



## Didactique et programmation à l'école

Christophe Parmentier

### ► To cite this version:

Christophe Parmentier. Didactique et programmation à l'école. Colloque francophone sur la didactique de l'informatique, Sep 1988, Paris, France. pp.129-145. edutice-00362054

**HAL Id: edutice-00362054**

**<https://edutice.hal.science/edutice-00362054>**

Submitted on 17 Feb 2009

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

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

## **DIDACTIQUE ET PROGRAMMATION À L'ÉCOLE**

**Christophe PARMENTIER**

**UFR de Sciences de l'Éducation  
Université Paris V  
28 rue Serpente  
75006 PARIS**

**RESUME**

Introduire des activités de programmation dès l'école primaire est un choix qui peut se justifier en fonction, d'une part des travaux concourants des disciplines concernées par l'informatique pédagogique, d'autre part par les premiers résultats obtenus lors de notre recherche en Sciences de l'Education. Alors, la mise en place d'un enseignement de la programmation se trouve confronté à des obstacles que nous chercherons à identifier. Certains sont liés aux choix opérés par l'institution, d'autres sont plus simplement liés aux langages et aux référents qu'ils activent, les derniers enfin, sont liés aux choix didactiques opérés lors de la mise en place des progressions.

## **DIDACTIQUE ET PROGRAMMATION À L'ÉCOLE**

**Christophe PARMENTIER**

**UFR de Sciences de l'Éducation  
Université Paris V  
28 rue Serpente  
75006 PARIS**

### **1. INTRODUCTION**

L'école doit se soucier de problèmes concernant la didactique de l'informatique depuis que les micro-ordinateurs l'ont pénétrée. A la suite de la bataille des disciplines où l'on se demandait si l'ordinateur devait être introduit dans l'enseignement comme moyen ou comme objet, les premiers textes concernant l'école primaire arrivent. Les circulaires de Mars 83 définissent des orientations concernant l'informatique, elles l'inscrivent dans les disciplines d'éveil, et anticipent la mise en place du matériel. Par le Plan Informatique pour Tous (I.P.T.) et ses diverses adaptations, les écoles primaires sont dotées de micro-ordinateurs en configurations isolées ou regroupées en réseaux. La récente analyse publiée dans "Le Monde de l'Education" [1] autorise à penser qu'il s'agissait plus d'une opération de soutien de l'industrie française dans ce domaine que d'un réel enjeu pédagogique concerté.

Toutefois les applications pédagogiques évoluent et les domaines d'utilisation s'étendent de l'E.A.O. la plus directive à l'enseignement de la programmation dès l'école élémentaire. Cette seconde tendance constitue la base des problématiques de notre recherche en Sciences de l'Education et peut se justifier, d'une part par une analyse des nombreux travaux déjà menés par des informaticiens, des psychologues, des pédagogues, des sociologues, d'autre part par les premiers résultats obtenus.

### **2. LES REACTIONS DE LA RECHERCHE EN FRANCE**

Quatre domaines de la recherche ont été prioritairement concernés par cette introduction de l'ordinateur à l'école. Nous exposerons succinctement quelques spécificités francophones de chaque tendance susceptibles d'éclairer notre problématique.

## 2.1. Les informaticiens

Sollicités à de nombreuses reprises par les instances gouvernementales lors des diverses opérations depuis qu'informatique et éducation ouvrent les voies de marchés importants, ils prennent une part active aux différents débats concernant l'informatique éducative en France. Les travaux de W. Mercouroff [2], et J. Hebenstreit [3] à la suite de l'opération des "58 Lycées" posent les premiers jalons d'une informatique pédagogique. Ils amorcent des axes de réflexion concernant une informatique objet ou outil d'enseignement. Ils permettent le développement d'un langage français de programmation : le LSE. En 1981 le rapport Le Corre-Pair [4] tente de dépasser cette querelle objet, outil, et insiste tant sur la formation que sur un contrôle des produits logiciels. Dans le même temps, sous l'influence de J. Arsac, l'informatique se constitue comme discipline, suivie par un Comité Scientifique National, dont l'enseignement ne commencera qu'en classe de seconde. La thèse de G.L. Baron [5] explique les mécanismes de la constitution de ce savoir savant en informatique.

De nos diverses rencontres et de la prise en compte de leurs travaux, il ressort clairement que l'enseignement de la programmation comme objet, à l'école élémentaire, ne semble pas utile à tous, ni nécessaire, étant donné les mutations technologiques à venir, ni pertinent pour cette classe d'âge. Toutefois, de nombreux "phantasmes Orwelliens", en partie résumés par ce propos de J. Arsac : "L'ignorance du plus grand nombre laisserait la part belle aux informatocrates", hantent nos esprits. En outre, nous aimerions pouvoir rêver avec M. Lacroix [6] lorsqu'il écrit : "...naguère le personnage de l'enfant a été le messager de la pensée alternative, de l'idéologie antirépressive. L'enfant colonisé a été un symbole mobilisateur. Et maintenant, possesseur d'un jouet merveilleux (l'ordinateur), il triomphe d'une société technicienne et déshumanisée, en tournant ses armes contre elle..."

## 2.2. Les psychologues

S'intéressant aux mécanismes intellectuels, voire thérapeutiques, impliqués en amont et en aval d'activités liées à l'informatique, ils proposent et utilisent des modèles particulièrement éclairants. J.M. Hoc [7], par exemple, met en évidence les démarches cognitives mises en oeuvre lors d'activités de planification en distinguant notamment les plans procéduraux et déclaratifs. Plus centré sur l'enfance et l'adolescence, P. Mendelsohn [8] fait porter ses travaux sur la description, l'analyse et la modélisation des processus cognitifs en jeu dans les activités de programmation. Les expériences conduites dans des lieux thérapeutiques comme celles de St Anne, l'INR de St Maurice, ou les ateliers de la Puce montrent également que des bénéfices peuvent être tirés au contact de l'ordinateur. Croire, par contre, que l'ordinateur peut avoir un effet démocratisant de lutte contre l'échec scolaire, sans une formation très approfondie, serait ne pas tenir compte de nombreuses recherches pédagogiques menées dans des perspectives sociologiques (G. Langouët [9]).

En psychologie comme dans les autres domaines, beaucoup d'études effectuées à l'étranger mériteraient d'être citées. Lors de l'introduction à ses travaux de recherche, M. Jullien-Mühlemann [10] en fait une synthèse qui l'autorise à écrire que le Logo à l'école "permettrait à l'enfant d'accumuler un certain nombre d'expériences qui lui serviraient de modèles intellectuels et favoriseraient, ultérieurement l'acquisition de structures plus formelles et plus complexes".

L'enseignement de la programmation dès l'école primaire pourrait donc être un moyen pour forger un esprit apte à résoudre des problèmes, à planifier... Prenant cette alternative à l'envers, J. Hebenstreit [11] note que "L'examen de ce qui se passe dans le secteur de l'informatique professionnelle ne permet en rien d'affirmer que ceux qui ont suivi un enseignement d'algorithmique sont plus capables ou mieux capables de poser ou de résoudre des problèmes car, comme le dit le Professeur Weizenbaum [12] dans une interview récente, si c'était vrai, alors les informaticiens devraient vivre mieux que les autres, être plus heureux en ménage que les autres et gagner plus d'argent à la Bourse que les autres !"

Par ailleurs, étant donné le découpage par matière des activités scolaires, il devient délicat, voire dangereux d'attribuer à telle activité ou à telle autre ces finalités spécifiques concernant l'éveil cognitif, et il n'est pas envisageable de créer à cette seule fin, un enseignement propre.

### **2.3. Les didacticiens et les pédagogues**

Deux types d'approches pédagogiques peuvent être distingués : d'une part, celles des didacticiens, d'autre part, celles de pédagogues travaillant à améliorer le matériel existant.

G. Vergnaud [13], par ses recherches, a apporté des éléments permettant une modélisation du champ conceptuel préalablement à toute démarche didactique. Il en a démontré l'efficacité. De nombreuses études ont donc été entreprises dans cette perspective. Ce colloque devrait d'ailleurs être le meilleur témoin de l'état d'avancement des recherches entreprises dans ce domaine. En conséquence, cette partie ne sera pas plus détaillée. Il convient toutefois de constater que les analyses du champ conceptuel et les problèmes didactiques qui surgissent alors, montrent assez clairement les limites d'un enseignement de la programmation à l'école.

Pour tenir compte de ces études, mais aussi pour s'adapter à une technique évoluant très rapidement, l'INRP a développé des outils pour favoriser un enseignement de la programmation à l'école : traduction du Logo en français, création de la tortue de sol, logiciels de simulation utilisant la programmation (Ecluse...), pilotage de robots par programmation... Il semble donc que l'innovation pédagogique encouragée par les instances officielles irait dans le sens d'un enseignement de la programmation.

## 2.4. Les sociologues

Différents rapports ont fait prendre conscience de l'informatisation sans cesse croissante de notre cadre de vie et de l'impérieux besoin "d'éviter un retard de l'école par rapport à la demande sociale" (circulaire n° 87-319). Ceux de Nora-Minc [14] en 1978 et J.C. Simon [15] en 1980 ont considérablement marqué les esprits. Toutefois, compte tenu de l'importance de la réussite, de la valeur de modèle pédagogique, cette adaptation de l'enseignement à l'informatique doit se faire en fonction d'une réflexion pédagogique préalable. Un enseignement presse-bouton, une "gadgeto-pédagogie", conduisent indubitablement le corps enseignant, habitué à voir se succéder modes et ministres, à opposer une forte résistance à l'encontre de ces méthodes. Quelques études menées sur le terrain (CAFIP [16], FEN [17], H. Josserson [18]) en témoignent.

Les limites de l'utilisation d'une E.A.O. "pauvre et encore limitée de l'ordinateur" (circulaire n° 87-160) semblent être prise en compte par les institutions. Toutefois, il convient de constater que l'utilisation de ces logiciels peuvent conduire à des apprentissages (Parmentier [19]). Ils sont construits en utilisant les potentialités des langages informatiques : boucles, séquentialité, sous-programmes, itération, concaténation ... Ces mécanismes de programmation, utiles à la construction des didacticiels, permettent donc de médiatiser certains savoirs. De nombreuses études pédagogiques depuis la création de l'enseignement coopératif, montrent clairement l'efficacité de l'apprentissage de formes de communication du savoir (A. Robert [20]). C'est aussi l'articulation de ces deux constats qui fut à la base de notre problématique.

## 2.5. Notre point de vue

Une recherche en Sciences de l'Education se devait de prendre en compte l'ensemble de cette dynamique au risque de ne pas satisfaire aux exigences spécifiques de tel ou tel point de vue. Notre problématique, construite à partir du terrain (l'école primaire), est un reflet des moyens matériels, humains et institutionnels existants.

## 3. NOTRE RECHERCHE

A l'issue de cette réflexion, nous souhaitons orienter nos travaux vers la prise en compte d'un enseignement de la programmation afin de constater son efficacité dans d'autres domaines de connaissance par le jeu de transferts. Hoc et Mendelsohn [21] ont, sur ce sujet, récemment écrit que "Beaucoup voient dans l'introduction de l'informatique dans l'enseignement un moyen de favoriser d'autres acquisitions. Il s'agit là de l'étude d'un transfert dont l'identification du résultat et des mécanismes n'est pas aisée et qui nécessite le développement de recherches approfondies, qui présentent des exigences plus fortes que les seules "expériences" pédagogiques."

L'expérimentation, dans sa phase active, concerne 48 élèves de deux classes de CM 1 en banlieue parisienne. Ce niveau a été retenu car les élèves n'avaient pas encore été concernés par l'informatique, et que les programmes la rendent en partie obligatoire à partir de ce niveau. Par ailleurs, ils connaissaient peu les concepts géométriques impliqués.

Nous avons cherché à comparer les résultats obtenus par 24 élèves découvrant implicitement certains concepts grâce à l'apprentissage de la programmation en Logo graphique, à 24 autres découvrant ces mêmes concepts explicitement sans ordinateur. Les deux traitements pédagogiques ont été administrés aux groupes selon le plan suivant :

|          | test 1 | entrainement | test 2 | entrainement | test 3 |
|----------|--------|--------------|--------|--------------|--------|
| groupe A |        | LOGO         |        | géométrie    |        |
| groupe B |        | géométrie    |        | LOGO         |        |

A la suite du test 1, deux classes sont respectivement scindées en 2 groupes homogènes : A et B. Ils suivent alors simultanément deux progressions préalablement définies, de 12 h. chacune, dans un ordre différent.

Les progressions ont été construites en équipe avec les différents partenaires concernés. Elles s'inscrivent dans le cadre scolaire. Des traces de l'activité des élèves sont régulièrement relevées. La progression "Logo" propose après 2 séances de découverte des primitives fondamentales de faire dessiner sur l'écran des figures choisies dans un corpus ordonné selon des difficultés croissantes. L'agencement de celles-ci permet, l'acquisition de l'itération, puis, la définition de procédures. Les figures proposées sont des polygones, des figures composées de polygones, de segments, d'angles droits ou non-droits. La progression de géométrie sans ordinateur, conduit les élèves à découvrir l'angle pour l'intégrer comme élément lors de situations de description. Ce descripteur sera alors retenu et combiné à d'autres afin d'élaborer une classification des polygones et figures inscrites au programme.

Des tests ont été construits et soumis à un large échantillon d'élèves, ils permettent de rendre compte des compétences lors du calcul de périmètres ou de mesures d'angles. Par ailleurs, ils vérifient l'évolution de certains pré-requis concernant l'aptitude à itérer une suite ou à discriminer droite et gauche sur un plan orienté. Enfin, ils évaluent certains "savoir faire-faire" en Logo.

#### 4. QUELQUES RESULTATS DU GROUPE A

A l'heure où nous écrivons, notre expérience n'est pas terminée. Elle ne le sera qu'à la fin de l'année scolaire 1988 pour cette partie expérimentale. Toutefois nous pouvons déjà utiliser quelques résultats qui nous semblent suffisamment significatifs pour



argumenter en faveur de l'introduction d'activités de programmation dès l'école primaire. Chaque résultat exposé appelle des commentaires d'ordre didactiques qui seront proposés dans la partie suivante.

Les 24 élèves du groupe A, ayant suivi une progression sur un trimestre de programmation LOGO, sans aucun autre complément géométrique, ont fait certaines acquisitions autres que celles spécifiques à LOGO. Les résultats suivants en témoignent :

#### 4.1. Calcul de périmètres groupe A

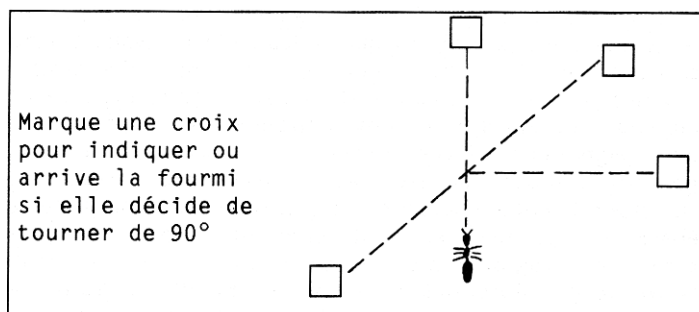
|        | moy.ex. | avec + | avec * |
|--------|---------|--------|--------|
| test 1 | 4,50    | 3,20   | 1,29   |
| test 2 | 6,62    | 1,71   | 4,91   |

Les tests 1 et 2 sont identiques, mais, comme le montre notre plan expérimental, ils sont passés avant et après un trimestre d'apprentissage. Ils proposent une série de situations (polygones, figures composées de polygones, problèmes) pour lesquelles les élèves doivent indiquer la mesure du périmètre et la justifier par une opération (cf annexe 1). Pour chaque situation un minimum d'information est donné, obligeant ainsi à des calculs. Le passage de l'addition à la multiplication est associé à la découverte de certaines propriétés des polygones. La colonne "moy. ex." représente la moyenne d'exercices réussis par élève, cette moyenne est ensuite décomposée suivant les stratégies adoptées : addition ou multiplication.

Nous constatons que les élèves ayant suivi la progression d'un trimestre de LOGO ont amélioré qualitativement et quantitativement leurs performances pour le calcul des périmètres proposés.

## 4.2. Calcul de mesures d'angles groupe A

|        |    |
|--------|----|
| test 1 | 8  |
| test 2 | 23 |



Des progrès considérables sont également notés lors de la passation de cette épreuve. Par contre 15 élèves utilisent encore TD 90, lors des tests informatiques, pour faire dessiner des angles ne faisant pas  $90^\circ$ .

Enfin, lors du passage, au test 2, d'épreuves concernant des mesures d'angles (cf annexe 2) à obtenir en utilisant, soit la complémentarité ou la suppléantarité, soit des propriétés des polygones, on constate que chaque élève réussit en moyenne 3,5 ex. sur les 10 proposés. Cette moyenne constitue une amélioration par rapport au résultats nuls que nous aurions été en droit d'attendre, au vu des pré- expériences, si nous avions fait passer ces épreuves au test 1. Les résultats obtenus par le groupe B (3,33) ne sont pas significativement différents. Toutefois les répartitions des populations ne sont pas identiques ; les écarts-types en témoignent :  $A = 2,64$  ;  $B = 1,62$ .

Ces premiers résultats, rapidement présentés, montrent qu'un enseignement de la programmation en Logo graphique au Cours Moyen peut se justifier par des acquisitions inscrites dans les Instructions Officielles. Mais cet enseignement se heurte alors à de nombreux problèmes didactiques, qui expliquent en partie les phénomènes exposés, qu'il convient maintenant de détailler

## 5. PROBLEMES DE DIDACTIQUES DE L'INFORMATIQUE

### 5.1. Ceux liés à l'institution et à ses choix

L'activité de programmation n'est pas autonome, elle est contextualisée. A l'école elle dépend des orientations prises par les instituteurs. Ces choix créent des assimilations, marquent des options, avec lesquelles les institués doivent composer. Nous les avons en partie évoqués puisqu'ils devraient provenir des modèles dégagés par la recherche, mais l'I.P.T. n'a pas pris en compte cet aspect des choses !

La forme des dotations oblige à un fonctionnement particulier car les machines ne sont pas dans la classe. Elles sont regroupées dans des sites. L'activité informatique, liée à ces sites, a donc le même statut que la piscine, le gymnase ou la bibliothèque puisqu'elle nécessite un déplacement. Elle se situe, de ce fait, à mi-chemin entre un loisir d'adhésion et une discipline scolaire, mais dans ce cadre, elle ne revêt plus aucun caractère spontané. Par ailleurs, les machines sont enfermées dans des salles blindées, alors qu'enseignants et élèves ressentent un manque de moyens adaptés : ballons, peinture, ateliers de lecture... Il n'est pas certain que les choix faits correspondaient réellement à une attente de la part des enseignants ou des élèves. Il s'agit maintenant de créer un besoin alors que par essence, une activité de programmation devrait s'inscrire dans une continuité. Par ailleurs la forme des dotations oblige à un travail en groupe au niveau, soit de l'harmonie entre les enseignants, soit de l'investissement des élèves et la structure de groupe mise en modèle par les dotations est celle du réseau piloté par un serveur. Cette structure à forte connotation hiérarchique, habituelle lors des situations de communications didactiques, n'est pas un modèle qui favorise réellement une activité de programmation.

Les dotations reçues par les écoles ne semblent pas adaptées à une utilisation intensive tel que le site informatique le prévoit. En effet, un seul site pour des groupes scolaires comportant jusqu'à 4 classes par niveau voit défiler les élèves durant 6 heures par jour. Ces élèves sont enthousiastes et empressés. Les machines sont légères, sensibles et mal ventilées. Alors, malgré toutes les précautions prises, après un à deux ans de fonctionnement, 1 poste sur 4 n'est plus fiable. Par ailleurs les problèmes de maintenance sont presque insolubles ! Tout cela contribue à forger des réactions de défense envers l'ordinateur parfois fort saines mais pas toujours fécondes. Ce propos d'un élève en témoigne : «Je préfère pas faire avec "pour" (créer une procédure) parceque ça sert à rien si elle se plante encore !\_».

Les conséquences didactiques ne sont pas objectivement mesurables puisqu'il semble impossible d'isoler ces variables. Seules des recherches internationales permettraient de comparer l'efficacité de certaines décisions. Toutefois, lorsque ces générations auront atteint leur pleine maturité ces choix seront en jeu, il convient donc de ne pas occulter ces éléments pour les évaluations futures.

## **5.2. Ceux liés à Logo :**

De nombreuses critiques ont été émises à propos de Logo. Si cet outil semble adapté à un enseignement de la programmation à l'école primaire, de nombreuses modifications pourraient encore être apportées afin que s'atténuent les obstacles didactiques véhiculés par l'adaptation française de ce langage. A. Giordan [22] et son équipe en ont relevés certains et la thèse de F. Tréhard [23] en souligne d'autres. Toutefois, certains problèmes sont plus marquants lorsqu'ils fondent, comme chez nos élèves, des représentations initiales, alors, une analyse étayée d'observations sur le terrain semble nécessaire.

Logo est un système sans unité explicite où les ordres sont conférés ainsi : AV 30 TD 30. L'élève doit assimiler que l'ordre comprend en lui même la notion d'unité. AV 30 veut dire : "avance de 30 pas de tortue" et TD 30 : "tourne de 30°". Les différentes définitions d'écran déterminent les valeurs réelles, ce qui conduit par exemple, à assimiler indifféremment à un cercle tous les polygones dont le nombre de côtés est important. La distinction a priori entre une figure représentant un polygone de 25 côtés et un cercle se fera en fonction de la surface et de la qualité l'écran, ce qui constitue un référent bien complexe pour un élève de CM 1 !

Ce manque d'unité discriminante renforce également l'assimilation souvent constatée auprès d'élèves travaillant sur les angles, entre longueur et écartement des côtés. Lors de la réalisation de mesures d'angles sur papier, beaucoup d'élèves du groupe A, commencèrent par utiliser la règle afin de mesurer la longueur des cotés. Aucun élève du groupe B n'a réagit de la sorte.

Nous avons également noté au paragraphe 4.2. que, si 23 élèves étaient capables d'identifier un angle de 90°, 15 utilisaient encore TOURNE 90 pour faire dessiner des angles non-droits. Pour eux TOURNE 90 permet d'obtenir indifféremment un angle de 90°, 60°, ou tout autre. L'argument qui suit l'ordre de rotation ne correspond pas, pour ces élèves, à une mesure parfaitement fonctionnelle. Cette différence constatée entre l'analyse et la programmation provient encore d'un manque d'outils adaptés permettant d'établir, de manière certaine, la liaison entre les deux systèmes de représentation ayant un référent commun.

Par ailleurs, le système de mesure des angles Logo est basé sur la division du cercle en 360°. Il suppose donc une certaine représentation de la division que n'ont pas tous les élèves en début de cycle moyen. Enfin, les effets de TOURNE peuvent se combiner, par addition et soustraction, dans un système de base 360, dans lequel, 0 et 360 ont un effet identique. Si cette logique rappelle celle du cadran horaire, ou du Big-Trak, elle constitue toutefois un système radicalement nouveau pour les élèves au début du cycle moyen. Le travail sur les angles en Logo, s'appuyant donc sur les concepts de division et de base, permet des conjectures performantes lorsqu'il est utilisé dans une géométrie sans ordinateur, mais ne semble pas facilement accessible. C'est, en partie, ce qui justifie la différence des écarts type constatée au paragraphe 4.2.

Ces résultats présentent les obstacles didactiques que constitue Logo à l'apprentissage du concept mathématique d'angle dans une géométrie classique. Afin de pouvoir vérifier si ces constats sont réversibles, donc, si une assez bonne représentation de la notion d'unité, constitue, dans un premier temps, un obstacle à l'acquisition et à la maîtrise du Logo graphique, nous devons attendre la deuxième partie de notre expérience et observer le groupe B. On peut imaginer dans cette suite que d'autres représentations ou savoirs puissent faire obstacle à l'acquisition de mécanismes de programmation adaptés à des dispositifs spécifiques. Un second exemple, nous est fourni par J. Aarsac [24] lorsqu'il

constate que les enseignants de français refusent habituellement cette définition des ancêtres : "Mes ancêtres sont mes parents ou les ancêtres de ceux-ci." Ce refus enseigné pourrait créer un obstacle à l'apprentissage d'un mécanisme de programmation : la récursivité. La programmation étant bien une certaine maîtrise du temps et un contrôle du mouvement, tout une dynamique personnelle en étaye donc l'ontogénèse.

Par contre, depuis Babbage et von Neumann jusqu'à l'intelligence artificielle, les machines programmables ont évolué. Cette évolution n'a pu se faire qu'en référence à des modèles et dès 1967 John Lilly [25] signalait des parallèles entre modèles intellectuels et informatiques dans son ouvrage "Programming and Metaprogramming of the Bio-Computer". Cette dynamique interactive des possibilités d'adaptation d'un sujet à des logiques de programmation conçues par d'autres, va encore nourrir la littérature scientifique pendant plusieurs années.

### 5.3. Ceux liés à la progression

L'absence d'unités dans le système Logo est un obstacle important, il est d'autant plus marquant que Logo est une géométrie dynamique alors que la géométrie papier-crayon est une géométrie statique. Logo conduit à linéariser les images : c'est à dire à les transformer en un trajet, puis en un discours à propos de ce trajet exprimé enfin en un programme adapté au dispositif. Cette propriété est certainement à la base des améliorations notées au paragraphe 4.1., pour le calcul du périmètre, puisqu'elle oblige à un travail prenant en compte les côtés.

Des rapports étroits existent entre multiplication, et itération. Piaget fait figurer l'itération parmi les structures logico-mathématiques et les travaux de recherche de H. Kayler, n'en sont qu'un exemple. L'acquisition de l'itération, son fonctionnement en Logo, favorisent la découverte corrélatrice, sous forme de théorèmes en acte, d'invariants liés aux polygones. Lors de l'élaboration d'un programme ayant pour but de faire dessiner un carré, le passage de [AV 30 TD 90 AV 30 TD 90 AV 30 TD 90 AV 30] à REPETE 4 [AV 30 TD 90] suppose la prise en compte de l'invariance des 4 côtés du carré. Il doit pouvoir s'accompagner du passage de  $30+30+30+30$  à  $4*30$  pour le calcul du périmètre. Cette explication permet, en partie, de justifier des améliorations notées au paragraphe 4.1. Lors de la mise en place de notre progression Logo, nous avons donc cherché à introduire prioritairement l'itération afin d'en optimiser l'effet. De ce fait, certains éléments annexes sont venus, par assimilation, se conjuguer à l'itération retardant ainsi son acquisition.

Deux étapes sont habituellement dégagées lorsqu'on observe des débutants en Logo graphique : le mode pilotage où est contrôlée l'adéquation entre l'anticipation et le résultat obtenu pour chaque ordre donné, et le mode de programmation où tout un programme (une suite d'ordres agencés) est alors vérifié. Pour faire passer d'un mode à l'autre on peut soit interdire d'appuyer plus d'une fois sur la touche "entrée", ce qui constitue une

contrainte peu justifiable, soit apprendre à définir et à utiliser des procédures puisque Logo est un langage procédural, soit introduire `REPETE` qui oblige à saisir un programme complet, avec ses structures de contrôle et sa syntaxe, avant de le valider.

Dans notre progression, nous avons prioritairement introduit la commande `REPETE`. Nous l'avons déjà justifié. Mais certains élèves n'étaient pas encore prêts à passer du pilotage à la programmation. Ils n'avaient pas encore assez jubilé de leur pouvoir naissant sur la machine. Leur latéralisation les prenait trop souvent à défaut ou ils n'avaient pas encore les moyens de dissocier dans le temps conception et réalisation. `REPETE` et mode de programmation étant liés, leurs acquisitions se trouvaient retardées. Elle ne fut effective que pour ceux qui apprirent à créer et utiliser une procédure. Ils pouvaient alors dissocier les fonctions, en accepter certaines, et éventuellement en refuser d'autres.

## 6. CONCLUSIONS

Le choix opéré dans l'agencement des savoirs à acquérir, parmi les outils à utiliser, en fonction d'objectifs précis constitue une nécessité. Toutefois cette organisation constatée dans les paragraphes 5.1. et 5.3., ainsi que les choix matériels du paragraphe 5.2., créent des assimilations, des "effets de bord". L'école est un terrain particulièrement sensible de ces effets puisque l'activité de programmation n'y est absolument pas spontanée et qu'elle fonde des représentations initiales. Il s'agit, en fait, des effets didactiques de l'organisation et de la planification de l'activité de programmation.

Notre étude, trop partiellement présentée, devrait permettre de donner à l'enseignement de la programmation à l'école une nouvelle dimension grâce notamment aux transferts spontanés résultants de l'acquisition de structures de programmation et les champs conceptuels auxquels elles sont appliquées. D'autres domaines que les rapports entre l'itération et le calcul du périmètre pourraient être ainsi explorés. La concaténation et la conjugaison doivent, par exemple, entretenir des rapports similaires.

C'est tout une réorganisation des savoirs à l'école qui est en jeu, adaptant les concepts pédagogiques à ceux mis en évidence par les nouveaux outils employés.

## BIBLIOGRAPHIE

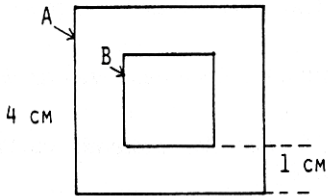
- [1] Le Monde de l'Education 11/87
- [2] W. MERCOUROFF, *L'expérience des "58 lycées"* E.I. 05/80 p 10-15
- [3] J. HEBENSTREIT, *Computers in Education : The french approach*. Alberta printout 05/83 V.4 N°2 p 19-24
- [4] LE CORRE-PAIR, *L'introduction de l'informatique dans l'Education Nationale*. rapport 10/81
- [5] G.L. BARON, *La constitution de l'informatique comme discipline scolaire*. Thèse 10/87
- [6] M. LACROIX, *Micro-informatique reflet des moeurs*. Esprit 02/85 p 9-21
- [7] J.M. HOC, *Psychologie cognitive de la planification*. PUG 87
- [8] P. MENDELSON, *Rapport d'activité 83/87*
- [9] G. LANGOUET, *Technologie de l'éducation et démocratisation*. PUF 1985
- [10] M. JULLIEN-MUHLEMAN, *Etude de quelques effets de l'initiation à l'informatique chez les élèves de CM2*. DEA 09/84
- [11] J. HEBENSTREIT, *Informatique et enseignement*. La vie des sciences 11/84 V.1 N°5 p 381-398
- [12] J. WEIZENBAUM, *Puissance de l'ordinateur et raison de l'homme*. Edition informatique 1981
- [13] G. VERGNAUD, *Concepts et schèmes dans une théorie opératoire de la représentation*. Psychologie Française 11/85 p 245-262
- [14] NORA-MINC, *L'informatisation de la société*. Point Seuil 78
- [15] J.C. SIMON, *L'éducation et l'informatisation de la société*. La documentation Française 1980
- [16] CAFIP LILLE, *Rapport d'évaluation 84/85*
- [17] M. DELAPIERRE, E. PELISSET, J. VICARD, *Un essai de bilan de l'expérience actuelle analyse d'une enquête par questionnaire*. Les cahiers de la FEN 07/85 p 43-192
- [18] H. JOSSERON, *Des maîtres et des ordinateurs : l'école résiste à l'innovation technologique*. Le Binet Simon 03/87 N°612 p 3-15

- [19] C. PARMENTIER-FALVERT, P.L. COUTURAT, C. PARMENTIER, *De l'introduction de l'ordinateur à l'école, approches*. Maitrise 84/85
- [20] A. ROBERT, *De quelques spécificités de l'enseignement des mathématiques dans l'enseignement post-obligatoire*. Cahiers de didactique IREM PARIS 7 N° 47
- [21] J.M. HOC, P.MENDELSON, *Introduction*. Psychologie Française 12/87 Tome 34 p 211 217
- [22] A. GIORDAN, J.L ZIMMERMAN, C. NIEDEGGER, *Logo deux ans après*. E.I. N°23 11-12/84
- [23] F. TREHARD, *Logiciels pouvant impliquer des activités mathématiques à l'école élémentaire*. Thèse 05/87
- [24] J. ARSAC, *Les machines à penser - des ordinateurs et des hommes*. Le Seuil 87
- [25] C. ENLART, *Fondement humains des programmes et des métaprogrammes*. Editions Cohérence 84
- [26] H. KAYLER, *Mathematical learning and repeat*. Proceeding of the second international conferences for LME 07/86 p 9-13



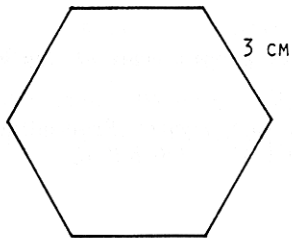
## ANNEXES 1

Nom : \_\_\_\_\_ AGE : \_\_\_\_\_ DATE : \_\_\_\_\_

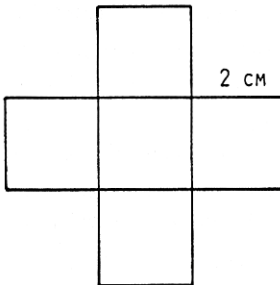


A PARCEQUE : \_\_\_\_\_

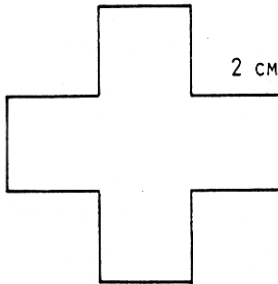
B PARCEQUE : \_\_\_\_\_



PARCEQUE : \_\_\_\_\_



PARCEQUE : \_\_\_\_\_



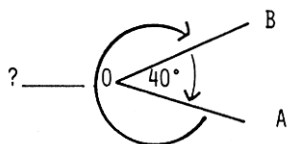
PARCEQUE : \_\_\_\_\_

## ANNEXES 2

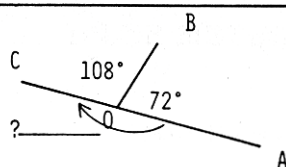
NOM : \_\_\_\_\_ AGE : \_\_\_\_\_ DATE : \_\_\_\_\_

FIGURE À COMPLÉTER

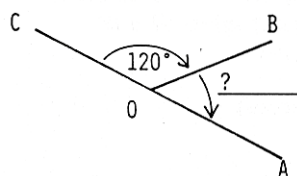
EXPLICATIONS



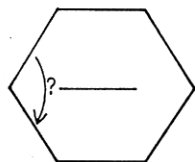
PARCEQUE : \_\_\_\_\_  
 \_\_\_\_\_



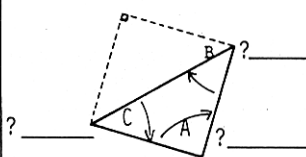
PARCEQUE : \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_



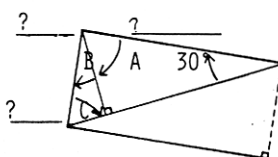
PARCEQUE : \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_



PARCEQUE : \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_



A PARCEQUE : \_\_\_\_\_  
 B PARCEQUE : \_\_\_\_\_  
 C PARCEQUE : \_\_\_\_\_  
 \_\_\_\_\_



A PARCEQUE : \_\_\_\_\_  
 B PARCEQUE : \_\_\_\_\_  
 C PARCEQUE : \_\_\_\_\_  
 \_\_\_\_\_