerne le cas 3, on est encore loin de pouvoir démontrer des théorèmes intéressants de façon formelle, même si les progrès en intelligence artificielle ont été considérables ces dernières années. Néanmoins les ordinateurs ont permis d'obtenir un certain nombre de résultats dans le cas suivant : Soit à démontrer un théorème de la forme :

Théorème : Pour tout élément x d'un ensemble E la propriété $P(x)$ est vraie.

Si l'ensemble E est fini et si, pour chaque élément de E , on peut vérifier la propriété P en un nombre fini d'opérations, alors on peut confier ce travail à une machine. Sa capacité à « démontrer » le théorème dépendra uniquement de la virtuosité du programmeur, de la taille mémoire et du temps de calcul. On a quelquefois contesté ce type de « démonstration », considéré par certains comme « non-mathématique ». Une telle réaction n'est pas fondée : ce type de démonstration est bien antérieur au calcul automatique. Ce qui pourrait être discuté, c'est la validité d'une telle preuve lorsque les temps de calcul sont tels que seul un autre ordinateur peut effectuer la vérification. En fait, même cet argument est faible car dans le cas des groupes finis simples, les spécialistes s'accordent pour dire que la classification est achevée tout en reconnaissant que la rédaction effective de la preuve, faisant la synthèse de tous les articles publiés sur le sujet, dépasserait probablement 10 000 pages et ne serait peut-être pas exempte d'erreurs ! (indépendamment du fait que l'existence de certains de ces groupes a été vérifiée à l'aide d'ordinateurs !)

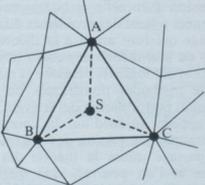
Lorsque les conditions, sur E et sur P , ne sont pas vérifiées, (ou conduisent à des temps de calcul trop grands), on s'efforce d'y remédier en démontrant que la propriété P est équivalente à une autre propriété P' vérifiable en un nombre fini (raisonnable) d'étapes. Trois cas peuvent alors se produire :

- On arrive également à trouver un ensemble E' fini et on est ramené au cas précédent.
- On est obligé d'explorer les éléments de E un par un, et on obtient seulement des résultats partiels. C'est le cas pour le « grand théorème de Fermat ». (Voir encadré p. 33).
- On a un peu de chance et on constate en cours de calcul qu'il est inutile d'explorer E en entier. C'est ce qui s'est passé pour la démonstration du « Théorème des quatre couleurs » par Appel, Haken et Koch.



On montre ensuite qu'il suffit de savoir colorier les graphes triangulés, ce qui réduit l'ensemble E à étudier (mais il est toujours infini !). On cherche alors un contre-exemple : s'il existe des graphes qui nécessitent cinq couleurs, alors il en existe un dont le nombre de sommets N est le plus petit possible, ce qui signifie qu'un graphe triangulé quelconque dont le nombre de sommets est inférieur à N peut être colorié avec quatre couleurs. Cela dit, quelle est la démarche de Kempe ? Il commence par montrer que tout graphe triangulé possède un sommet de degré 3, 4 ou 5. On dit que ces configurations forment un ensemble inévitable (cet ensemble est fini !). Puis il montre qu'un graphe qui contient l'une de ces configurations peut toujours être colorié à l'aide de quatre couleurs, on dit alors que la configuration est réductible. Il ne peut donc exister de contre-exemple et le théorème est démontré. Voyons ce qui se passe pour un sommet de degré 3 ou 4.

Supposons que le contre-exemple contienne un sommet S de degré 3. Retirons-le ainsi que les trois arêtes qui y aboutissent : on obtient un nouveau graphe, qui possède $N-1$ sommets et que l'on peut donc colorier avec 4 couleurs. En particulier, les sommets A, B, C seront, par exemple, Rouge, Vert et Jaune. Ré-introduisons le sommet S . Il nous reste une couleur. On peint S en bleu et on a gagné ; le contre-exemple ne peut donc pas posséder de sommet de degré 3.



Est-il possible qu'il possède un sommet S de degré 4 ? La réponse est non mais c'est un peu plus délicat à montrer. Comme précédemment, on supprime S et les arêtes SA, SB, SC, SD puis on colore en 4 couleurs le nouveau graphe obtenu.

Si les sommets A, B, C, D n'utilisent que 3 couleurs on attribue la quatrième à S et c'est fini. Dans le cas contraire — par exemple A vert, B rouge, C bleu et D jaune —, on va essayer de modifier le coloriage de façon à n'utiliser que 3 couleurs. Si les voisins immédiats de B (rouge) sont seulement verts et bleus, on peut changer B en jaune et on a gagné. Si B a un voisin jaune, on pourra échanger ces deux couleurs sauf si ce jaune a un voisin rouge distinct de B . On est donc amené à considérer, à partir de B , la chaîne des arêtes qui joignent alternativement les sommets rouges et jaunes. Si cette chaîne n'atteint pas le sommet D c'est terminé : il suffit d'échanger le rouge et le jaune. Si la chaîne atteint D c'est inutile car B va devenir jaune pendant que D deviendra rouge. On regarde alors la chaîne vert-bleu issue de A .

Elle ne peut pas atteindre C car elle devrait croiser la chaîne jaune-rouge, ce qui est impossible puisque le graphe est planaire et que la chaîne $J-R$ ne contient évidemment pas de sommets bleus ou verts. On peut donc échanger le vert et le bleu dans la chaîne issue de A . A devient bleu sans que C devienne vert. On n'utilise plus alors que 3 couleurs et on attribue la quatrième à S : le graphe initial n'était donc pas un contre-exemple possible.

On a donc obtenu le résultat suivant : S'il existe un contre-exemple, il ne possède aucun sommet de degré 3 ou 4. Par un raisonnement analogue, Kempe s'est démontré également qu'un sommet de degré 5 est impossible, ce qui résolvait le problème. Malheureusement, Heawood montra en 1896 que le raisonnement était erroné. Mais le principe était intéressant :

On construisit un ensemble fini de configurations inévitables et on montre que chacune d'elles est réductible. C'est ce qu'on fait Appel et Haken grâce à des procédés de réductibilité dus à Birkhoff et Heesch. Puisqu'on ne peut pas démontrer la réductibilité d'un sommet de degré 5, il suffit de le remplacer par des configurations inévitables plus complexes. Si l'une d'elles n'est pas réductible on la remplace de nouveau. On sinsi construit progressivement un ensemble de 1877 configurations qui, par chance, ont fini par être toutes réductibles par les méthodes employées. La plus grosse possédait 14 sommets et a conduit à l'étude de 192921 configurations possibles. Il aurait fort bien pu se faire que ce processus ne s'arrête pas, bien qu'un argument probabiliste permette d'affirmer que c'était peu probable. Par contre, ce résultat est un peu décevant car il n'apprend rien de plus sur les propriétés topologiques du plan. Cela explique que les travaux se poursuivent sur ce problème, notamment par l'usage des polynômes chromatiques.

(3) J.C. Fournier, *Le Théorème des quatre couleurs*, Revue du Palais de la Découverte. N° spécial 12, 1978, Épuisé.

LE GRAND THÉORÈME DE FERMAT

Dans un ouvrage de Diophante traitant d'équations résolubles en nombres entiers, Fermat mentionna, en regard du paragraphe sur l'équation $X^2 + Y^2 = Z^2$:

« Pour $n > 2$, il n'existe pas de solutions entières (non triviales) à l'équation $X^n + Y^n = Z^n$. J'en ai une preuve merveilleuse, mais elle ne tient pas dans la marge. »

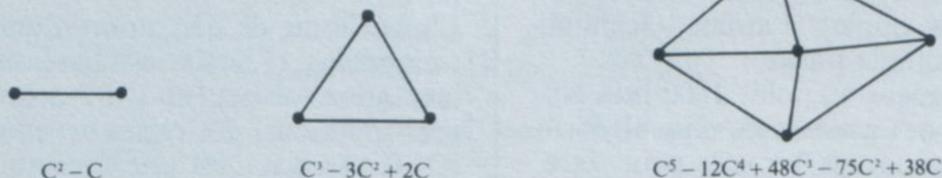
Malgré trois siècles de recherches acharnées, on ne sait toujours pas si ce théorème est vrai pour tout n , mais le calcul a permis de montrer qu'il est vrai pour $n < 30\,000$, et si l'on impose la condition supplémentaire que X, Y, Z soient premiers avec n , il est vrai pour $n < 253749889$.

L'ensemble E est ici l'ensemble des valeurs de n et la propriété P concerne tous les triplets (X, Y, Z) . On a commencé par montrer qu'il suffisait de démontrer le théorème pour $n=4$ et pour n premier. Malheureusement ce nouvel ensemble E' est infini. Ensuite Kummer (1850) a montré que le théorème est vrai pour toutes les valeurs de n telles que les numérateurs des nombres de Bernoulli B_{2m} , pour $1 \leq m \leq (n-3)/2$, ne sont pas divisibles par n . La vérification de cette propriété P' est alors accessible au calcul !

Dans le second cas, on peut montrer que si $X^n + Y^n = Z^n$ a une solution, alors $(k^{n-1} - 1) \equiv 0 \pmod{n^2}$, pour $2 \leq k \leq 43$, ce qui est également vérifiable par le calcul.

LES POLYNÔMES CHROMATIQUES

Considérons un graphe G planaire triangulé possédant N sommets. Quel est le nombre P de façons de le colorier avec C couleurs, deux sommets adjacents n'étant pas de la même couleur ? Birkhoff, qui étudiait ce problème en liaison avec celui des quatre couleurs, montra en 1913 que P est la valeur, au point C , d'un polynôme $P(G, C)$ de degré N , appelé polynôme chromatique du graphe G . En voici quelques exemples :



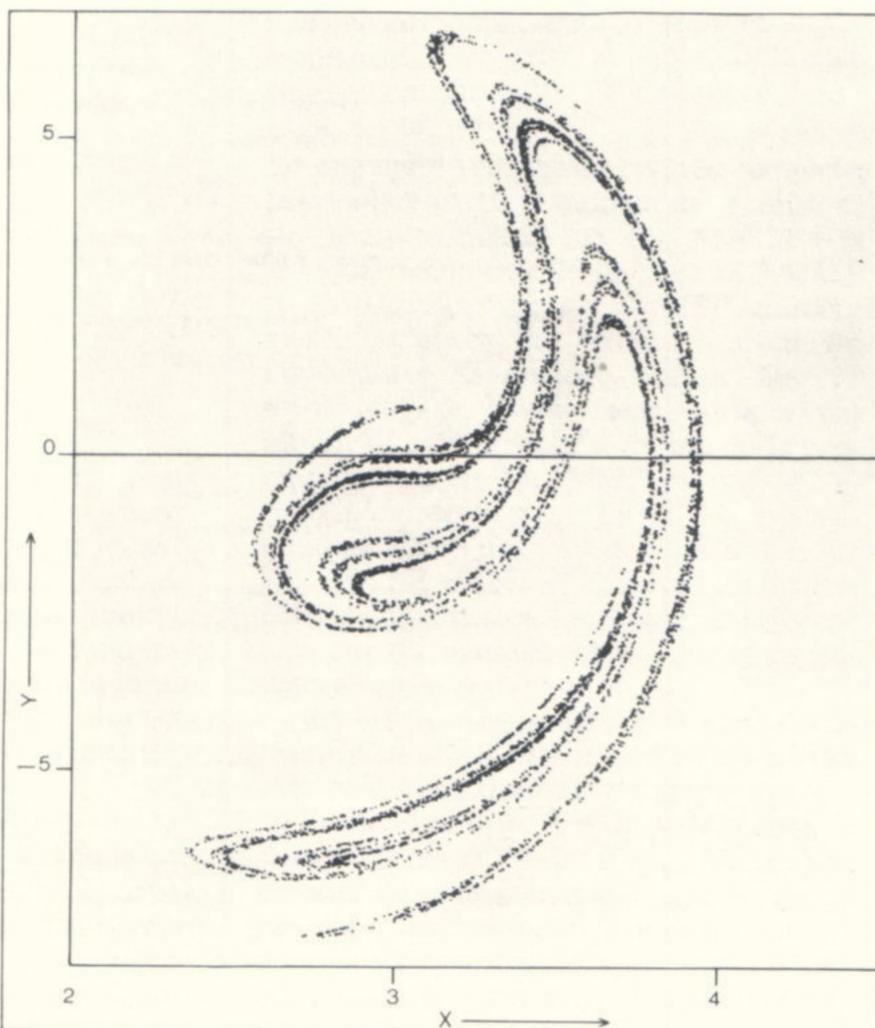
Résoudre le théorème des quatre couleurs revient à montrer que les polynômes chromatiques des graphes planaires triangulés n'admettent pas la racine $C=4$. On possède de nombreux résultats sur ces polynômes — par exemple il existe un terme de chaque degré et les signes sont alternés — mais on ne sait toujours pas démontrer le résultat essentiel. Par contre on sait les calculer et, s'ils n'ont de signification intuitive que pour les valeurs entières positives de C , rien n'empêche de calculer aussi leurs racines, réelles ou complexes. C'est ce qu'a fait Tutte pour une centaine de graphes. Il a alors constaté que chaque polynôme avait une racine proche de 2.618 et que cette racine en était d'autant plus proche que le degré du polynôme était élevé (4).

Ce résultat expérimental lui suggéra que ces racines étaient groupées autour du carré du nombre d'or ϕ , racine de l'équation $C^2 - C - 1 = 0$. Il a ensuite effectivement démontré que l'on avait l'encadrement : $0 < P(G, \phi) \leq \phi^{5-n}$, pour tout graphe G planaire triangulé, à n sommets.

Dans une moindre mesure, d'autres racines se groupent également autour de la valeur 3,24. (appelée par Tutte nombre d'argent), sans que l'on sache exactement pourquoi. Un de ses étudiants, Beraha, a suggéré que les nombres 2,618..., 3, 24..., etc qui interviennent sont en fait les termes de la suite $B_n = 2 + 2 \cos(2\pi/n)$, suite qui converge effectivement vers 4 !!!

Comme le mentionnait Tutte, à propos de la recherche d'un contre exemple au théorème des 4 couleurs et avant la démonstration de Appel et Haken : « Il est intéressant de comparer ces deux méthodes d'attaque. Dans la première (configurations réductibles), si vous ne réussissez pas vous n'obtenez rien. Dans la seconde, vous obtenez quand même des résultats théoriques sur les polynômes même si vous échouez pour $C=4$, ce qui permet sans doute d'approcher la structure même du problème. Ce qui importe en Mathématiques, ce sont les structures, pas les résultats isolés ».

(4) W. Tutte *Map coloring problem and chromatic polynomials*, American Scientist, Vol. 62, 1976.



« L'attracteur » de Yoshisuke Ueda, obtenu en résolvant l'équation différentielle $x'' + 0,1 x' + x^3 = 12 \cos t$. Les points affichés ont pour coordonnées $x(t), y(t) = x'(t)$ pour $t = 2n$. cf. D. Ruelle. *Strange attractors*. Mathematical Intelligencer. Springer Vol. 2, n° 3, 1980.

DOSSIER

Outre la démonstration de théorèmes, les ordinateurs peuvent également assister l'imagination (pourtant fertile) des mathématiciens en leur fournissant suffisamment de résultats pour leur suggérer de nouveaux théorèmes.

On a ainsi énoncé la conjecture suivante sur la valeur de l'exposant du p ième nombre de Mersenne premier : il serait de l'ordre de $1,5$ puissance p . Malheureusement, rien d'autre que l'observation ne vient étayer une telle conjecture, elle n'a aucun support théorique et rien ne suggère un moyen permettant de démontrer ce résultat. C'est donc pour l'instant ce qu'on pourrait appeler une conjecture stérile. On notera à ce propos que les découvreurs de grands nombres de Mersenne ont pris l'habitude d'énoncer des conjectures sur ces nombres. Je vous laisse méditer sur la dernière, due à Slovincski : « Il y aura toujours plus de conjectures sur les nombres de Mersenne que de nombres de Mersenne premiers connus ! ».

Il existe cependant quelques exemples où les résultats obtenus par l'ordinateur ont conduit à conjecturer, puis à démontrer, des théorèmes. L'un des plus spectaculaires (et inattendu) est certainement le théorème de Tutte sur les polynômes chromatiques.

Les conjectures valent essentiellement ce que valent les idées qui y ont conduit et les théories dans lesquelles elles s'insèrent, et les calculs ne servent, le plus souvent, que de garde-fous : un contre-exemple évite de chercher trop longtemps à démontrer un théorème faux.

Un exemple maintenant célèbre a été fourni il y a quelques années, réfutant une ancienne conjecture d'Euler qui affirmait (sans doute sans raisons très sérieuses) que l'équation :

$X^5 + Y^5 + Z^5 + T^5 = U^5$ n'avait pas de solutions en nombres entiers.

Lander et Parkin, grâce à un CDC 6600, en ont trouvé une :

$X = 27 ; Y = 84 ; Z = 110 ; T = 133 ; U = 144$ et je crois que l'on a trouvé depuis des solutions paramétriques !

Signalons enfin que la capacité qu'ont les ordinateurs de pouvoir répéter un très grand nombre de fois les mêmes opérations, sans se lasser et sans trop d'erreurs, permet quelquefois de découvrir de nouveaux phénomènes, notamment grâce à l'usage de supports graphiques. On leur doit par exemple la découverte de la constante de Feigenbaum ou celle des « attracteurs étranges » obtenus par itérations de transformations ponctuelles du plan et l'on peut raisonnablement supposer que la fécondité de ces nouveaux instruments mis au service des mathématiciens sera croissante dans les années à venir.

Jean Brette
Section Mathématiques
Palais de la Découverte

ORDINATEURS ET CALCULS SYMBOLIQUES

L'origine du problème

Le domaine privilégié — ou perçu comme tel — du calculateur électronique, de la calculatrice de poche à l'ordinateur, est le maniement des nombres décimaux, facilement représentables en machine. Quelle déception pour l'élève en classe de Mathématique, de recevoir 0,201680672 comme résultat du calcul $1/7 + 1/17$, ou d'apprendre que la racine positive du trinôme $x^2 - x - 1$ ressemble à 1,618033989, ce qui lui attirera les foudres de son professeur, resté fidèle — à juste titre — à une conception « exacte » (lorsqu'il faut se peut) du calcul algébrique. Cet enseignant, qui découvre la micro-informatique, s'empresse de mettre tous ses chers algorithmes en machine, et déchante en essayant de programmer un calcul de fonction dérivée exacte, dont la méthode est tellement « formelle » que son implémentation informatique ne devrait pas être si difficile à écrire. Et que dire à ce chercheur en Théorie des Nombres qui aimerait avoir, vite et facilement, un développement limité au quatrième ordre à coefficients rationnels...

Bref, il fallait réagir, la communauté scientifique — et pédagogique — attirée par la puissance des gros ordinateurs, espérant mieux que le déluge usuel d'approximations décimales.

Plus sérieusement, il ne semble pas trop hasardeux d'affirmer qu'un algorithme simple doit pouvoir être écrit simplement. Le problème est que la plupart des langages de programmation qui ont été popularisés en micro-informatique (BASIC en premier lieu) travaillent sur des données structurées de manière à permettre des transformations rapides sur les variables du calcul (par le mécanisme d'affectation), mais pas directement sur les expressions algébriques elles-mêmes. Les langages de la famille d'ALGOL (comme PL1, Pascal ou Ada) autorisent des structures de données plus riches (les arborescences), mais leur manipulation effective reste souvent du ressort du spécialiste.

LISP et le traitement des « listes »

Vers 1960, grâce aux chercheurs du Laboratoire d'Intelligence Artificielle du MIT,

et en particulier à l'un d'entre eux, John Mac Carthy, apparut un nouveau langage, LISP (LIST Processor), dont l'originalité consistait à exploiter à fond la structure arborescente des expressions algébriques ou linguistiques — parallèlement à des recherches logiques plus formelles sur le « lambda-calcul » de Church, (1, 2) — et à ne manipuler en conséquence que ce type de structure de données : la « liste » (arborescence linéaire), dont les éléments sont des « atomes » (pensez aux nombres ou aux identificateurs)... ou des listes.

Par exemple $L = (1\ 2\ (3\ 4)\ 5)$ est une liste à 4 éléments, les atomes 1, 2, 5 et la liste (3, 4). On peut accéder au premier élément de la liste L par l'opération CAR et à la liste privée de son CAR par l'opération CDR :
 $(CAR\ L) \rightarrow 1$
 $(CDR\ L) \rightarrow (2\ (3\ 4)\ 5)$

Chaque atome est pourvu d'une « liste de propriétés » (p-liste) qui contient des attributs de cet atome, et la valeur correspondante. Par exemple, l'atome Jeannot pourra avoir comme p-liste :

(lapin vrai longueur 40 poids 1500 prix 60)
 Pour aborder des problèmes plus algébriques, l'expression $(X + Y) * Z$ peut être représentée par les notations :

PREFIXE	INFIXE	POSTFIXE
de liste :	de liste :	de liste :
$(*(+ X Y)Z)$	$((X + Y)*Z)$	$(X Y +)Z*$

Les habitudes des calculatrices reconnaîtront vite les deux dernières, supports d'une quereelle d'école bien connue... Nous les laisserons dos à dos pour nous intéresser à la première, utilisée par LISP.

Le langage a un principe de fonctionnement interactif très simple : l'utilisateur entre une liste L que l'interprète LISP perçoit comme application d'une fonction (le CAR de L) à une liste d'arguments (le CDR

de L), ce processus étant à son tour répété pour évaluer chacun des arguments, pouvant eux-mêmes être des listes.

Une valeur de fonction composée, telle $f(x,y) = x + g(x,h(y))$, se fera par un appel d'évaluation de la liste $(+ (g (h y)))$. Cette structure de l'interprète explique la dénomination de « langage applicatif » donné à LISP et à ses dérivés. Le langage est ainsi spontanément récursif, et encourage ce style de programmation ; les sauts à étiquette (GO) sont licites, mais pas en odeur de grande sainteté, et tendent à être abandonnés dans les dernières versions (3). L'isolement dans lequel LISP a été tenu pendant de nombreuses années (celui des milieux de l'Intelligence Artificielle) a souvent conduit ses partisans à des positions très tranchées ; il faut avouer que les arguments sont de poids, et que ce langage — à la limite cet état d'esprit (pensez à LOGO) — semble assurément d'un bon développement (4, 5).

L'algorithme de dérivation d'une fonction rationnelle, récursif, est basé sur l'implémentation (on peut en LISP définir ses propres fonctions) des règles usuelles :
 $(DER\ 0) = 0$ si a est une constante
 $(DER\ X) = 1$ si x est la variable
 $(DER (+ F G)) = (+ (DER F) (DER G))$
 $(DER (- F G)) = (- (DER F) (DER G))$
 $(DER (* F G)) = (+ (* (DER F) G) (* F (DER G)))$
 $(DER (/ F G)) = (/ (- (* (DER F) G) (* F (DER G))))$

Ceci n'est pas tout et fait du LISP, mais y ressemble fortement (cf encadré 1). Que les parenthèses ne vous effrayent pas : on s'y habitue, et de plus certaines versions (comme LOGO) sont — pour d'évidentes raisons pédagogiques — beaucoup moins pointilleuses sur l'usage des parenthèses et

```

Encadré 1
*****

L'algorithme de dérivation d'une fonction rationnelle en LISP:

(OE DER L)
(COND
  ((ATOM L) (IF (EQL 'X) 1 0)) ; 'X pour ne pas chercher valeur(')
  ((EQL 'CAR) L) ; '
  ((EQL 'CDR) L) (DER (CDR L)) ; CDR est CAR (CDR) etc...
  ((EQL 'CAR) L) ; '
  ((EQL 'CDR) L) (DER (CDR L)) ; '
  ((EQL 'CAR) L) ; '
  ((LIST 'L) (LIST 'L (CDR L) (DER (CDR L)))
  ((LIST 'L) (LIST 'L (CDR L) (DER (CDR L)))
  ((EQL 'CAR) L) ; '
  ((LIST 'L) (LIST 'L (CDR L) (DER (CDR L)))
  ((LIST 'L) (LIST 'L (CDR L) (DER (CDR L)))
  ((LIST 'L) (CDR L) (DER (CDR L)))
  ((LIST 'L) (CDR L) (DER (CDR L)))
  ))

Voici le même en PD-SIMP, la fonction étant entrée sous forme préfixe:

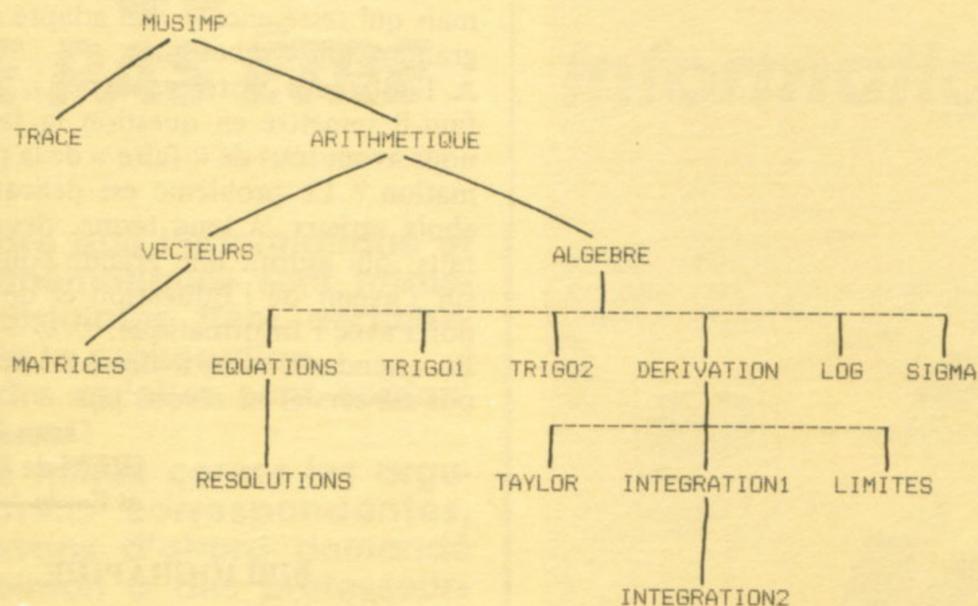
FUNCTION DER(L)
  WHEN ATOM(L)
  WHEN L='X, 1 EXIT
  0 EXIT
  WHEN FIRST(L)='+
  LIST('DER(SECOND(L)),DER(THIRD(L))) EXIT
  WHEN FIRST(L)='-
  LIST('-(DER(SECOND(L)),DER(THIRD(L))) EXIT
  WHEN FIRST(L)='*
  LIST('*(LIST('L,SECOND(L),DER(THIRD(L)))
  WHEN FIRST(L)='/
  LIST('(/(LIST('L,DER(SECOND(L)),THIRD(L)),
  LIST('L,DER(SECOND(L)),DER(THIRD(L))))
  LIST('L,THIRD(L),THIRD(L))) EXIT
ENDFUN

Et voici la fonction de conversion infixe -> préfixe:

FUNCTION INF-PRE(L)
  WHEN ATOM(L), L EXIT
  LIST(SECOND(L),INF-PRE(FIRST(L)),INF-PRE(THIRD(L)))
ENDFUN

Exemples:
? L: '(X (X + 1)) ; L reçoit une expression infixe ; * sans echo %
? DER-INF-PRE(L) ; % pour la notation en liste %
? (+ (+ (+ (+ X 1)) (X + 1) 0)) ; % simplifier...%
? (+ (+ (+ (+ X 1) 1) 1) 1) ; % en infixe usuel %

```



L'arbre des fichiers MU-MATH. Chaque module suppose chargé tous les précédents du même rameau.

le préfixage des expressions algébriques. Il faudrait bien sûr écrire une fonction « simplifier » afin de réduire des expressions comme (+ F 0) ou (* F 1), puis des fonctions de conversion « infixe <—> préfixe » si l'utilisateur rechigne à penser de manière applicative... (cf encadré 1). Un ensemble de telles fonctions réalisant ce genre de manipulations symboliques constitue un « système de calcul formel ».

C'est sur cette base conceptuelle que sont apparus les premiers grands systèmes, dès les années 60 (par des simulations de gestion de listes en FORTRAN), puis surtout à partir des années 70 avec MACSYMA (au M.I.T.) et REDUCE (à Utah University), tous deux écrits en LISP.

Un exemple de factorisation sur MACSYMA :

$$X^{30} + 1 \rightarrow (X^2 + 1)(X^4 - X^2 + 1) \cdot (X^8 - X^6 + X^4 - X^2 + 1) \cdot (X^{16} + X^{14} - X^{10} - X^8 - X^6 + X^2 + 1)$$

en 0,4 sec.

Et sur nos micros ?...

Réjouissons-nous haut et fort, nous n'avons plus à rougir de nos petits micro-ordinateurs. Il existe désormais sur le marché un excellent logiciel de calcul symbolique portable sur les machines utilisant le système d'exploitation CP/M. Son support n'est pas LISP, mais un cousin nommé MU-SIMP, possédant une syntaxe plus traditionnelle, et mieux adapté au calcul formel.

Développé sous le soleil d'Honolulu par A. Rich et D. Stoutmeyer (6), il semble que MU-SIMP soit du domaine public, le copyright portant sur MU-MATH; maintenant distribué, avec MU-SIMP en prime, par la firme américaine MICROSOFT, et disponible chez tous les bons revendeurs. La ver-

sion que nous avons testée tourne sur Apple II, avec 64K de mémoire, et une carte d'interface Z-80.

MU-MATH consiste en fait en un lot de 16 fichiers (listables !...) sur disque qui sont appelés par l'utilisateur au moment approprié durant une session de travail (cf encadré 2). Cette modularité permet un gain appréciable de place en mémoire, mais le chargement de ces fichiers (écrits en MU-SIMP) s'accompagne d'une sorte de compilation optimisée, destinée à les représenter en mémoire sous forme de listes chaînées, et l'opération est malgré tout assez lente. Il est alors possible de sauver sur un autre disque cette version compilée (non listable), accessible bien plus rapidement. Dans le test (cf encadré 3), les chargements des différents modules ne seront pas toujours mentionnés : pour obtenir un développement limité par exemple, il faudra charger en particulier le module « TAYLOR », etc.

Le manuel d'utilisation est très complet : la première partie comprend des précisions sur chaque module de MU-MATH ; la seconde documente le langage MU-SIMP et, ainsi que les ouvrages sur LISP, contient une simulation de certaines primitives importantes du langage, écrites dans ce même langage, ce qui s'avère un excellent moyen d'apprentissage à la fois de la programmation en MU-SIMP et des opérations d'entrées-sorties. Enfin, les concepteurs ont prévu une série de « leçons », sur MU-MATH et MU-SIMP, dont les fichiers correspondants sont sur disque, et le listage à la fin du manuel. Deux regrets cependant :

— les « listes de propriétés » sont documentées in abstracto, et un lien plus étroit avec MU-MATH aurait été apprécié.

— l'utilisateur dispose d'un certain nombre de « variables de contrôle » au moment de prendre des décisions quant à la forme de certaines expressions. Là aussi, faute d'explications suffisantes, il faut une certaine pratique pour bien en saisir le fonctionnement.

Dans les deux cas, l'ultime ressource est de se plonger avec stupeur et ravissement (?) dans les listages (plusieurs dizaines de pages sur imprimante) des fichiers MU-MATH. Enfin, et surtout, il manque un bon éditeur de fonctions ; ceci est néanmoins compréhensible lorsqu'on sait la place-mémoire qu'il prendrait...

Et la pédagogie ?...

N'ayant pas encore eu personnellement l'occasion de placer des élèves en situation devant ce logiciel, je vais me borner à jouer le rôle de l'avocat du diable, en explicitant deux idées qui font leur chemin, à pas lents bien sûr, et en marge des discours usuels sur les rapports entre Informatique et Éducation (5, 7).

Sur MU-MATH d'abord : apprendre à un élève le maniement d'un logiciel de calcul symbolique devrait contribuer à développer chez ce dernier *des aptitudes à conjecturer* que la pratique du calcul décimal, avec ses problèmes de contrôle de la précision, ne favorise guère. Les êtres mathématiques seront alors perçus sous un angle formel et combinatoire, et l'élève manipulant des constantes fondamentales comme e et i, ou des développements limités à coefficients rationnels, apprendront mieux à percevoir les « propriétés » dont ils sont tissés, plutôt qu'une valeur à 10 décimales, qui reste bien entendu vitale pour l'arpenteur ou l'astronome.

Ce point de vue « algébrique » n'a d'intérêt que si l'élève n'est pas un spectateur passif des miracles symboliques, mais y prend une part active, soit en y gagnant une certaine vision des mathématiques, soit en écrivant, ou modifiant pour un besoin particulier, tel ou tel algorithme « formel » : manipulations (récurives) d'ensembles sous forme de listes, de fonctions plusieurs fois dérivables, de polynômes à une indéterminée à coefficients entiers (ou entiers de Gauss : en voient-ils réellement beaucoup d'autres ?...), séries de Taylor propices à des conjectures multiples, de la convergence tout court à la « convergence en machine » (la série harmonique $1 + 1/2 + 1/3 + 1/4 + \dots$, dont la somme est infinie, admet une représentation décimale convergente) qui permettra de soulever alors les vrais problèmes numériques, parfois fort difficiles.

Plus généralement, il semble que l'un des moyens d'apprentissage de la programmation consisterait à utiliser des langages à vocation symbolique, capables de traiter des problèmes variés, allant des algorithmes numériques classiques aux problèmes purement formels, en manipulant des objets caractérisés par leurs propriétés : le langage LISP vient bien entendu à l'esprit, sous une version peut être moins « sèche » comme LOGO par exemple (*) qui autorisera toute pratique numérique, formelle ou graphique en classes scientifiques, sans pour cela que les professeurs de langue ou d'histoire se sentent coupables d'activités non-numériques...

Ces langages sortent de l'ombre dans laquelle BASIC les a — hélas — tenu trop longtemps. LSE, qui est un peu une sorte de compromis entre BASIC et Pascal, utilise une structure de données très performante, la chaîne de caractères, mais il sem-

ÊTRE OU NE PAS ÊTRE

Prétendre que l'informatique et les mathématiques sont totalement disjointes tient certainement de la provocation.

Prétendre qu'elles sont confondues aussi !

Afin de mieux cerner les argumentations correspondantes, nous avons d'abord demandé leur opinion à des professeurs de lycée. (La rédaction se gardera bien de trancher le débat, elle attend vos propres réactions !)

Mais, tout n'étant qu'affaire de définitions, il nous a semblé utile de consulter le professeur NIVAT sur le sujet ; voici son objective ordonnance.

Non ! Il n'est pas nécessaire d'être mathématicien pour "faire de l'informatique"

J'ai déjà fait la Grèce et l'Italie, j'ai fait mon service militaire, j'ai fait Corneille, Molière et Racine, puis-je aussi "faire de l'informatique" ? Que faut-il mettre alors sous cette expression magique ?

Pour le profane, "faire de l'informatique" c'est d'abord savoir se servir d'un ordinateur, c'est-à-dire manœuvrer un interrupteur, appuyer sur des touches, placer une disquette, connaître les quelques commandes nécessaires pour faire fonctionner un programme, éventuellement brancher quelques câbles, toutes activités où le mathématicien n'est pas plus avantage que le littéraire. Il s'agit essentiellement de manipuler un objet technique en se servant d'un mode d'emploi. Sans nier qu'il y ait des blocages à ce niveau, on peut cependant noter que cette tâche n'est pas plus difficile que l'utilisation d'un appareil audio-visuel ou électro-ménager. Seul le mystère qui entoure l'ordinateur et la crainte de destructions irrémédiables en cas de fausses manœuvres peuvent paralyser l'utilisateur. Les professeurs de sciences expérimentales qui savent que les expériences ne réussissent pas toujours sont ici mieux armés que les mathématiciens ou les littéraires. Ils n'hésitent pas à recommencer et à procéder par tâtonnements successifs, ce qui leur permet de suppléer à l'insuffisance des notices techniques qui sont rarement faites pour l'utilisateur débutant.

mathématicien et/ou informaticien



Il en est souvent de même pour la connaissance des langages de programmation, car les manuels sont rarement complets et l'apprentissage de tout langage passe par une expérimentation de ce qu'est capable de réaliser chaque instruction ou fonction d'un langage particulier sur une machine particulière. Cette situation ne devrait pas surprendre un linguiste qui a pris l'habitude de distinguer langue et parole et qui sait qu'un mot n'a de signification que dans un contexte.

Mais cet empirisme inévitable, parce qu'il est lié à une réalisation concrète, n'empêche pas l'informatique d'être aussi la science du traitement de l'information. Or le langage naturel autant que le langage mathématique permet de traiter de l'information. Les linguistes autant que les mathématiciens doivent distinguer syntaxe et sémantique, signifiant et signifié.

Les activités d'analyse et de programmation exigent des qualités de rigueur, de logique et une extrême minutie. Elles nécessitent peu de connaissances strictement mathématiques, sauf lorsqu'il s'agit de traiter des applications proprement mathématiques. Par contre, une formation uniquement mathématique ne permet pas de traiter valablement une application d'un domaine non mathématique, même si ce domaine a parfois recours à l'outil mathématique. Il me paraît indispensable, pour travailler dans quelque discipline que ce soit, de posséder à la fois une connaissance approfondie de cette discipline et une compétence informatique. Par exemple, les littéraires sont les mieux désignés pour analyser le système de la conjugaison latine ou française, pour déterminer les différentes phases d'un traitement et même pour en programmer la réalisation. Quoique cette dernière étape ne soit qu'un codage, elle n'est pas non plus hors de portée pour un littéraire qui a pratiqué le thème grec ou le thème latin, exercices où les contraintes syntaxiques sont infiniment plus complexes. N'est-ce pas d'ailleurs un latiniste

(1) qui a trouvé le mot "ordinateur" c'est-à-dire machine à ordonner les informations pour désigner ce que les anglo-saxons n'appellent que "computer", c'est-à-dire calculateur.

Je ne concède un avantage aux mathématiciens que dans le domaine de l'informatique théorique et dans la conception des systèmes d'exploitation où la formalisation atteint un niveau très élevé.

Enfin n'oublions pas que l'informatique revêt aussi un aspect culturel de plus en plus important et que les philosophes, les historiens et les littéraires ont vocation pour le prendre en compte s'il est vrai que les processus d'apprentissage, les modes de pensée et les relations sociales risquent d'en être affectés profondément.

Enseignants de toutes disciplines, n'ayez pas de complexes. Ne laissez pas aux mathématiciens ce qui est un bien commun. Malgré les constructeurs qui ne savent pas présenter leur matériel d'une manière claire, malgré les informaticiens qui enseignent l'informatique en se servant uniquement d'exemples mathématiques, je vous dis qu'il n'est pas difficile d'apprendre l'informatique, qu'il vous est possible de résoudre vous-mêmes les problèmes de votre discipline et même d'enseigner l'informatique en utilisant des exemples empruntés à votre expérience.

Pierre Muller
*Professeur de français
Institut National
de Recherche Pédagogique*

(1) C'est en effet Jacques Perret, professeur à la Sorbonne, qui proposa ce mot à la suite d'une demande de la compagnie IBM.

ÊTRE INFORMATICIEN ?

Est-ce être capable de mettre en route un ordinateur et de jouer contre lui aux échecs en utilisant un programme donné par le constructeur ?

Suffit-il de savoir conduire une voiture pour être mécanicien ?

Suffit-il de savoir téléphoner à New-York pour être spécialiste des télécommunications ?

Être informaticien c'est au minimum :

- savoir analyser le problème à résoudre ;
 - savoir choisir ou définir et mettre en œuvre un algorithme performant en fonction du problème à résoudre et du matériel utilisé ;
 - savoir écrire le programme correspondant ;
 - savoir prouver que le programme résout bien le problème posé ;
 - connaître les limites d'utilisation du programme, dues d'une part à l'algorithme utilisé d'autre part aux limites calculatoires de la machine.
 - savoir déceler les erreurs dues à ces limites ;
 - savoir, éventuellement, adapter le programme d'un autre à ses propres besoins.
- Ces savoir-faire nécessitent à l'évidence la

possession de quelques notions (ou de quel que esprit) mathématiques. Comme toutes les sciences, l'informatique s'appuie sur un outil mathématique. Il est donc nécessaire pour dessiner et comprendre ce qu'on fait en informatique de connaître cet outil. Il ne viendrait pas à l'idée d'un physicien de renier les mathématiques. Il en est certainement de même pour les informaticiens. Ceux-ci savent apprendre les notions mathématiques au moment où ils en ont besoin. S'ils n'ont pas eu de formation mathématique profonde au départ, ils apprennent à chercher et à acquérir au moment opportun le savoir mathématique qui leur est nécessaire. Même si les apparences sont parfois trompeuses, ceux qui font de l'informatique sont ou deviennent naturellement des (petits) mathématiciens.

André Meyer
Professeur de mathématiques
au lycée Victor Duruy

INFORMATIQUE ET MATHÉMATIQUE DANS L'ENSEIGNEMENT

Les quelques réflexions que voici s'appuient en particulier sur les observations que j'ai pu faire lors d'un très récent séjour aux États-Unis. Il y a désormais deux informatiques qui paraissent devoir se séparer de plus en plus, mais devraient au contraire se rapprocher.

- la première est l'art, ou la science, du calcul.
Cette informatique est très liée aux Mathématiques, elle en est le prolongement, le complément direct. On peut aussi la voir comme partie intégrante de la Mathématique.

- la seconde est beaucoup plus un outil de communication, d'archivage, de travail qui, à terme, va bouleverser toutes les conditions de production intellectuelle, scientifique ou non.

De fait, il y a une troisième informatique sur laquelle, hélas, nous avons très peu de prise : c'est l'informatique des "grs systèmes", commutateurs téléphoniques, systèmes de réservation de places de train ou d'avion, fichiers de police, etc.

Revenant aux deux premières, on peut faire diverses constatations.

La première et principale source d'algorithmes est la mathématique.

L'étude d'un algorithme, la mesure de son efficacité comme la preuve de sa validité, sont de la mathématique. Il importe peu que ce soit aujourd'hui une mathématique "non classique", elle le deviendra bientôt. La seule séparation entre la mathématique qu'on enseigne et l'informatique algorithmique

est une différence dans les niveaux d'abstractions où se situent l'une et l'autre : la théorie des graphes en fournit un excellent exemple.

Selon qu'un graphe est représenté en machine par une liste d'arêtes ou par une matrice d'incidence, le même algorithme portant sur un graphe défini comme relation sur un ensemble de sommets prend des aspects très différents (et peut être effaçé dans un cas, réadhibitoire dans un autre).

Tout cela est désormais bien connu, l'algorithmique a fait ces dix dernières années des progrès considérables : malheureusement, si elle a pénétré suffisamment la théorie des nombres et l'analyse, elle n'a pas vraiment entamé l'algèbre, domaine pourtant très propice au développement d'une algorithmique féconde.

Pour parler toujours de la première informatique, il ne suffit pas d'avoir identifié un algorithme ou d'en avoir démontré les vertus. Il faut encore le programmer, c'est-à-dire l'exprimer dans un certain langage. Là, on tombe dans un domaine qui s'apparente beaucoup plus à la logique mathématique qu'à autre chose : un programme n'est jamais qu'une formule qu'il convient d'interpréter de deux façons. Une sémantique mathématique associe à un programme un algorithme, mathématiquement défini, et une sémantique opérationnelle qui, elle, dit ce qu'une machine doit faire pour trouver le résultat en un point. Les problèmes sont alors d'équivalence de ces sémantiques, de preuve, abstraite, qu'un programme représente bien l'algorithme que l'on a voulu lui faire représenter, d'optimalité ou au moins d'efficacité suffisante de la sémantique opérationnelle.

La deuxième informatique est celle de la communication. Elle fait en ce moment des progrès tout à fait spectaculaires et l'on doit s'attendre à ce qu'elle en fasse bien plus encore et modifie profondément la vie de tout le monde. Je lui vois trois points très forts en ce moment.

- Les systèmes de traitement de textes, désormais accessibles à tous (l'industrie du livre s'est elle depuis longtemps informatisée) : grâce à un tel système qui délègue à la machine tout le travail fastidieux de la mise en page, on gagne un temps précieux et une souplesse inconnue des moyens de dactylographie ou de composition traditionnels. C'est la fin de la rature et des pattes de mouche. Les avantages sont encore bien plus évidents si l'on travaille à plusieurs sur un même texte dont le dernier état peut être mis instantanément à la disposition de tous les intéressés, pour peu qu'ils soient reliés à un même réseau.

- Les systèmes graphiques : comme chacun sait "un bon dessin vaut mieux qu'un long discours". Les systèmes de représentation et manipulation graphiques offrent un nouveau mode de communication et plus encore peut être de conception. Il s'y rattache le fascinant problème de la perception de l'univers réel qui est le problème fondamental de la robotique avec le passage de l'analyse de scènes photographiques à la prise de décision (reconnaissance d'un objet, contournement d'un obstacle, cheminement vers un but à atteindre...)

- Les bases de données, distribuées ou non. Bases de données ou bases de connaissances : la machine autorise l'accumulation et

l'exploitation de masses importantes d'informations, leur partage aussi entre de nombreux usagers. L'interrogation et l'utilisation de ces bases est à mes yeux le cœur de ce que l'on peut appeler l'intelligence artificielle et ne doit guère surprendre le mathématicien : n'a-t-on pas appris à faire jouer les hypothèses, toutes les hypothèses et rien que les hypothèses pour construire une démonstration. L'extraction d'une information d'une base de connaissance n'est rien d'autre que cela (même si parfois le problème n'est qu'implicitement formulé). Tout un pan de cette activité de constitution et exploitation d'une base de données se rattache directement à la logique mathématique : on y fait des unifications, substitutions, déductions, résolution comme dans le plus traditionnel des systèmes axiomatiques.

En conclusion, il me semble que ce que j'ai appelé la première informatique, à savoir, l'étude des algorithmes et de leur représentation par des programmes est si proche des mathématiques qu'elle pourrait aussi bien être considérée comme en faisant partie. L'analyse numérique, que je préfère appeler l'algorithmique numérique, en fait bien partie : peut-être faut-il seulement que la mathématique (française) s'ouvre plus largement sur la combinatoire, l'étude des structures algébriques non commutatives et "pauvres" (monoïdes et magmas) et sur la logique dont je ne peux que déplorer qu'elle soit aujourd'hui si peu enseignée dans nos Universités et Grandes Écoles.

Pour moi, cette informatique, c'est une mathématique concrète, expérimentale, qui n'en est qu'à ses débuts mais est porteuse d'immenses espoirs.

La seconde informatique est dès aujourd'hui un outil d'une incomparable efficacité (sauf - il faut le dire - que son usage en France est très en retard sur ce qu'il peut être aux États-Unis, dans le Royaume-Uni et sans doute au Japon) dont les mathématiciens doivent apprendre à se servir et tireront ce faisant le meilleur.

Quant à la troisième informatique, celle des "grs" systèmes, dont la dimension même empêche aujourd'hui la formalisation, elle est sans doute plus active d'ingénieurs que de chercheurs scientifiques. Elle implique un travail considérable, collectif, clérical subalterne et dont une large partie est assez obscure. Nous ne devrions pas l'oublier pour autant : un des buts essentiels que l'on peut assigner à l'informatique mathématique telle que je la conçois est d'aider à la conception, l'écriture, la vérification de ces gros systèmes. Ma conviction est que ce seront des mathématiciens informaticiens qui feront progresser ce domaine peu à peu, c'est mon dernier souhait, qu'ils veuillent bien adopter une mentalité d'ingénieur.

Maurice Nivat,
professeur à l'Université
Paris 7



Voici d'abord un petit programme extrait du distrayant petit livre de Serge Pouts-Lajus "L'ordinateur en fête"*. Il contient de très nombreux programmes souvent courts, bien souvent drôles...



PETITE ANNONCE

Le déchiffrement et la composition de petites annonces est une activité qui présente parfois des difficultés imprévues. Les habitués savent que ch. signifie « cherche » et non pas « chapeau » et que j.f. signifie « jeune femme » et non pas « jaune foncé ». La méthode courante consiste à supprimer la plupart des voyelles.

La question à laquelle vous allez tenter de répondre est la suivante : si l'on supprime toutes les voyelles, un texte reste-t-il lisible sans ambiguïté ?

On sait que l'Arabe écrit n'utilise pas les voyelles. Pourrait-on faire de même en Français ?

Probablement pas, car il serait extrêmement fâcheux que vous confondiez une fille grande et maigre avec une femme gironde et mégère.

Objet du programme : Un joueur entre son message et le programme supprime les voyelles avant de donner le résultat au lecteur qui doit déchiffrer. L'accès à la solution est codé par le rédacteur.



Programme sur T07

```

10 CLS:SCREEN 3,4,4:CLEAR 250
20 PRINT "ENTREZ VOTRE MESSAGE"
30 INPUT M$
40 INPUT "CODE SECRET D'ACCES A LA Solut
ION:";C$
50 CLS
60 PRINT "DECHIFFREZ CETTE ANNONCE"
65 PRINT"EN RETROUVANT LES VOYELLES"
70 ?
80 '*** SUPPRESSION DES VOYELLES ***
90 ?
100 LOCATE 0,8
105 ATTRB1,1
110 FOR Z=1 TO LEN(M$)
120 L#=MID$(M$,Z,1):L=ASC(L$)
130 IF(L-65)*(L-69)*(L-73)*(L-79)*(L-85
)*(L-89)=0 THEN 150
140 PRINT L$+" ";
150 NEXT Z
160 '***
170 '*** SOLUTION ***
180 '***
185 ATTRB0,0
190 LOCATE 0,20
200 INPUT "SOLUTION-CODE SECRET:";R$
210 IF R$=C$ THEN PRINT M$ ELSE 190
220 END
    
```

...Parties vues Parties Cachée...

Dessiner un polyèdre

Dans ce programme, le repère choisi a son origine au centre du polyèdre qui se trouve être également le centre de l'écran.

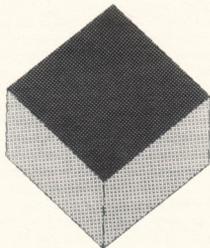
L'axe x' ou x est l'axe horizontal de l'écran
L'axe y' ou y est l'axe vertical de l'écran
Et l'axe z' ou z traverse l'écran et l'observateur

Il y a quelques astuces dues au puissant langage HP ;

- Mat $A = B * C$: Produit matriciel
- Mat Plot qui permet de tracer les graphiques dans un tableau afin de remplir les contours (FILL) selon une colorimétrie définie dans "Area Color t, s, e," dont les 3 paramètres sont teinte, saturation, luminosité.

Mais on remarquera aussi les petits trucs suivants :

- 2 tableaux servent à définir le polyèdre : 1 tableau nommé "Sommets" contient les coordonnées des sommets
- 1 tableau nommé "Faces" contient les sommets de chaque Face
- Le tableau Somproj contient les coordonnées des sommets après transformations (ici rotations dans l'espace) mais le tableau décrivant les faces reste valable quelle que soit la transformation.
- L'élimination des parties cachées est faite en orientant ces faces, dans les initialisations, par la liste ordonnée de leurs sommets. Les faces vues "tournent" dans le sens trigo ; les faces cachées "tournent" en sens contraire (*). Le produit vectoriel de deux vecteurs d'une face permet de savoir si la face est vue ou cachée : mieux, la projection de ce produit vectoriel sur z' ou z donne l'intensité avec laquelle la face est éclairée... !

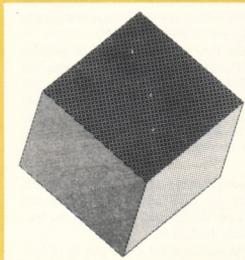


(*) Ce "truc" marche évidemment pour tous les polyèdres convexes mais aussi pour quelques autres.

```

10  OPTION BASE 1
20  DEG
30  DIM Somproj(8,3),Sommets(8,3),Faces(6,5),Rx(3,3),Ry(3,3),Rz(3,3),Buf(3,3),
  Transf(3,2),R(18,3)
40  |
50  |
60  |                               Le cube en perspective...
70  |
80  | Initialisations...
90  |
100 DATA 1,-1,-1,1,1,1,1,1,1,1,-1
110 DATA -1,-1,-1,-1,1,1,-1,-1,1,1,-1,-1
120 FOR S=1 TO 8
130  REDEF Sommets(S,1),Sommets(S,2),Sommets(S,3)
140  NEXT S
150 DATA 1,4,3,2,1
160 DATA 4,6,7,3,4
170 DATA 6,5,8,7,6
180 DATA 5,1,2,8,5
190 DATA 8,2,3,7,8
200 DATA 4,1,5,6,4
210 FOR F=1 TO 6
220  FOR S=1 TO 5
230  REDEF Faces(F,5)
240  NEXT S
250  NEXT F
260  |
270  |
280  PLOTTER IS "GRAPHICS"
290  GRAPHICS
300  SHOW -3,3,-3,3
310  |
320  Thetax=0
330  Thetay=30
340  Thetaz=45
350  Boucle:                               ! Pour faire tourner le CUBE
360  Cos=COS(Thetax)
370  Sin=SIN(Thetax)
380  Cos=COS(Thetay)
390  Siny=SIN(Thetay)
400  Cos=COS(Thetaz)
410  Sinz=SIN(Thetaz)
420  MAT R=IDN
430  Rx(2,2)=Cosx
440  Rx(3,3)=Cosx
450  Rx(3,2)=Sinx
460  Rx(2,3)=-Sinx
470  MAT Ry=IDN
480  Ry(1,1)=Cosy
490  Ry(3,3)=Cosy
500  Ry(3,1)=Siny
510  Ry(1,3)=-Siny
520  MAT Rz=IDN
530  Rz(1,1)=Cosz
540  Rz(2,2)=Cosz
550  Rz(2,1)=Sinz
560  Rz(1,2)=-Sinz
570  MAT Buf=R*Y
580  MAT Transf=Buf*Rz
590  MAT Somproj=Sommets*Transf          !!!!!
600  |
610  |
620  GCLEAR
630  FOR F=1 TO 6
640  V1=Somproj(Faces(F,2),1)-Somproj(Faces(F,1),1)
650  V1=Somproj(Faces(F,2),2)-Somproj(Faces(F,1),2)
660  Z1=Somproj(Faces(F,2),3)-Somproj(Faces(F,1),3)
670  V2=Somproj(Faces(F,3),1)-Somproj(Faces(F,1),1)
680  V2=Somproj(Faces(F,3),2)-Somproj(Faces(F,1),2)
690  Z2=Somproj(Faces(F,3),3)-Somproj(Faces(F,1),3)
700  GOSUB Product
710  Cach=Z3
720  IF Cach<0 THEN 850
730  PLOTTER IS OFF
740  PLOTTER IS RC
750  FOR S=1 TO 5
760  IF S=1 THEN MOVE Somproj(Faces(F,5),1),Somproj(Faces(F,5),2)
770  IF S<1 THEN DRWF Somproj(Faces(F,5),1),Somproj(Faces(F,5),2)
780  NEXT S
790  PLOTTER RC(*) IS OFF
800  PLOTTER IS ON
810  Lumin=23*FNNorm(X3,Y3,Z3)
820  AREA COLOR S/6,1,Lumin
830  MAT PLOT R,FILL
840  MAT PLOT R
850  NEXT F
860  WRIT 1000
870  Thetax=Thetax+10
880  Thetay=Thetay+5
890  Thetaz=Thetaz+2
900  GOTO Boucle
910  END
920  |
930  |
940  |
950  Product:      X3=Y1+Z2-Z1+Y2
960                Y3=Z1+X2-X1+Z2
970                Z3=X1+Y2-Y1+X2
980  RETURN
990  |
1000 DEF FNNorm(X,Y,Z)=SOR(X^2+Y^2+Z^2)
1010 |
1020 |
9999 |
    
```

**Programme sur
Hewlett Packard
9845 C**



J.B. Touchard (IREM P7 - SFRS)

Un « imagiciel » : L'AIRE D'UN TRIANGLE

(voir l'article de Serge Hocquenghem :
Enseignement des maths illustré par ordinateur, p. 23)

Programme sur Apple II

```

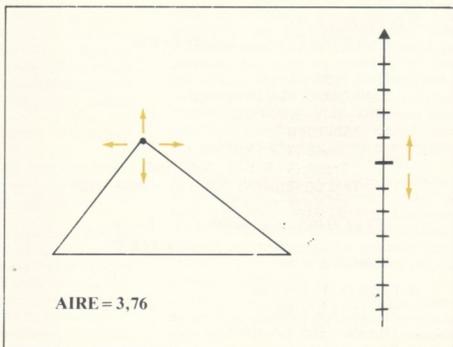
1  ONERR GOTO 3000
2  REM APPLE II E N & B
100 HOME : HGR
110 HCOLOR= 3
120 XA = 90:YA = 120
130 XB = 160:YB = 120
140 XM = 100:YM = 80
150 Y0 = 159
160 Q = .4
195 DI = 100
190 SH = 12
200 DL = 2
220 P5 = .5
230 HPLLOT XA,YA TO XB,YB TO XM,YM TO XA,YA
239 REM TRACE DE L'AIRE
240 HPLLOT 277,Y0 TO 277,0
242 X1 = 265:X2 = 276
243 FOR Y = Y0 TO 0 STEP - 12: HPLLOT 278,Y TO 279,Y:
NEXT
244 HPLLOT 275,2: HPLLOT 279,2: HPLLOT 276,1: HPLLOT 276,1
250 GOTO 390
299 REM FIN INITIALISATION
300 VTAB 1: GET A#
310 IF A# = "J" THEN DX = - DL:DY = 0: GOTO 380
320 IF A# = "I" THEN DX = 0:DY = - DL: GOTO 380
330 IF A# = "M" THEN DX = 0:DY = DL: GOTO 390
340 IF A# = "K" THEN DX = DL:DY = 0: GOTO 380
350 IF A# = "?" THEN GOSUB 1000
360 IF A# = CHR# (27) THEN 30000
370 GOTO 300
390 REM MODIFICATION DU DESSIN
390 HCOLOR= 0: HPLLOT XA,YA TO XB,YB TO XM,YM TO XA,YA
392 HPLLOT X1,Y1 TO X2,Y2
395 XM = XM + DX:YM = YM + DY
400 SD = ABS ((XM - XA) * (YM - YB) - (YM - YA) *
(XM - XB)) / 2:SD = INT (SD * Q) / DI
405 YS = Y0 - SD * SH
410 GOSUB 610
450 HCOLOR= 3: HPLLOT XA,YA TO XB,YB TO XM,YM TO XA,YA
455 HPLLOT X1,Y1 TO X2,Y2
460 GOTO 300
600 POKE - 16304,0: POKE - 16300,0
605 HOME
610 VTAB 22: PRINT "AIRE = ": STR# (SD) " "
620 RETURN
999 REM INSTRUCTIONS
1000 TEXT : HOME
1010 PRINT "AIRE DU TRIANGLE"
1020 PRINT "-----"
1030 PRINT
1040 PRINT
1050 PRINT
1060 PRINT "LE SOMMET MOBILE EST DEPLACE PAR LES"
1070 PRINT "TOUCHES I J K M"
1080 PRINT
1090 PRINT "LA VALEUR DE L'AIRE EST AFFICHEE EN"
1100 PRINT "BAS DE L'ECRAN."
1110 PRINT
1120 PRINT "ELLE EST REPRESENTEE SUR UNE DROITE "
1130 PRINT

```

```

1140 PRINT "A DROITE DE L'ECRAN"
1150 PRINT
1160 PRINT
1170 PRINT
1180 PRINT
1190 PRINT "ESC POUR SORTIR DU PROGRAMME."
1200 PRINT
1210 PRINT " C POUR QUELQUES COMMENTAIRES BREFS"
1220 PRINT
1230 PRINT " ESPACE POUR REVENIR AU DESSIN"
1240 GET U#: IF U# = "C" THEN 30000
1245 GOSUB 600
1250 RETURN
3000 RUN 100
30000 TEXT : HOME : END
50000 TEXT : HOME
50010 PRINT "AIRE D'UN TRIANGLE AVEC UN COTE FIXE"
50020 PRINT "-----"
50030 PRINT "<PROG. TRIAIRE>"
50040 PRINT
50050 PRINT "QUE REPRESENTE LA GRADUATION DE DROITE?"
50060 PRINT
50070 PRINT "DE QUOI L'AIRE EST-ELLE FONCTION?"
50080 PRINT
50090 PRINT "POURQUOI EST ELLE CONSTANTE SUR UNE"
50100 PRINT "HORIZONTALE? COMMENT VARIE-T-ELLE AVEC"
50110 PRINT "LA POSITION DU SOMMET MOBILE? QUEL EST"
50120 PRINT "LE LIEN AVEC LA VALEUR ABSOLUE?"
50130 PRINT
50135 PRINT " EN GEOMETRIE DANS L'ESPACE, QUELLE"
50140 PRINT "SURFACE OBTIENT-ON EN PORTANT L'AIRE"
50150 PRINT "SUR UNE PERPENDICULAIRE AU PLAN, AU"
50160 PRINT "DESSUS DU SOMMET MOBILE?"
50170 PRINT
50180 PRINT "ETC..."
50190 PRINT
50270 WAIT - 16304,123: POKE - 16369,0
50280 GOSUB 600: RETURN

```



Vos Programmes

Voici le programme du petit dessin animé géométrique dont il est question dans la leçon n° 9 : Structurer (encart p. 3/4).

- 1 (Présentation du menu)
(R=1 : Hauteurs, R=2 : Médiannes, R=3 : Médiatrices)
ENTRÉE R
- 2 **DONNÉES** PI=3,14 H=0,1
XB=-2 YB=-1 XC=3 YC=-1
- 3 **POUR I = PI/2 A PI/2 + 2PI PAS H FAIRE :**
XA = 4cos (I) YA = 4sin (I)
EXECUTER Triangle
SELON R EXECUTER Hauteurs, Médiannes,
Médiatrices. Attente : 5 secondes.
- 9 **FIN**
- 10 **Triangle**
EXECUTER
Transfo (XA, YA ; XAM, YAM)
Transfo (XB, YB ; XBM, YBM)
Transfo (XC, YC ; XCM, YCM)
TRACER SEGMENTS (XAM, YAM) - (XBM, YBM) -
(XCM, YCM) - (XAM, YAM)
Retour
- 20 **Hauteurs**
EXECUTER
Droite (XA, YA ; YC-YB ; XB-XC)
Droite (XB, YB ; YA-YC ; XC-XA)
Droite (XC, YC ; YB-YA, XA-XB)
Retour
- 30 **Médiannes**
EXECUTER
Droite (XA, YA ; 2XA-XB-XC, 2YA-YB-YC)
Droite (XB, YB ; 2XB-XC-XA, 2YB-YC-YA)
Droite (XC, YC ; 2XC-XA-XB, 2YC-YA-YB)
Retour
- 40 **Médiatrices**
EXECUTER
Droite (XB+XC, YB+YC ; YC-YB, XB-XC)
$$\frac{2}{2}$$

Droite $\frac{(XC+XA, YC+YA)}{2}$; YA-YC, XC-XA)
$$\frac{2}{2}$$

Droite $\frac{(XA+XB, YA+YB)}{2}$; YB-YC, XA-XB)
$$\frac{2}{2}$$

Retour
- 50 **Droite (P,Q ; U, V)**
N=SQR (U^2+V^2) E=U/2N F=V/2N
T=1
55: X = P Y = Q
TANT QUE (X-6,5) (X+6,5) < 0
ET (Y-5) (Y+5) < 0 **FAIRE :**
EXECUTER
Transfo (X, Y ; XM, YM)
Transfo (X+E, Y+F ; XME, YMF)
TRACER SEGMENT (XM, YM) - (XME, YMF)
X=X+2E Y=Y+2F
SI T=2 ALORS retour SINON T=2 E=-E
F=-F ALLER 55
Retour
- 60 **Transfo** (X, Y ; XM, YM)
XM=(X+6,5) * 640/13
YM=312 - (Y+5) * 312/10
Retour.

Version FRI



DROITES REMARQUABLES D'UN TRIANGLE

Version

Micral 8022G



```

1 HGR:TEXT:"MENU
2 PRINT"VOUS ALLEZ VOIR SOIT LES HAUTEURS"
3 PRINT"SOIT LES MEDIANES + SOIT LES MEDIATRICES"
4 PRINT"D'UN TRIANGLE VARIABLE"
5 PRINT
6 PRINT"QUE VOULEZ VOUS VOIR ?          FRAPPEZ:"
7 PRINT
8 PRINT*   1   POUR LES HAUTEURS"
9 PRINT*   2   POUR LES MEDIANES"
10 PRINT*   3   POUR LES MEDIATRICES"
15 INPUT R
19 .
20 "LES ANGLES SONT EN RADIAN
21 DATA 3,14+0,1
22 READ PI:H
25 "POINTS B ET C
26 DATA -2,-1+3,7-1
27 READ XB,YB,XC,YC
29 .
30 "DESSIN
40 FOR I=PI/2 TO 2*PI+PI/2 STEP H
50 XA=4*COS(I):YA=4* SIN(I)
51 HGR:TEXT
60 GOSUB 100:"DESSIN DU TRIANGLE
70 ON R GOSUB 200+300+400:"DESSIN
80 FOR K=1 TO 3000:NEXT K
90 NEXT I
99 END
100 "DESSIN DU TRIANGLE
110 X=XA:Y=YA:GOSUB 600:XAM=XM:YAM=YM
120 X=XB:Y=YB:GOSUB 600:XBM=XM:YBM=YM
130 X=XC:Y=YC:GOSUB 600:XCM=XM:YCM=YM
140 LINE (XAM,YAM)-(XBM,YBM):PSET
150 LINE (XBM,YBM)-(XCM,YCM):PSET
160 LINE (XCM,YCM)-(XAM,YAM):PSET
170 RETURN
200 "DESSIN DES 3 HAUTEURS
210 F=XA:Q=YA:U=YC-YB:V=XB-XC:GOSUB 500
220 F=XB:Q=YB:U=YA-YC:V=XC-XA:GOSUB 500
230 F=XC:Q=YC:U=YB-YA:V=XA-XB:GOSUB 500
299 RETURN
300 "DESSIN DES 3 MEDIANES
310 F=XA:Q=YA:U=2*XA-XB-XC:V=2*YA-YB-YC:GOSUB 500
320 F=XB:Q=YB:U=2*XB-XC-XA:V=2*YB-YC-YA:GOSUB 500
330 F=XC:Q=YC:U=2*XC-XA-XB:V=2*YC-YA-YB:GOSUB 500
399 RETURN
400 "DESSIN DES 3 MEDIATRICES
410 F=(XB+XC)/2:Q=(YB+YC)/2:U=YC-YB:V=XB-XC:GOSUB 500
420 F=(XC+XA)/2:Q=(YC+YA)/2:U=YA-YC:V=XC-XA:GOSUB 500
430 F=(XA+XB)/2:Q=(YA+YB)/2:U=YB-YA:V=XA-XB:GOSUB 500
499 RETURN
500 "DESSIN D'UNE DROITE
510 N=SQR(U^2+V^2):E=U/(2*N):F=V/(2*N)
520 T=1
550 X=P:Y=Q
560 GOSUB 600
565 IF XM<1 OR XM>638 OR YM<1 OR YM>310 THEN 590
570 XME=XM:YME=YM
575 X=X+E:Y=Y+F:GOSUB 600
579 IF XM<1 OR XM>638 OR YM<1 OR YM>310 THEN 590
580 LINE (XM,YM)-(XME,YME):PSET
585 X=X+E:Y=Y+F:GOTO 560
590 IF T=2 THEN 599
595 T=2:E=-E:F=-F:GOTO 550
599 RETURN
600 "TRANSFORMATION DES COORDONNEES
610 XM=(X+6,5)*640/13,3
620 YM=312-(Y+5)*312/10
699 RETURN
    
```

LA TORTUE MATERNELLE

Programmation par cartes perforées



Les objectifs pédagogiques qui ont conduit une équipe de l'I.N.R.P. (1) à réaliser une tortue de plancher programmable par de très jeunes enfants ont été décrits dans le numéro 10 d'E & I.

Cet article traite essentiellement des aspects techniques et logiciels mis en œuvre. En effet, bien que reprenant les idées de base de LOGO, il fallait, étant donné le public très particulier auquel s'adressait le système, définir un matériel et un langage spécifiques.

Principe du système, aspects matériels.

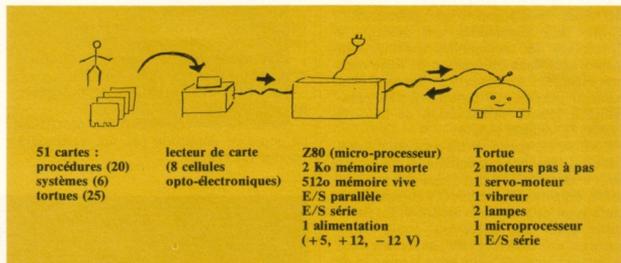
Le prototype développé à l'I.N.R.P. (2) comprend une tortue de plancher, un micro-ordinateur, un lecteur de carte et un jeu de cartes représentant les commandes (voir schéma).

La tortue est un terminal particulier dont la caractéristique principale est d'être mobile. Elle possède un certain nombre de fonctions propres gérées par un microprocesseur interne qui commande directement :

- deux moteurs pas à pas pour les déplacements et les rotations,
- un moteur pour lever ou descendre un stylo,
- un avertisseur sonore,
- deux voyants lumineux symbolisant les yeux.

Chaque fonction peut être déclenchée lorsqu'un code correspondant à cette fonction est envoyé à la tortue.

Ainsi, si la tortue reçoit par l'intermédiaire de son E/S série le caractère "G", ses



moteurs pas à pas tournent d'un pas de telle sorte que la tortue tourne d'un tiers de degré. Une rotation de 90 degrés nécessite donc que le micro-ordinateur envoie 270 fois la commande élémentaire "G".

La programmation du système se fait à l'aide de cartes perforées sur un côté selon un code à 8 positions. Ce code constitue un octet qui est détecté par les 8 cellules opto-électroniques équipant le lecteur de carte. Il est lu par le micro-ordinateur, conçu autour d'un microprocesseur 8 bits, par l'intermédiaire d'un circuit d'interface parallèle.

Un logiciel gravé dans une mémoire morte interprète la commande ainsi transmise et l'exécute éventuellement en pilotant en conséquence la tortue.

Bien que cela soit souvent confondu dans la description du système, il convient donc de bien distinguer la tortue qui n'effectue que des commandes élémentaires très simples (rotation de 1/3 de degré...) du micro-ordinateur qui exécute grâce à son logiciel des commandes plus complexes (rotation de 90 degrés...).

Le langage de commande

Le langage de programmation est essentiellement un langage de commande interprété. La programmation se fait, au niveau supérieur, en définissant des procédures. Une procédure est par exemple une suite d'actions pour que la tortue effectue un parcours déterminé tel qu'un carré. Une fois définie, une procédure a le même statut qu'une commande tortue de base telle que "tourne de 90 degrés". L'utilisateur peut donc ainsi enrichir le système de commandes nouvelles adaptées à ses besoins propres.

Il n'existe que trois types de commandes : les *procédures*, les *commandes tortue de base* et les *commandes système*. Les cartes qui les matérialisent ne possèdent volontairement, à l'origine, aucun symbole, mais ces trois catégories se différencient par des couleurs spécifiques.

— Les *cartes procédures*, de couleur jaune, ne sont pas connues du système lors de la mise sous tension. Nous verrons comment l'utilisateur peut les définir.

(1) I.N.R.P. Institut National de Recherche Pédagogique.

(2) Par P. Bastide, F.M. Blondel, J.C. Le Touzé, F. Robert.

— Les *commandes tortue de base* sont blanches. Elles sont relatives à un déplacement linéaire ou à une rotation de la tortue sur elle-même : avance ou recule de une ou deux longueurs préfixées (12 ou 24 cm soit le diamètre de la tortue), tourne à droite ou à gauche de 30, 45, 60 ou 90 degrés. Ces valeurs tiennent compte du fait que les enfants de 5-6 ans ne possèdent ni la notion de distance ni la notion d'angle et qu'elles permettent de tracer de façon visible la plupart des figures géométriques simples.

Klaxonne, lève ou descend le stylo, allume ou éteins les yeux, font aussi partie des commandes tortue de base.

L'expérimentation (3) nous a conduit à ajouter 8 autres cartes commandant le déplacement de la tortue selon des quarts de cercle pour permettre aux enfants le tracé de "ronds", de "fleurs"...

Chacune de ces commandes peut être exécutée immédiatement (mode "pas à pas") ou mémorisée pour une exécution ultérieure (mode "définition de procédure"). A l'initialisation le système est en mode "exécution immédiate".

— Les *commandes système* : au nombre de six, elles permettent de passer d'un mode à l'autre, de nommer des procédures, de corriger une ligne...

• **ÉCOUTE** (carte rouge) place le système en mode "définition d'une procédure". Toutes les commandes qui suivent, à l'exception des commandes système, sont mémorisées dans un tampon de ligne mais non exécutées. Le tampon de ligne qui peut contenir plus de 100 commandes est réinitialisé à chaque introduction de la carte ÉCOUTE dans le lecteur.

• **EXÉCUTE** (carte verte) provoque l'exécution de toutes les commandes contenues dans le tampon de ligne. Celui-ci étant conservé, on peut en redemander autant d'exécutions qu'on veut ou le modifier. En particulier après une exécution, toute carte, autre qu'une carte système, introduite dans le lecteur entraîne l'ajout dans le tampon de ligne de la commande correspondante.

• **CECI EST** (carte jaune) : cette carte doit toujours être suivie d'une carte procédure. Le code de cette dernière est alors mémorisé avec les commandes contenues dans le tampon de ligne dans une zone mémoire dite "zone des procédures définies" et il y a retour au mode "exécution immédiate". La procédure est alors connue du système et a même statut qu'une commande tortue de base. Une procédure peut faire appel, dans sa définition, à d'autres procédures préalablement définies. CECI EST permet donc de représenter une succession de commandes à l'aide d'une carte unique sur laquelle on peut dessiner la fonction ou le parcours exécuté par la tortue.

• **OUBLI** (jaune barré de noir) suivie d'une carte procédure, a pour effet de supprimer en mémoire la procédure correspondante.

Il y a réorganisation de la zone des procédures.

• **ENLÈVE** (marron) peut-être utilisée en mode "exécution immédiate" ou en mode "définition de procédure". Dans le premier cas elle commande à la tortue de revenir dans son état précédent en effectuant le mouvement inverse de celui qu'elle vient d'effectuer. Dans le second cas elle permet de supprimer la dernière commande du tampon de ligne.

• **STOP** (rouge barré de blanc) permet d'interrompre une exécution en cours.

Bien que simplifié, ce langage de commande possède les éléments de base pour programmer la tortue. De plus la tortue émet de brefs signaux sonores :

— pour signaler la prise en compte d'une commande,

— avant l'exécution de chaque commande du tampon de ligne. Dans ce cas il y a en plus un bref arrêt dans le mouvement de la tortue, de sorte que l'enfant qui a constitué un listage en disposant les unes à côté des autres les cartes utilisées, peut repérer la commande qui est exécutée. S'il s'agit d'une carte procédure, elle n'est signalée que par un seul arrêt et un message sonore, quel que soit le nombre de commandes qu'elle représente, puisqu'elle a même statut qu'une commande de base.

— Lors de la détection d'une erreur, il y a émission de plusieurs signaux sonores et lumineux. Ces erreurs sont :

— commande non connue ou utilisation d'une carte procédure non définie,
— tentative d'exécution d'une procédure qui a été détruite,
— tentative de définition d'une procédure en cours de définition par une carte déjà connue du système.



(3) Elle a eu lieu en 1981 et 1982 à l'école maternelle Paul Langevin à Villejuif avec des enfants de 5-6 ans.



Comment fonctionne le logiciel du micro-ordinateur

Il est composé de quatre parties principales : les tables, un noyau d'exécution, les blocs de reconnaissance de la commande et de communication avec l'extérieur.

— *Les tables* : à chaque type de commande correspond une table particulière dans laquelle on trouve les codes des cartes et des données nécessaires au traitement. Ainsi la table des "commandes tortue de base" contient outre les codes, les caractères à envoyer à la tortue pour qu'elle effectue un mouvement élémentaire ainsi que le nombre de fois qu'il faut l'envoyer.

— Le bloc "reconnaissance de la commande" recherche si la commande est connue du système. Il reçoit en entrée le code correspondant à la commande, explore les différentes tables ainsi qu'éventuellement la zone des procédures définies, et fournit en sortie soit un pointeur sur la commande trouvée, soit un message d'erreur en positionnant un indicateur.

— Le bloc "communication" gère les entrées-sorties. Le port parallèle auquel correspond le lecteur de carte est simplement lu. Le port série (communication avec la tortue) est programmé soit en sortie lors de

LA TORTUE MATERNELLE

l'envoi d'un caractère vers la tortue, soit en entrée pour écouter la réponse de cette dernière. Le protocole de communication entre le micro-ordinateur et la tortue consiste à contrôler que la tortue a bien reçu le caractère ou à attendre un acquiescement indiquant que l'ordre a été physiquement effectué.

— le noyau d'exécution fait appel aux blocs précédents ainsi qu'à différents modules d'exécution. Son principe est simple :

- si la commande est une commande système, il fait appel au module de traitement correspondant. Ainsi ÉCOUTE a pour effet de positionner l'indicateur de mode (mode "définition de procédure"). Nous avons déjà détaillé les fonctions réalisées par ces différentes commandes.
- dans les autres cas si la commande n'est pas reconnue il y a appel du module traitement d'erreur sinon la commande est mémorisée dans le tampon de ligne. L'examen de l'indicateur de mode précise s'il faut écouter la commande suivante ou exécuter celle qui vient d'être mémorisée.
- au cours de l'exécution proprement dite, un pointeur sur le tampon de ligne permet d'extraire la prochaine commande à effectuer. Dans le cas d'une procédure, on pointe sur le code correspondant dans la zone des procédures et les commandes qui la définissent constituent un nouveau tampon de ligne à exécuter...

Une, deux... cent tortues !

Ce premier prototype réalisé par l'I.N.R.P. a été à la base d'un nouveau développement. Une collaboration entre l'I.N.R.P. et le C.F.P.T. (4) de Cachan a permis la mise au point d'un système dont le principe est analogue à celui décrit ici mais dans lequel la tortue est radio-commandée. Une série d'une centaine d'exemplaires est actuellement réalisée pour les besoins d'expérimentation (5).

Cette recherche n'est qu'un moment de notre préoccupation permanente qui est de concevoir des environnements (matériels et logiciels) programmables par des débutants. Les objectifs qui sous-tendent ces travaux sont essentiellement : une initiation à l'informatique, une aide au raisonnement, la manipulation d'objets techniques.

● Jean-Claude Le Touzé
I.N.R.P.

(4) C.F.P.T. Centre de Formation des Professeurs Techniques.
(5) cf. E & I n° 13, p. 11.

Vous Avez Dit ?..



Après avoir reçu quelques coups en essayant de démontrer aux maîtres d'armes, de danse et de musique que l'informatique est la première de toutes les sciences et qu'on doit en faire une discipline obligatoire dans tous les établissements scolaires, le professeur d'informatique reste seul avec Monsieur Jourdain pour lui donner sa première leçon.

ACTE II-SCÈNE II

PROFESSEUR D'INFORMATIQUE
MONSIEUR JOURDAIN

PROFESSEUR D'INFORMATIQUE
- En reboutonnant son gilet - Venons à notre cours.

MONSIEUR JOURDAIN - Ah ! monsieur, je suis fâché des coups qu'ils vous ont donnés.

PROFESSEUR D'INFORMATIQUE
- Cela n'est rien. Un informaticien doit savoir résoudre tous les problèmes, et je vais écrire un beau logiciel qui les remplacera tous et les réduira au chômage. Laissons cela. Que voulez-vous apprendre ?

MONSIEUR JOURDAIN -
L'informatique, car c'est une science à la mode, et je voudrais savoir me servir du micro-ordinateur que j'ai installé dans mon salon.

PROFESSEUR D'INFORMATIQUE
- Ce désir est raisonnable. We live in a world of programming and keep programming most of the time, whether we realize it or not (1). Vous entendez cela, et vous savez l'anglais sans doute ?

(1) phrase extraite de la conférence du professeur Ershov, prononcée à Lausanne en juillet 1981, et qui nous prouve que notre informaticien a fréquenté le congrès WCCE (cf. E & I n° 8).

INFORMATIQUE...

pas indispensable et qu'on peut aussi fort bien faire de l'informatique sans ordinateur.

MONSIEUR JOURDAIN - Faire de l'informatique sans ordinateur ! Comment cela est-il possible ?

PROFESSEUR D'INFORMATIQUE
- Cela tient à la nature de l'information. L'information est la représentation d'une connaissance et non cette connaissance elle-même. Ainsi, quand vous écrivez votre nom sur une feuille de papier, les lettres qui composent votre nom constituent une information, et la connaissance associée à cette information est l'image d'un gentilhomme parisien élégant et instruit.

MONSIEUR JOURDAIN - Cela est vrai. Ah ! que n'ai-je étudié plus tôt pour savoir tout cela. Ainsi quand j'écris mon nom, je fais de l'informatique !

PROFESSEUR D'INFORMATIQUE
- Non ! pas encore. Vous utilisez seulement une information. Pour faire de l'informatique, il faut réaliser un traitement rationnel. Vous effectuez un traitement de ce genre lorsque vous écrivez un nom dans votre agenda. En effet vous comparez successivement chacune des lettres qui sont imprimées sur le côté droit des pages de l'agenda avec la première lettre du nom. S'il s'agit de la même lettre, vous ouvrez l'agenda et vous écrivez le nom sur l'une des pages qui portent cette lettre. Sinon vous passez à la lettre suivante. Quand vous procédez ainsi, vous exécutez ce qu'on appelle un algorithme, c'est-à-dire une suite d'opérations en nombre fini qui pourraient être automatisées et qui permettent de résoudre une classe de problèmes.

MONSIEUR JOURDAIN - Quoi ! quand je mets dans mon agenda le nom de la belle Dorimène, j'effectue un traitement de l'information ?

PROFESSEUR D'INFORMATIQUE
- Oui, monsieur.

MONSIEUR JOURDAIN - Par ma foi ! il y a plus de quarante ans que je fais de l'informatique sans que j'en suse rien ; et je vous suis le plus obligé du monde de m'avoir appris cela.

MONSIEUR JOURDAIN - Oui, mais faites comme si je l'ignorais. Dites-moi ce que cela signifie.

PROFESSEUR D'INFORMATIQUE
- Cela veut dire que nous vivons dans un monde où règne la programmation et par conséquent que nous devons tous apprendre l'informatique.

MONSIEUR JOURDAIN - Cet anglais-là a raison.

PROFESSEUR D'INFORMATIQUE
- Mais d'abord, savez-vous ce qu'est l'informatique ?

MONSIEUR JOURDAIN - oui, c'est la science des ordinateurs.

PROFESSEUR D'INFORMATIQUE
- Grossière erreur ! que commentent également les anglo-saxons. L'informatique n'est pas la science des ordinateurs, mais la science du traitement rationnel, notamment par machines automatiques, de l'informatique considérée comme le support des connaissances humaines, dans les domaines technique, économique et social. (2) Vous remarquerez que si l'on utilise souvent des machines automatiques, c'est-à-dire des ordinateurs, pour traiter des informations, cela n'est

(2) Définition de l'Académie Française.

FAUT-IL AVOIR PEUR DES DIDACTICIELS AMÉRICAINS ?

Voyons comment sont conçus, réalisés, diffusés, et utilisés les didacticiels opérationnels sur petits micro-ordinateurs.

Conception et réalisation

Quels que soient le sujet, le niveau et les institutions, ce sont des équipes constituées de quatre à cinq personnes qui élaborent le didacticiel. Prenons l'exemple du MECC (Minnesota Educational Computer Consortium) : l'équipe comprend un analyste, un "curriculum designer", deux programmeurs. L'analyste est en fait un pédagogue, qui définit les sujets de didacticiels à mettre en œuvre, après consultation et séances de travail auprès d'une population d'enseignants, et fixe le contenu du "courseware". Le "curriculum designer" est le responsable du produit tant sur le plan pédagogique que sur le plan informatique : il choisit la stratégie pédagogique, la composition et l'enchaînement des écrans, mais il assure également la direction du travail de programmation et la validation du produit final. Les programmeurs écrivent en

BASIC (Apple 2 et, depuis peu, ATARI). Remarquons cependant que le PASCAL tend à s'imposer partout.

Une telle équipe travaille environ six mois pour aboutir à un produit fini.

Diffusion

Dans le cas du MECC, un minimum de diffusion commerciale est assurée au départ puisque les sujets de didacticiels sont retenus en fonction de résultats d'enquêtes préalables auprès des enseignants. En revanche, le SMERC (San Matteo Educational Resource Center) à l'instar de CONDUIT, sélectionne, pour son catalogue de produits diffusés, les didacticiels proposés par des équipes indépendantes d'enseignants-programmeurs, essentiellement sur des critères de bon fonctionnement informatique.

Les produits sont diffusés sous forme de disquettes. (5 pouces) avec brochure d'accompagnement, pour un prix moyen de 250 francs.

A côté de cette diffusion spécifique, les constructeurs de micro-ordinateurs proposent aussi des produits à vocation pédagogique, souvent orientés vers les jeux et presque exclusivement graphiques.

Certains éditeurs commencent timidement à s'intéresser au marché de l'E.A.O., mais ne le jugent rentable que dans le secteur de la formation professionnelle (Mc GRAW HILL, WHILEY par exemple).

Utilisation

Un didacticiel standard, ou "courseware", tient en général sur une seule disquette 5 pouces (simple face, simple densité) et dure moins d'une heure. L'apprenant n'a pas à manipuler de disquettes en cours d'exécution du didacticiel. Les réponses (éléments de suivi pédagogique) ne sont pas enregistrées.

Le développement des micro-ordinateurs individuels suscite partout un rejet des systèmes centralisés, qui sont jugés trop chers et trop lourds à l'exploitation. (La clientèle de PLATO, système typique, n'est plus limitée qu'à quelques grandes entreprises.) Les constructeurs, pour leur part, souhaitent proposer chacun des produits d'enseignement qui restent très demandés. Sous cette pression, les produits pédagogiques doivent s'accommoder de sévères limitations matérielles et logicielles. De rares centres comme celui de A. BORK continuent à élaborer des didacticiels de haut niveau pédagogique, plus dans un esprit de recherche que de diffusion, mais ils ne reflètent pas les tendances actuelles de l'E.A.O. aux Etats-Unis.





Les sujets abordés et les stratégies pédagogiques sont très variés. On rencontre traditionnellement "jeux", "drill and practice" mais encore du contrôle de connaissances et de l'enseignement de concepts, et enfin la simulation.

Les jeux sont en fait un peu marginaux en ce sens que l'intérêt pédagogique n'est que sous-jacent : il s'agit plutôt d'un intérêt éducatif (recherche de stratégie gagnante : jeu sur les nombres premiers à TORONTO).

Les "drill and practice" constituent la grande majorité des didacticiels : les analyses de réponse sont rudimentaires et sont souvent incohérentes dans leur degré de tolérance au cours d'un même didacticiel. Le contrôle des connaissances peut aller jusqu'à l'examen automatique dans lequel l'ordinateur tire au hasard des exercices (École Polytechnique de MONTRÉAL).

L'enseignement de concepts nouveaux est abordé par très peu d'auteurs ; en général les didacticiels sont de qualité, mais au prix d'un gros effort de conception (innovation) et de programmation ; à noter que, dans ce dernier cas, le développement de quelques outils informatiques généraux est indispensable (Notation de vitesse et de durée : université d'IRVINE).

La simulation est pratiquée aussi bien par l'élève lui-même que par l'enseignant devant un groupe d'élèves : c'est en réalité un programme de pur calcul, le plus souvent, largement agrémenté de graphiques ou d'animations ; c'est à l'enseignant d'apporter les commentaires et explications d'ordre pédagogique (la nutrition dans les rivières : MECC).

Pourquoi cette situation ?

On peut s'étonner de la très grande disparité des réalisations et de leur caractère artisanal. Il faut sans doute penser à l'absence complète de coordination et même d'incitation officielle à développer un moyen de production et de communication qui permettent aux enseignants de travailler indépendamment des informaticiens. Dans ces conditions, personne ne cherche à créer un ensemble d'outils informatiques généraux propres à l'E.A.O., qui constituent une référence de portabilité et de communicabilité entre les utilisateurs, comme le projet DIANE cherche à le faire en France, sous l'égide de l'Agence de l'Informatique (1). Chacun est donc amené à programmer directement en BASIC ou PASCAL ses didacticiels (qui ne sont alors que des logiciels pédagogiques) et doit nécessairement limiter ses ambitions pédagogiques de telle sorte que la réalisation soit praticable et rentable.

Plus de peur que de mal

Les produits pédagogiques américains actuellement disponibles commercialement sont tous pratiquement de petits logiciels autonomes, correspondant pour la plupart à des activités de niveau élémentaire, à forte composante ludique, et faisant largement appel à des ressources graphiques rudimentaires. Leurs prix modiques et leurs aspects attrayants séduisent de prime abord, d'autant plus qu'ils rentrent dans les catalogue des logiciels proposés par les fournisseurs de matériels "vedette".

Toutefois la forte dépendance du produit pédagogique vis à vis du matériel rend difficile, voire impossible, son transfert vers un autre micro-ordinateur, et limite ainsi sa durée de vie. Ce dernier point est crucial puisque la micro-informatique évolue quotidiennement (percée des 16 bits et 32 bits). On doit remarquer que, si la présentation

est souvent élaborée (approche originale, illustration, animation...), la communication avec l'élève reste très limitée (pas de messages construits par l'élève, pas d'analyse réelle de ses difficultés pendant le déroulement du dialogue...), ce qui réduit beaucoup la portée pédagogique.

Dans la mesure où les enseignants européens attendent plus de l'interaction élève-machine, les produits d'outre-atlantique ne peuvent satisfaire les utilisateurs et inquiéter les auteurs. Ces derniers peuvent en fait espérer une large diffusion de leurs didacticiels pour peu qu'ils aient été conçus à l'aide d'outils informatiques assurant la portabilité.

Michel Barrier,
Université Paris VI
Michel Jacques,
Université Paris VII

(1) Voir E & I n° 13

VISEZ L'ANGLAIS AVEC SPEAKEASY

TEL QU'ON L'ECRIT

Les journaux «**Speakeasy**» (12 pages : niveaux moyen et avancé) et «**Easy Speakeasy**» (8 pages : niveau faux débutant) vous proposent l'anglais par l'actualité : *English through the news*. Les journaux sont accompagnés d'un «**Notebook**» comprenant les exercices auto-correctifs.

TEL QU'ON LE PARLE

Vous lisez l'anglais, vous le parlez, mais quand on vous parle, vous êtes perdu. Alors, écoutez les cassettes «**Speakeasy Broadcasts**» (60 mn, niveau avancé). Vous y trouverez des interviews, reportages et conversations avec des accents différents, pris sur le vif de Londres à Los Angeles.

J'aimerais recevoir une documentation sur les publications «**Speakeasy**».

J'aimerais m'abonner à **Speakeasy** : 5 numéros par an. . . . 38 F

J'aimerais m'abonner à **Easy Speakeasy** : 5 numéros par an. . . . 28 F

J'aimerais me procurer **Speakeasy Broadcasts** (avec livret d'exploitation et textes transcrits), la cassette. . . . 79 F

N° 1 N° 2

(N° 1 disponible en Janvier 82) prix spécial

pour 3 cassettes. . . . 200 F

Nous vous prions de faire des chèques séparés : un pour l'abonnement au journal, un pour les cassettes.

Ci-joint veuillez trouver la somme

de _____

Nom _____

Prénom _____

Adresse _____

Ville _____

Code Postal _____

A retourner à **Speakeasy**, Éditions Fernand Nathan,
9, rue Méchain 75675 Paris Cédex 14.



forme-1. PERFORM séquence-de-traitement { nom-de-donnée } TIMES,
entier-1.

forme-2. PERFORM séquence-de-traitement UNTIL expression relationnelle.

forme-3. PERFORM séquence-de-traitement
VARYING compteur FROM valeur-initiale BY pas-1
UNTIL expression relationnelle.

4.6. — Les entrées/sorties.

4.6.1. — *Lecture et écriture à partir de l'organe principal du système*
(lecteur de cartes et imprimante en traitement par lots, clavier et écran en mode conversationnel)

ACCEPT nom-de-donnée [FROM { DATE }
DAY }
TIME }]
DISPLAY { littéral-1 } [{ littéral-2 }
{ nom-de-donnée-2 }] [UPON { SYSSOUT }
{ CONSOLE }
{ SYSPUNCH }]

4.6.2. — Manipulation des fichiers.

4.6.2.1. — ouverture et fermeture

OPEN [INPUT nom-de-fichier-1 [,nom-de-fichier-2]...]
[OUTPUT nom-de-fichier 3 [,nom-de-fichier-4]...]
[INPUT-OUTPUT nom-de-fichier 5 [,nom-de-fichier-6]...]].
CLOSE nom-de-fichier-1, [,nom-de-fichier-2]...

4.6.2.2. — lecture

accès séquentiel
READ nom-de-fichier [INTO nom-de-donnée]
AT END instruction impérative.

accès direct

READ nom-de-fichier [INTO nom-de-donnée]
INVALID KEY instruction impérative.

4.6.2.3. — écriture

accès séquentiel
WRITE nom-d'enregistrement [FROM nom-de-donnée].

{ INVALID KEY instruction-impérative.
{ BEFORE } nom-de-donnée
{ AFTER } entier }
ADVANCING nom-de-donnée
entier }
LINES }

accès direct

WRITE nom-d'enregistrement [FROM nom-de-donnée]
INVALID KEY instruction-impérative.

4.6.2.4. — modification (accès direct)

REWRITE nom-d'enregistrement
INVALID KEY instruction-impérative.
4.6.2.5. — suppression (accès direct)
DELETE nom de fichier.

5 — APPEL DE SOUS PROGRAMME EXTERNE.

CALL nom de programme.

COBOL

d'un coup d'œil.

Pierre Bouchet et Jean Vicard

Les mots en majuscules sont des mots réservés, les mots en minuscules sont à écrire par le programmeur, les mots soulignés sont obligatoires, les parties entre crochets sont facultatives, lorsque plusieurs mots sont entre accolades, la présence de l'un d'entre eux est obligatoire. Dans les instructions, les phrases entre parenthèses sont considérées comme un commentaire.

PROGRAMME COBOL : 4 DIVISIONS :
IDENTIFICATION DIVISION ; ENVIRONMENT DIVISION ; DATA DIVISION ;
PROCEDURE DIVISION ;

1 — IDENTIFICATION DU PROGRAMME.

IDENTIFICATION DIVISION.
PROGRAM-ID. nom de programme.

2 — DÉFINITION DES DONNÉES : ELLES SONT DÉCRITES DANS LA DATA DIVISION.

Définition de la structure et pour les données non décomposables, de la structure, c'est-à-dire des données élémentaires :
définition : du type, de la plus grande valeur, du mode de représentation en mémoire de la valeur.

2.1. — Données élémentaires :

TYPE	REPRÉSENTATION en mémoire	nombre niveau	ÉCRITURE	type	représentation en mémoire
numérique	• BINAIRE	77	nom-de-donnée	<u>PIC</u> 9(n)	<u>COMP</u> .
	• DÉCIMAL CONDENSE (BCD)	77	nom-de-donnée	<u>PIC</u> 9(n)	<u>COMP-3</u>
	• DÉCIMAL DELATE	77	nom-de-donnée	<u>PIC</u> 9(n)	
9	• VIRGULE FLOTTANTE : simple précision	77	nom-de-donnée		<u>COMP-1</u> .
	double précision	77	nom-de-donnée		<u>COMP-2</u> .
alphabétique		77	nom-de-donnée	<u>PIC</u> A(n)	
alpha-numérique		77	nom-de-donnée	<u>PIC</u> X(n)	

Pliez selon le pointillé

77 : nombre niveau d'une donnée élémentaire isolée

n : nombre maximum de chiffres, lettres ou caractères de la valeur

Si donnée numérique décimale : PIC 9(n)Y(9/m)

(Position de la virgule virtuelle).

2.2. — Données structurées

2.2.1. — Type homogène (tableau, 3 dimensions maximum)

- 01 nom de tableau (définition de la donnée élémentaire)
- 02 nom d'élément OCCURS n PIC... (nombre d'éléments du tableau (à une dimension)).

2.2.2. — Type non homogène.

a) RECORD

- 01 nom-de-donnée structurée. (racine de l'arbre, niveau 0)
- 02 nom-de-donnée structurée. (nœud de l'arbre, niveau 1)
- 03 nom-de-donnée structurée. (nœud de l'arbre, niveau 2)

i nom-de-donnée élémentaire (feuille de l'arbre, niveau i-1) PIC... (description de la donnée élémentaire)

b) FICHER

Description des enregistrements, du facteur de blocage, des labels.

FILE SECTION.

FD nom de fichier

BLOCK CONTAINS n RECORD (si blocage, n définit le nombre d'enregistrements par blocs)

LABEL RECORD IS

{ OMITTED | STANDBARD } existence ou non des labels

DATA RECORD ARE nom d'enregistrement-1.... nom d'enregistrement-n) (désignation des différentes structures d'enregistrement)

- 01 nom d'enregistrement-1. } description de la structure 1
- 02 ... }

- 01 nom d'enregistrement-2. } description de la structure 2
- 02 ... }

description de la structure 2

3. — DESCRIPTION DU TYPE DE PÉRIPHÉRIQUE, DE L'ORGANISATION ET DU MODE D'ACCÈS DU FICHER.

ENVIRONNEMENT DIVISION. INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT nom de fichier ASSIGN TO nom de fichier système

ORGANIZATION { SEQUENTIAL } (organ. séquentielle)

{ INDEXED } (organ. séquentielle indexée)

{ DIRECT } (organ. directe)

ACCESS IS { SEQUENTIAL } (accès séquentiel)

{ RANDOM } (accès aléatoire)

RECORD KEY IS nom-de-donnée (définit la place de la clé dans l'enregistrement en séquentielle indexée)

SYMBOLIC KEY IS nom-de-donnée (nom de la clé d'accès au fichier si indexé ou direct)

ACTUAL KEY IS nom-de-donnée (nom de la variable dont les valeurs sont produites par l'algorithme de calcul des adresses de l'organisation directe)

4 — LES ACTIONS : Elles sont décrites dans la PROCEDURE DIVISION.

4.1. — Les opérateurs arithmétiques.

ADD

SUBTRACT

MULTIPLY

DIVIDE

COMPUTE (et les opérateurs + - * / **)

4.2. — L'affectation.

MOVE nom-de-donnée-1 TO nom-de-donnée-2 [nom-de-donnée-3]...
littéral

4.3. — Les expressions relationnelles simples.

IS [NOT] LESS THAN IS [NOT] ALPHABETIC

IS [NOT] GREATER THAN IS [NOT] NUMERIC

IS [NOT] EQUAL TO IS [NOT] = { ZERO | POSITIVE | NEGATIVE }

4.4. — Les expressions relationnelles composées.

Elles sont formées à l'aide des expressions relationnelles simples reliées par les opérateurs logiques :

NOT

AND

OR

4.5. — Les instructions de contrôle.

Une Séquence de traitement est composée d'un ou plusieurs paragraphes(s) et est identifiée de façon suivante :

nom-de-paragraphe-1 [THRU nom-de-paragraphe-n]

4.5.1. — Le branchement inconditionnel

GO TO nom-de-paragraphe.

4.5.2. — Le branchement avec retour

PERFORM séquence-de-traitement.

4.5.3. — La choix

IE expression relationnelle THEN { instruction-1 } NEXT SENTENCE

[ELSE { instruction-2 } NEXT SENTENCE]

* Pliez selon le pointillé *

VIFI NATHAN: ET L'ORDINATEUR DEVIENT INTELLIGENT.



VIFI NATHAN:
des dizaines de
jeux et de program-
mes éducatifs développés
par Nathan pour les enfants et
les adultes. Jeux de stratégie, de
logique, de déduction, de mémoire,
d'aventure; Microdidacts® de mathématiques,
sciences, informatique, langues, français, etc.
Les premiers programmes (compatibles avec Thomson TO 7) sont
disponibles dès novembre 1982 dans les librairies, les boutiques
"micro", chez les marchands de jeux et par correspondance.

Renseignements sur les points de vente VIFI :
VIFI NATHAN, 32, bd Saint-Germain, 75005 PARIS


NATHAN

marque déposée
Librairie Fernand Nathan

marque déposée
Thomson Brandt



L'INFORMATIQUE POUR TOUS

Nouvelle Édition

Nouveau

METHODES DE TRAVAIL
POUR L'INFORMATIQUE

Les guides
de l'informatique

JEAN-MICHEL TROUCHE

Le traitement de texte,
point de départ
de la bureautique

CEDIC

lire
LSE

Le langage français de programmation

manuel de
l'utilisateur

Christian Lafond
Pierre Muller
Préface de Jacques Hebenstreit



CEDIC/FERNAND NATHAN

cou
et exerc

1^{er} cycle univers
LUT, M.I.A.G.
et formatio
continue

cobol vol. 1

introduction
au langage cobol et à
la programmation
structurée

p. bouchet et j. vicard



CEDIC/FERNAND NATHAN

E et I n° 14

Lire Pascal _____ Codes
L'anglais pour informaticiens _____ Réf. 0310 60,50 F
Les microprocesseurs _____ Réf. 2031 149,00 F
Premières leçons de programmation _____ Réf. 0302 58,00 F
Enseignement et ordinateur _____ Réf. 0304 88,50 F
Premier livre de programmation _____ Réf. 0311 60,50 F
Initiation au Basic TO 7 _____ Réf. 0313 64,00 F
Lire L.S.E. _____ Réf. 0510 75,00 F
_____ Réf. 0300 60,50 F

Lire Basic _____ Codes
Les ordinateurs _____ Réf. 0303 60,50 F
Dictionnaire de micro-informatique _____ Réf. 0301 71,70 F
et micro-électronique _____ Réf. 0307 145,00 F
COBOL vol. 1 _____ Réf. 0308 70,00 F
Sachez choisir votre micro-
ordinateur de gestion _____ Réf. 2035 149,00 F
Le traitement de texte,
point de départ de la bureautique _____ Réf. 2034 149,00 F

Nom _____ Prénom _____

Adresse _____

Code postal _____

Enseignant Entreprise Particulier Montant _____ F

Frais d'envoi _____ 5,00 F

désire recevoir le catalogue Total _____ F

désire recevoir les ouvrages cochés Date _____

Ci-joint :

chèque bancaire postal mandat Signature _____

EDITIONS

CEDIC

Veuillez présenter ce bon à votre libraire ou à défaut nous le retourner.

32, bd Saint-Germain 75005 Paris
Tél. : 1.326.42.71