

MICRO-ORDINATEUR ET ENSEIGNEMENT DES MATHÉMATIQUES :

UN ENSEIGNEMENT EXPERIMENTAL D'INTRODUCTION AU BASIC A HALLE (RDA)

Lothar FLADE
Manfred PRUZINA
Section de Mathématiques
de l'Université Martin Luther de Halle-Wittenberg (RDA)

I - REMARQUES PRELIMINAIRES.

Une des caractéristiques de la révolution scientifique et technique réside en grande partie dans le développement explosif et l'extension croissante de l'utilisation de l'ordinateur grâce à l'évolution de la micro-électronique. On peut maintenant faire exécuter par des machines des processus mentaux qui peuvent être décrits par des algorithmes ; il en découle un nouveau type de productivité qui permet d'effectuer des tâches intellectuelles jusque là réservées aux humains.

On peut penser que dans les années à venir les micro-ordinateurs seront toujours en plus grand nombre dans les lieux de travail. L'intérêt social du développement des connaissances fondamentales en informatique chez les personnes actives, qui leur permettraient d'utiliser au mieux l'ordinateur dans leur travail n'est plus à démontrer.

La fondation en 1985 de la société pour l'informatique en RDA témoigne de la reconnaissance sociale de l'informatique dans notre pays. Notre jeunesse se passionne actuellement pour l'informatique. Une enquête que nous avons réalisée en 1985 auprès de 70 jeunes révèle que 80 % d'entre eux que ce soit à l'école ou dans les entreprises aimerait apprendre comment fonctionne un ordinateur et comment l'on s'en sert. L'école et en particulier l'enseignement des mathématiques ne peuvent ignorer ces demandes.

Il n'est d'ailleurs pas étonnant que les conséquences de l'extension de la micro-informatique pour l'école soient discutées dans tous les pays industrialisés et que des

(1) NdT : Lors d'une mission en République démocratique allemande dans le groupe de méthodique des mathématiques de l'université Martin Luther à Halle, j'ai pu constater que nos collègues allemands s'intéressaient à l'introduction de l'informatique dans l'enseignement obligatoire. Ils avaient mené des enseignements expérimentaux d'initiation à la programmation en BASIC sur des micro-ordinateurs Robotron (8 bits) fabriqués dans leur pays. Il m'a paru intéressant de traduire en français le rapport d'expérimentation qu'ils avaient publié en RDA à la fois pour la présentation du contenu de l'enseignement dispensé et pour les données recueillies sur les élèves confrontés à cet enseignement.

Colette LABORDE

recherches soient menées à propos de l'utilisation de l'informatique à l'école. On peut distinguer les directions de recherche suivantes :

- l'ordinateur en tant qu'objet d'enseignement (les élèves sont initiés à l'utilisation d'un ordinateur, ils apprennent à en connaître les constituants fondamentaux et apprennent un langage de programmation ou des éléments d'un langage de programmation) ;

- l'ordinateur en tant que moyen d'enseignement

- a) utilisé par l'enseignant comme outil de démonstration (l'ordinateur a dans ce cas une fonction de médiation des connaissances),

- b) utilisé par l'enseignant et les élèves en tant qu'outil de solution de problèmes (numériques ou non),

- c) utilisé par l'enseignant en tant qu'aide à l'enseignement et par les élèves en tant qu'aide à l'apprentissage (enseignement assisté par ordinateur).

Quelques pays ont déjà pris des décisions relatives à l'utilisation d'ordinateurs à l'école. Parmi ceux que nous connaissons bien, citons **l'Union soviétique** où depuis 1985-1986 l'informatique est enseignée comme une discipline à part entière en 9^{ème} classe⁽²⁾ (Programme du cours d'informatique, 1985). Chaque lycée **en Hongrie** est équipé depuis 1983 d'au moins un micro-ordinateur pour les besoins de la formation continue des enseignants et de clubs d'élèves.

En **RDA** depuis 1985-1986 un enseignement obligatoire de deux ans d'"informatique mathématique" a lieu dans les écoles spécialisées à orientation scientifique et technique pour les classes 9 et 10⁽³⁾ (Simonics, 1984). Depuis 1986, la discipline "Fondements de l'automatisation" est introduite dans le cadre de la formation professionnelle théorique dans toutes les branches professionnelles (enseignement d'environ 108 heures). Les élèves y apprennent des langages de programmation au sein de la sous-unité "utilisation de l'ordinateur dans les processus d'automatisation" (Programme pour la formation des travailleurs spécialisés, 1986) p... 36). D'autre part pour les classes 9, 10, 11, et 12⁽⁴⁾ ont lieu actuellement des enseignements expérimentaux facultatifs d'informatique.

C'est la raison pour laquelle une première recherche pratique et théorique s'est développée en différents lieux de recherche de notre pays ; elle a trait aux possibilités et aux problèmes suscités par l'emploi de micro-ordinateurs en tant qu'outil et objet de formation dans la scolarité obligatoire.

Utiliser des micro-ordinateurs en tant qu'instrument de solution de problèmes mathématiques suppose que les élèves sachent se servir d'un micro-ordinateur, maîtriser et utiliser certains éléments d'un langage de programmation.

Dans une première phase exploratoire nous avons cherché à savoir comment ces différentes compétences pouvaient être acquises par les élèves d'âges divers en concevant un enseignement de 10 heures d'introduction au langage BASIC pendant l'année scolaire 1984-1985 et en le réalisant avec des élèves de 15/16 ans et de 16/17 ans dans le cadre d'"ateliers mathématiques facultatifs.

(2) NdT : équivalente de la classe de 3^{ème}.

(3) NdT : équivalentes des classes de 3^{ème} et 2^{ème}.

(4) NdT : équivalentes des classes de 3^{ème}, 2^{ème}, 1^{ère} et terminales.

II - OBJECTIFS ET DISPOSITIF DE LA RECHERCHE.

Les questions suivantes sont au cœur de notre recherche expérimentale :

- 1) dans quelle mesure et à quel moment les élèves en contact direct avec un micro-ordinateur s'approprient-ils les éléments du langage de programmation BASIC, pour
 - comprendre des programmes simples en BASIC (linéaires, avec branchements, ou avec boucles),
 - réaliser des programmes BASIC pour résoudre des problèmes numériques simples ?

C'est pourquoi nous avons d'abord cherché à répondre à la question suivante :

- 2) Quelles sont les conduites identifiables des élèves travaillant sur et avec un micro-ordinateur ?

La recherche a été menée dans le cadre d'un atelier mathématique auquel la participation était volontaire et non sanctionnée.

Le premier groupe se composait de 24 élèves des classes 11 et 12 de deux "écoles secondaires élargies" de la ville de Halle⁽⁵⁾ et le deuxième groupe se composait d'élèves des classes 9 et 10 d'une école secondaire polytechnique classique comportant dix niveaux scolaires⁽⁶⁾ de la Ville Neuve de Halle. Alors que l'enseignement s'est étalé sur deux semaines pour le premier groupe, il a été concentré sur une semaine (deux heures par jour pendant cinq jours) pour le deuxième. Les cours avaient lieu dans une salle réservée aux micro-ordinateurs, les élèves travaillaient à deux ou trois par ordinateur. Les ordinateurs disposaient d'une version étendue de BASIC.

III - CONCEPTION DE L'ENSEIGNEMENT "INTRODUCTION AU LANGAGE BASIC" (10 heures).

Notre enseignement avait pour objectif de transmettre des connaissances fondamentales d'utilisation d'un micro-ordinateur et d'enseigner des éléments de programmation en BASIC. On devait pour cela s'appuyer sur des contenus mathématiques simples et déjà connus : par exemple, on pouvait demander aux élèves de fabriquer des programmes

- de calcul de l'aire et de la circonférence d'un disque de rayon donné,
- de constitution de tableaux (par exemple des valeurs prises par une fonction),
- de résolution d'équations du deuxième degré,
- du calcul de tous les diviseurs d'un nombre entier.

On a pu faire en sorte que les élèves travaillent à l'ordinateur dès la première heure d'enseignement, pour appliquer aussitôt les éléments de programmation qui leur étaient enseignés. On satisfaisait aussi les attentes des élèves qui désiraient travailler le plus vite possible à l'ordinateur.

Dans cet enseignement de 10 heures il n'était pas possible pour des raisons de temps de développer le concept théorique d'algorithme et d'introduire des notations du type organigramme ou diagramme structuré.

(5) NdT : comparables aux lycées français.

(6) NdT : c'est-à-dire pour des élèves de 6 à 16 ans relevant de la scolarité obligatoire.

Nous considérons la maîtrise d'éléments de programmation et les savoir-faire impliqués dans l'usage d'un micro ordinateur comme une condition raisonnable pour parler d'algorithmes aux élèves (Knauer).

Dans la suite nous esquissons le contenu de l'enseignement :

1) Introduction à l'utilisation d'un micro ordinateur, du clavier, des commandes de la machine.

Prise de contact avec la place de travail à la machine, mise en marche, familiarisation avec le clavier en particulier avec les touches SHIFT et ENTER, la barre d'espace, et les curseurs, instructions de commande de l'ordinateur et de l'écran : CLS, RUN, STOP, LIST, RESET, WBASIC, NEW.

2) Eléments de programmation en BASIC, exercices de programmation.

Infrastructure d'un programme en BASIC : lignes du programme, numéros de ligne, instructions.

Présentation d'expressions mathématiques en BASIC : opérations (+, -, *, /), parenthèses, variable, représentation des nombres, fonctions SQR(X), PI, ABS(X)⁽⁷⁾ INT(X)⁽⁷⁾.

Règles de priorité entre expressions, instructions INPUT, LET, PRINT, END, IF, THEN..., FOR..., NEXT⁽⁷⁾, GOTO.

Les élèves n'ont pas été familiarisés avec d'autres instructions et d'autres possibilités de programmation durant ces dix heures comme par exemple le mode direct ou le mode EDIT. Pour corriger leurs programmes, les élèves utilisaient l'effacement, l'insertion, et le report de lignes.

Les éléments de programmation cités plus haut permettent de réaliser des structures typiques de programme (linéaires, avec branchement, avec boucles). Ces éléments suffisent à la programmation de problèmes numériques simples dans le cadre de la transmission de savoirs et savoir-faire mathématiques. Le nombre des fonctions standard introduites est resté très restreint, parce que des ajouts n'auraient pas apporté un point de vue qualitativement différent sur la programmation.

On a exigé dès le début de l'introduction à la programmation une présentation des programmes qui respecte leur aspect à l'écran (édition du texte, disposition claire, soulignement), sans entrer explicitement dans le fonctionnement des chaînes et fonctions chaînes. A titre d'exemple citons un des premiers programmes élaboré avec les élèves tel qu'il se présentait (on avait renoncé à l'indication d'unités) :

```

1Ø CLS
2Ø PRINT "VOLUMEN EINER KUGEL" (PRINT "Volume d'une sphère")
3Ø INPUT "Gib Radius ein" ; R (INPUT "Donne le rayon")
4Ø LEV V = (4/3) * PI * R * R * R
5Ø PRINT "Volumen" ; V (PRINT "Volume")
6Ø END

```

L'utilisation par les élèves d'éléments du langage de programmation était toujours rattachée à la résolution d'un problème. Le point de départ consistait en la résolution du problème et en la description verbale des étapes de la solution. Ce n'est qu'ensuite que

(7) Ces instructions n'ont été introduites qu'aux élèves du niveau du baccalauréat.

de nouveaux éléments de langage étaient introduits et les réactions de l'ordinateur explicitées. Ainsi pour introduire l'instruction conditionnelle IF... THEN... on a posé d'abord le problème du programme de calcul de la surface et du périmètre d'un triangle rectangle dont l'hypoténuse et un côté de l'angle droit étaient donnés. Le programme était testé par l'affectation de valeurs irrecevables aux données d'entrée (valeurs négatives). On voulait provoquer les réactions de l'ordinateur à de tels cas pour conduire aux formulations suivantes

Si la longueur de l'hypoténuse est inférieure ou égale au côté de l'angle droit, alors imprimer "un tel triangle n'existe pas" et arrêter le travail.

Si les longueurs de l'hypoténuse et des côtés de l'angle droit ne sont pas positives, alors imprimer "les données sont fausses" et changer les données d'entrée.

Venait ensuite l'écriture en BASIC et le programme ainsi obtenu pouvait être testé et l'effet de l'instruction IF... THEN montré.

On a donné une place importante dans le processus d'enseignement au fonctionnement sans ordinateur de programmes déjà écrits ("test à froid") et aux exercices de correction de programmes BASIC donnés. Donnons quelques exemples :

1) Quelles fautes présente le programme ? Corrigez le programme et trouver la valeur de A imprimée.

```
1Ø LET A = 5
2Ø LET 7 = B
3Ø LET A = 5 * A - B
4Ø PRINT A
15 END
```

2) Faites tourner le programme suivant avec les valeurs indiquées et donnez les valeurs imprimées

```
a) 1Ø INPUT A, B
2Ø LET C = A * B
3Ø LET A = B * C
4Ø LET B = C * A
5Ø PRINT A, B, C
```

valeurs d'entrée	A	3	-1	-2
	B	2	-2	1

```
b) 1Ø INPUT N
2Ø LET F = 1
3Ø LET F = F * N
4Ø LET N = N - 1
5Ø IF N > 0 THEN 3Ø
6Ø PRINT F, N
```

valeurs d'entrée pour N : 4 et 6

```
c) 1Ø FOR A = 4 TO -1 STEP -1
2Ø LET B = INT (A/2)
3Ø PRINT A, B
4Ø NEXT A
```

Ces programmes construits servaient avant tout à la consolidation de l'instruction LET.

Les élèves du niveau du baccalauréat avaient des devoirs à faire à la maison. Ils ont eu aussi à

- calculer la surface extérieure d'un cylindre circulaire
- résoudre l'inéquation du type $ax + b > c$

- calculer des termes d'une suite de nombres et leurs sommes partielles.

On a donné à la disposition des élèves un aide-mémoire contenant un résumé d'éléments du langage BASIC et de commandes du micro-ordinateur et de l'écran. Les 10 heures d'enseignement pouvaient être utilisées très intensivement pour permettre aux élèves de s'approprier de façon active des savoirs et savoir-faire.

IV - QUELQUES RESULTATS DE LA RECHERCHE.

4.1 Compréhension d'éléments de programmation en BASIC.

Pour vérifier et développer les savoirs et savoir-faire des élèves on leur a proposé pour l'essentiel deux types de tâches :

- faire fonctionner des programmes sans ordinateur
- construire des programmes pour résoudre des problèmes numériques relativement simples.

Avant d'exposer les résultats des élèves aux devoirs de contrôle de fin d'enseignement, présentons les problèmes de méthode et les erreurs typiques qui sont apparus au cours de l'enseignement. Nous omettons les erreurs dues à des connaissances mathématiques insuffisantes.

(Par exemple, $V_{\text{sphère}} = \pi r^3$ au lieu de $V_{\text{sphère}} = 4/3 \pi r^3$).

Quand nous avons conçu l'enseignement nous nous attendions déjà à ce que le signe d'égalité dans l'instruction LET présentât une difficulté de par sa signification différente de celle qu'il a habituellement en mathématiques.

La signification que les élèves ont accordée à l'instruction LET a connu pour l'essentiel l'évolution suivante.

Lors de la première séance, l'instruction LET a été employée dans le programme de calcul du volume d'une sphère (cf. plus haut).

4Ø LET V = (4/3) * PI * R * R * R

et le caractère particulier de l'égalité a été souligné.

De nombreux élèves ont réussi alors à écrire d'autres programmes linéaires de calcul plus simples. Remarquons cependant que le signe d'égalité est employé comme habituellement en mathématiques dans des programmes du type :

LET V = A * A * A

LET Ø = 4 * A * A

LET Ø = 2 * A * 2 * B

L'exercice 1) n'était destiné qu'à permettre une compréhension plus approfondie du sens du signe "=" dans l'instruction LET. Malgré un enseignement répété les élèves n'ont pas cessé de connaître ensuite encore des difficultés dans le maniement de cette instruction, comme le montrent les lignes suivantes du programme erroné :

LET A + 1 = A

LET (2 * x) + (3 * y) = 57

Encore un autre exemple :

Les élèves devaient écrire un programme de résolution de l'inéquation

$$ax + b > c.$$

La plupart d'entre eux ont distingué, il est vrai, deux cas suivant la valeur de a, mais ils ont eu des difficultés avec l'affectation.

Le programme suivant montre une erreur typique :

```

1Ø CLS
2Ø PRINT "résolution de a * x + b > c"
3Ø INPUT "a =" ; A
4Ø INPUT "b =" ; B
5Ø INPUT "c =" ; C
6Ø IF A = Ø THEN PRINT "Si b > c, alors n'importe quel x est solution" :
PRINT "Si b < c, alors il n'y a pas de solution" : END
7Ø IF A > Ø THEN LET x > (C - B) / A
8Ø IF A < Ø THEN LET x > (C - B) / A
9Ø IF A > Ø THEN PRINT "x >" ; x
1ØØ IF A < Ø THEN PRINT "x <" ; x
11Ø END

```

De nombreux élèves ont corrigé d'eux mêmes leur programme en se souvenant de la structure générale et de la signification de l'instruction LET. Les progrès dans la compréhension et l'utilisation de cette instruction ont surtout été apparents dans l'exercice suivant.

3) Fabriquez un programme qui calcule les 20 premiers termes de la suite (a_n) avec $a_n = n + 1/n$ et les sommes partielles correspondantes.

La plupart des élèves ont reconnu que l'égalité récursive $S_{n+1} = S_n + a_{n+1}$ pouvait être exprimée par l'affectation LET $S = S + A$.

La nécessité de donner une valeur initiale à S (ici $S = Ø$) a été alors reconnue. Le travail ultérieur a montré cependant qu'on devait accorder encore davantage d'attention à l'entrée des valeurs initiales.

La boucle FOR-NEXT a constitué une autre source d'erreurs. Comme le montre l'exemple suivant, c'est surtout dans la programmation du corps d'itération que se sont produites les erreurs.

Il s'agissait de faire un programme en BASIC qui imprime un tableau des valeurs de la fonction $y = 2x^2 - 5$ dans l'intervalle $-3 < x < 3$ (pas : 0,5).

Donnons ici des extraits d'un premier essai de quelques élèves :

Tino 3Ø FOR X = -3 TO 3 STEP 5
 4Ø LET Y = 2 * X * X - 5
 5Ø NEXT X
 6Ø PRINT X, Y : END

Steffen 4Ø FOR -3 TO 3 STEP 5
 5Ø LET X = -3
 6Ø LET Y = 2 * X * X - 5
 7Ø PRINT X, Y
 8Ø NEXT X

Cornelia 3Ø LET Y = 2 * X * X - 5
 4Ø FOR X = -3 TO 3 STEP 0.5
 5Ø PRINT X
 6Ø PRINT Y
 7Ø NEXT

Après avoir entré les programmes à l'ordinateur, la plupart des élèves ont pu corriger d'eux mêmes leurs erreurs.

La mise en page de l'écran (texte, soulignement, lignes vides) a pu être pratiquée sans problème simplement grâce à des exemples montrés par l'enseignant sans qu'il ait à introduire explicitement des instructions particulières ou les chaînes de caractères.

En outre il a été possible de rendre visible le traitement séquentiel du programme par le micro-ordinateur. Ainsi certains élèves ont-ils trouvé particulièrement difficile de numéroter les lignes du programme lorsqu'ils avaient à demander l'impression de lignes vides. Plusieurs d'entre eux au début ont confondu formellement les lignes de programme avec les instructions d'impression de lignes vides. La question de la place dans le programme de l'impression des têtes de colonnes du tableau des valeurs d'une fonction calculées par une boucle FOR...NEXT a aussi nécessité une analyse minutieuse du problème de la part des élèves.

De même pour de nombreux élèves ce qui apparaissait sur l'écran était identique au contenu des mémoires. Par exemple lors de la correction des programmes, la ligne concernée était certes corrigée à l'écran mais l'entrée en machine de cette correction par ENTER était souvent oubliée. On doit accorder davantage d'attention à la nécessité de la prise de conscience de la distinction entre l'intérieur et l'extérieur. Laborde (1985) propose de faire tourner des programmes à l'aide d'une feuille de simulation⁽⁸⁾ :

| numéro de ligne | valeur des variables | trace sur l'écran | commentaire |

Des différences de rythme de travail et d'apprentissage entre groupes d'élèves sont apparues évidemment comme dans les disciplines obligatoires. En particulier il nous a semblé au début du cours que des élèves passaient relativement beaucoup de temps à frapper des programmes courts et que les différences entre groupes étaient alors assez importantes.

C'est pourquoi nous avons décidé d'avoir des actions différenciées. Lors de la séance introductive quelques groupes ont fait du travail supplémentaire. Nous avons alors demandé à quelques élèves de rendre les programmes plus accessibles aux usagers en utilisant du texte, du graphisme...

A la fin du processus d'enseignement nous avons proposé aux élèves les exercices de contrôle suivants à travailler seul.

4) Faites tourner le programme BASIC suivant avec les valeurs d'entrée indiquées. Notez dans chaque cas les valeurs de sortie.

- | | |
|--------------------------|--------------------------------|
| a) 1Ø INPUT A | |
| 2Ø LET B = 1.5 * A - 1/A | Valeur d'entrée A = 2 |
| 3Ø PRINT B | Valeur de sortie B = |
| 4Ø END | |
| b) 1Ø INPUT N, B | Valeurs d'entrée N = 11, B = 9 |
| 2Ø LET N = N + 2 | Valeurs de sortie N = |
| 3Ø LET C = SQR (B) | C = |
| 4Ø LET B = B * B | B = |
| 5Ø PRINT N, C, B | |
| 6Ø END | |

(8) NdT : Il s'agit d'une feuille dans laquelle les élèves doivent indiquer pour chaque ligne de programme l'état des variables, et ce qui apparaît à l'écran lors de l'exécution du programme.

- c) 1Ø INPUT M,C Valeurs d'entrée M = 5, C = 64
 2Ø LET L = 1 L
 3Ø LET C = (C/2) * L C
 4Ø LET L = L + 1 Valeurs de sortie L =
 5Ø IF L < M THEN 3Ø C =
 6Ø PRINT L, C
 7Ø END
- d) 1Ø LET B = Ø K
 2Ø FOR K = 19 TO 21 A
 3Ø LET A = INT (K/1Ø) B
 4Ø LET B = B + A Valeur de sortie B =
 5Ø NEXT K
 6Ø PRINT B
 7Ø END

5) Ecrire un programme BASIC de calcul de la surface intérieure d'une couronne circulaire de diamètre d_1 et d_2 donnés.

(On rappelle que l'aire A est donnée par $A = \pi/4 (d_1^2 - d_2^2)$ pour $d_1 > d_2$).

6) Ecrire un programme BASIC qui calcule le produit $p = 2.4.6...n$ pour n'importe quel entier n pair.

Ci-dessous le pourcentage de réponses exactes pour l'exercice 4

Exercice	4a	4b	4c	4d ⁽⁹⁾
Elèves "lycées"	100	98	60	65
Elèves "scolarité obligatoire"	75	88	40	

Les résultats (en particulier ceux des exercices 4a et 4b) montrent que
 - les élèves savaient faire tourner un programme BASIC et qu'ils comprenaient des instructions BASIC simples ;
 - en particulier les instructions LET, dans lesquelles étaient affectées de nouvelles valeurs à des variables, étaient traitées correctement.

En revanche les pourcentages chutent pour les programmes c) et d) avec boucle. Les raisons ne tiennent pas fondamentalement à une mauvaise compréhension des instructions IF... THEN, resp. FOR NEXT, mais à un manque de concentration ou à des fautes d'inattention. En témoigne le fait que dans le tableau de valeurs de l'exercice 4c qui devait comporter 4 colonnes, les élèves "lycées" ont rempli correctement en moyenne 3,6 colonnes et les élèves "scolarité obligatoire" environ deux colonnes.

Une enquête complémentaire auprès du groupe des élèves "scolarité obligatoire" a montré qu'il avait compris correctement l'instruction IF... THEN, car tous les élèves de ce groupe ont résolu sans erreur les exercices suivants.

- 7) Faites tourner le programme BASIC suivant avec les valeurs d'entrée
 $A = 5, B = 3$

et indiquez les valeurs de sortie.

- 1Ø INPUT A, B
 2Ø LET C = 2
 3Ø LET B = 2 * B + B B
 4Ø LET C = C + 1 C

(9) L'exercice 4d n'a pas été posé aux élèves "scolarité obligatoire".

```

5Ø PRINT B, C           Valeurs de sortie B =
6Ø IF C < A THEN 3Ø     C =
7Ø END

```

Plus de 65% des élèves "scolarité obligatoire" et plus de 80% des élèves "lycée" ont maîtrisé la sémantique de l'exercice 5). Entre autres fautes de syntaxe sont apparues :

π au lieu de PI, 0 au lieu de Ø, PI/(4...) au lieu de PI/4 * (...).

Les élèves auraient sûrement corrigé d'eux-mêmes ces fautes en faisant tourner le programme sur l'ordinateur. Il est remarquable que presque tous les élèves du niveau "lycée" ont tenu compte de l'impossibilité de certaines valeurs d'entrée ($d_1 < = d_2$, resp. $d_1 - d_2$ non positif) par une instruction IF...THEN bien que l'énoncé de l'exercice ne le précisât pas.

La plupart des élèves du niveau "lycée" et quelques élèves "scolarité obligatoire" n'ont pas seulement programmé le calcul numérique mais ont inséré dans leur programme l'impression d'éléments textuels (c'est-à-dire des titres, des indications d'entrée ou de sortie de variables).

Les élèves "lycée" ont résolu en complément l'exercice 6) qui devait selon nos attentes soulever davantage de difficultés.

47% des élèves ont proposé un programme correct avec l'emploi de la boucle FOR...NEXT. Seul un élève a construit un branchement pour écarter les valeurs impaires ou négatives de n. Les erreurs typiques ont été :

- aucune valeur d'entrée pour p ;
- instruction LET P = 2 * 4 * 6 * ... * K dans le corps de la boucle ou boucle

proposée :

```

FOR X = 2 TO N STEP 2
LET P = X * NEXT X

```

Davantage d'exercices de ce type sont visiblement nécessaires pour permettre en particulier la reconnaissance du corps de boucle, l'affectation de valeurs d'entrée ou la modification de valeur des variables.

En bref nous pouvons constater que les objectifs essentiels de notre enseignement de BASIC en 10 heures ont été atteints. Cependant nous trouverions raisonnable de le prolonger pour permettre une consolidation des éléments de programmation en BASIC et un maniement plus assuré du micro-ordinateur.

4.2 Comportements et opinions des élèves.

Des aspects importants de l'évaluation du contenu et de la conception didactique de notre enseignement sont fournis par les opinions exprimées par les élèves, par leurs conduites, qui ont pu être observées, ainsi que par leur présence régulière. En plus d'observations et d'entretiens avec les élèves, nous avons procédé à une enquête écrite sur différents points. Dans ce qui suit, sont présentées et interprétées quelques unes des données recueillies.

Dans l'ensemble nous pouvons constater que notre ligne directrice d'enseignement a trouvé un accueil favorable auprès des élèves. On a pu sentir chez la plupart d'entre eux un intérêt pour l'activité informatique. Le dépouillement d'un questionnaire posé aux élèves dans la dernière heure d'enseignement le confirme. Nous avons demandé aux élèves :

- a) Auriez-vous envie de continuer à travailler avec l'ordinateur ?
très _____ pas du tout
- b) Que choisiriez vous entre une heure de cours dans votre discipline favorite et une heure d'enseignement informatique ?
discipline favorite/ordinateur

Les élèves avaient à indiquer en réponse à la question a) une marque sur le trait dont les extrémités représentaient les préférences extrêmes. Tous les élèves interrogés ont placé leurs marques sur le tiers gauche du segment. Le travail à l'ordinateur leur plaisait.

Le grand **intérêt** des élèves pour ce travail est également indiqué clairement par leur réponse à la question b). Plus de 80% des élèves se sont prononcés pour la situation fictive d'une heure d'enseignement d'ordinateur. Les raisons évoquées par les élèves révèlent cependant encore l'influence sur leur choix de l'effet de nouveauté. Ils disaient par exemple qu'ils avaient eu au moins une heure de cours par semaine dans leur discipline favorite alors que les possibilités de pouvoir travailler à l'ordinateur avaient été limitées à 10 heures.

Le fait qu'une partie des élèves aurait voulu rester plus longtemps que prévu dans la salle des ordinateurs pour faire tourner des variantes de leurs programmes témoigne aussi de leur **engagement**. Cette dernière caractéristique et la **motivation de départ** des élèves ne doit pas laisser croire au même comportement de la part de tous les élèves de cet âge. On doit prendre en compte que tous les élèves concernés étaient volontaires et que ceux qui n'avaient aucun goût pour l'informatique ne se sont pas manifestés.

On doit aussi tempérer les opinions positives émises par les élèves par la disparition de l'effet de nouveauté et la diminution de l'attrait de l'ordinateur qui auraient lieu après un temps plus long de travail. En outre après un usage plus important de l'ordinateur et l'écriture de programmes plus compliqués les élèves sentiraient que, comme toute activité intellectuelle, la programmation a aussi ses contraintes.

La **programmation de jeux** tient une place importante dans les attentes des élèves à propos des activités informatiques. Ainsi de nombreux élèves interrogés auraient-ils aimé programmer des jeux en BASIC. Très vite ils se sont aperçus qu'il leur fallait des connaissances étendues. La possibilité d'utiliser dans l'atelier des jeux tout programmés a cependant stimulé les élèves d'une certaine façon. Pour les uns c'était un objectif de détente, pour les autres la prise de conscience de ce que recouvre le programme de tels jeux, les conduisait à vouloir acquérir davantage d'éléments de programmation en BASIC.

Dans la conduite de l'atelier nous cherchions à développer des compétences et un auto-contrôle chez les élèves. Ces derniers n'étaient pas toujours disposés au début à travailler un programme sans ordinateur (test à froid). De nombreux élèves préféraient programmer aussitôt à l'écran. Ce n'est que peu à peu - après avoir vécu le travail que représente la frappe d'un programme - qu'ils ont vu l'intérêt de demander une vérification du programme avant son entrée à l'ordinateur.

V - CONCLUSIONS.

Le vaste champ d'application des techniques informatiques dans la société, les réflexions théoriques et les premières expériences pratiques nous donnent à penser que dans un avenir proche, il est possible et même nécessaire d'initier les élèves du cycle

supérieur (à partir de la 9^{ème} classe⁽¹⁰⁾) à un langage de programmation en contact direct avec un ordinateur de façon qu'ils puissent utiliser l'ordinateur pour résoudre des problèmes numériques de l'enseignement mathématique obligatoire et qu'ils acquièrent des méthodes de travail et de pensée requises dans l'utilisation de techniques de traitement de l'information.

Il ne s'agit **nullement** que l'enseignement mathématique ait pour objectif de former les élèves à programmer parfaitement.

L'utilisation finalisée vers des exercices ou des activités est à notre avis une stratégie qui permet d'amener les élèves à utiliser les micro-ordinateurs.

Dans un cours d'initiation au BASIC pour les besoins de l'enseignement mathématique on n'a besoin d'introduire que les éléments du langage qui sont nécessaires pour une utilisation raisonnable de l'ordinateur en tant qu'outil.

On doit familiariser les élèves à quelques commandes de l'ordinateur et de l'écran ainsi qu'à des éléments fondamentaux d'un programme, les écritures d'expressions mathématiques (opérations +, -, *, /, et éventuellement \wedge , les parenthèses, les variables (numériques), la représentation des nombres, les fonctions SQR (X), PI, ABS (X), INT (X), éventuellement aussi RND (X), les instructions INPUT, LET, PRINT, END, IF...THEN, FOR...NEXT, GOTO. Les élèves doivent savoir en outre comment structurer un programme pour obtenir une présentation à l'écran porteuse d'informations.

Des considérations sur le concept d'algorithme ne doivent pas être préliminaires à un enseignement de BASIC. Ce n'est qu'après une initiation des élèves à un langage de programmation et après des activités à l'ordinateur qu'il nous paraît raisonnable de parler explicitement d'algorithmes.

Pour que les élèves n'élaborent pas des conceptions partielles des compétences d'un micro-ordinateur, il est recommandé de leur montrer aussi à l'occasion des exemples de résolution non numérique à l'ordinateur (par exemple des programmes de tri et de recherche, des programmes de jeux).

Il nous paraît utile de ne parler des effets numériques sur des systèmes digitaux de calcul qu'au milieu de l'enseignement alors que les élèves sont déjà initiés aux fondements de la programmation et qu'ils ont pu observer à plusieurs reprises de tels effets.

Une condition essentielle de l'intégration d'éléments d'informatique en discipline obligatoire ou facultative dans la scolarité obligatoire est l'existence régulière d'une formation continue à l'informatique des enseignants de diverses matières.

D'autres recherches en méthodologie des mathématiques devraient se développer sur ce thème, en particulier sur les exigences résultant du progrès scientifique et technique dans le domaine de l'informatique auxquelles devrait satisfaire une formation générale en mathématiques. Dans ce contexte, l'ordinateur devrait être utilisé comme instrument de résolution de problèmes mathématiques.

Certains contenus traditionnels pourraient ainsi perdre en importance mais d'autres seraient intégrés dans les programmes.

(10) NdT : La 9^{ème} classe correspond à la 3^{ème} française.

REFERENCES.

KNAUER W. 1980 Informatik als Schulfach (Informatique en tant que discipline scolaire°, *Tübingen* p. 36.

LABORDE C. 1985, Computer in der Schule - Bericht von einem Unterrichtsversuch (des ordinateurs en classe, compte rendu d'une expérimentation), *exposé à l'Université Martin Luther de Halle-Wittenberg*, 23.10.1985.

Lehrplan für die Facharbeiterausbildung - Grundlagenfach : Grundlagen der Automatisierung (Programme pour la formation des travailleurs spécialisés : fondements de l'automatisation), Berlin, 1986, *Editions d'état de la RDA*.

Programme du cours "Fondements de l'informatique et des techniques de calcul" (en russe), 1985, in *Matematika V Skole*, Vol. 3, p. 4-7.

SIMONICS I. 1984, Propositions de thèmes pour la formation aux techniques de calcul des enseignants en Hongrie (en hongrois) Veszprem, *Veszprem, OOK, Etude*.