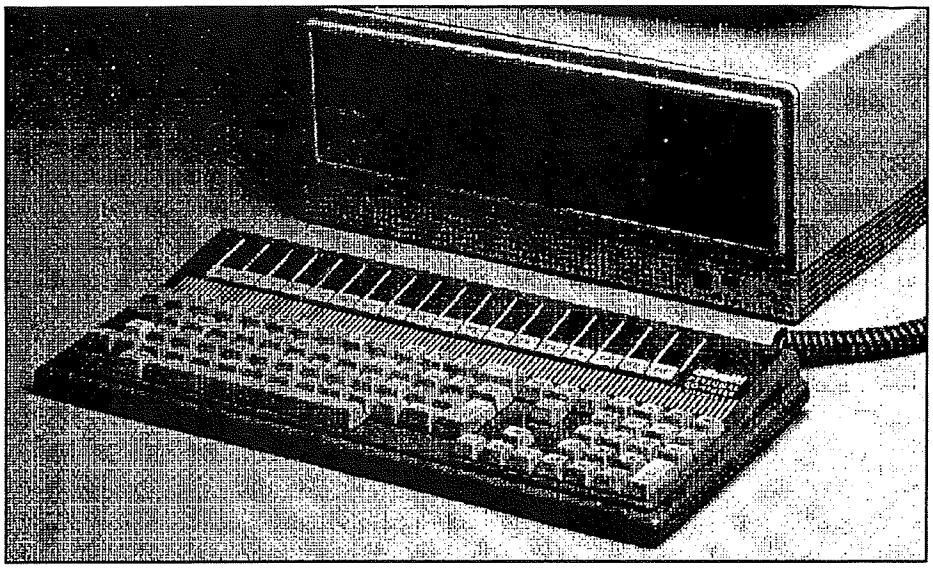
REPUBLIQUE ET CANTON DE GENEVE

Département de l'instruction publique

DISPOSITIF DE RECHERCHE

de l'enseignement secondaire

APPRENTISSAGE DU CLAVIER



Didacticiel réalisé par

Anne-Marie Praplan
et
Jean-Claude Armici

Genève 1988

(c) 1988, Département de l'instruction publique de Genève et
Dispositif de recherche de l'enseignement secondaire
avenue du Bouchet, 16
1209 Genève
Tous droits réservés

Pour tous renseignements:
SMAV, Service des moyens audio-visuels
Cité-Vieussieux 9
1203 Genève

APPRENTISSAGE DU CLAVIER

Didacticiel réalisé par

Anne-Marie Praplan
et
Jean-Claude Armici

Genève 1988

Turbo Pascal est un produit de la firme Borland International, Inc.
MS-DOS est une marque déposée de la firme Microsoft Corporation.
IBM PC/XT/AT sont des produits de la firme IBM Corporation.
Olivetti est une marque déposée de Ing. C. Olivetti & C., S.p.A.
PageMaker est un produit de la firme Aldus Corporation.
ALR est un produit de Wearnes Technology Limited.

Le contenu et la mise en page de ce document ont été entièrement réalisés par les auteurs à l'aide du traitement de texte Word 4.0 (Microsoft), du logiciel de PAO PageMaker 3 (Aldus) et du logiciel de dessin Picasso (CPI S.A.).

CONTENU

1. Présentation du projet 7

1.1 Introduction 7
1.2 But visé 7
1.3 Public visé 7
1.4 Ressources disponibles 8

2. Description du didacticiel 8

2.1 Organisation 8
2.2 Présentation sommaire du contenu des modules 10
2.3 Options prises au cours du développement 14
2.4 Paramétrage et configuration du didacticiel 14

3. Fonctionnement du didacticiel 16

3.1 Point de vue de l'enseignant 16
3.2 Fichiers nécessaires 16
3.3 Installation sans réseau 17
3.4 Installation en réseau 19

4. Phase de développement 19

4.1 Introduction 19
4.2 Elaboration du scénario 19
4.3 Obstacles 20
4.4 Développement d'outils (généraux ou spécifiques) 20

5. Difficultés rencontrées 21

5.1 Terminologie 21
5.2 Cohérence 21
5.3 Enchaînement des modules 22
5.4 Problèmes de "timing" 22
5.5 Problèmes techniques 23

6. Phase d'évaluation 24

6.1 Tests du didacticiel 24
6.2 Collaboration externe 24
6.3 Comment tenir compte des résultats des tests 24
6.4 Validité des tests 25
6.5 Diffusion du didacticiel en test 25

7. Echos, questions et remarques 25

7.1 Diversité des réactions 25
7.2 Quelques exemples de critiques 26
7.3 Répercussions sur le didacticiel 30

8. Buts atteints et buts non atteints 30

8.1 Points négatifs 30
8.2 Points positifs 30
8.3 Suggestions 31

9. Remarques générales 31

9.1 Importance de la phase d'étude 31
9.2 Importance de la phase d'implémentation 32
9.3 Limites techniques 32
9.4 Adaptabilité et durabilité du didacticiel 32

10. Description technique 34

10.1 Choix du langage 34
10.2 Choix des outils 35
10.3 Organisation et modularité 35
10.4 Quelques chiffres 35

11. Parties réutilisables du didacticiel 36

1. PRESENTATION DU PROJET

1.1 Introduction

Ce didacticiel à été réalisé dans le cadre du Dispositif de recherche de l'enseignement secondaire. Il fait partie d'un projet comportant deux parties: l'apprentissage du clavier constitue la première partie, la seconde étant consacrée à l'apprentissage du système d'exploitation MS-DOS.

1.2 But visé

Dès le départ, notre intention était d'accomplir un travail visant un objectif pédagogique et utilisant l'informatique comme moyen de développement. Pour diverses raisons, il est apparu opportun de choisir comme champ d'application l'informatique elle-même. Puis il a semblé logique de s'attaquer au premier problème rencontré par ceux qui abordent l'informatique: la confrontation avec un clavier d'ordinateur.

L'on sait, en effet, que beaucoup de personnes et d'élèves éprouvent de l'appréhension, parfois un blocage, face à un clavier. Il n'en reste pas moins qu'une certaine maîtrise du clavier est indispensable à qui veut réellement utiliser un ordinateur. Par ailleurs, le choix de cet objectif d'apprentissage nous a semblé d'autant mieux convenir qu'il est constitué par un ensemble de processus relativement simples, mais diversifiés, que pour l'enseignant cette partie du cours peut s'avérer répétitive et qu'idéalement elle devrait se dérouler par très petits groupes, voire avec un seul élève à la fois.

Le didacticiel d'apprentissage du clavier est censé intervenir dans cette période d'approche et de sensibilisation.

1.3 Public visé

Le public susceptible d'utiliser ce didacticiel, en tant qu'apprenant, est composé de toute personne utilisant un ordinateur pour la première fois. Le didacticiel peut

être utilisé par un élève seul, de manière autonome, s'intégrer dans un cours, être découpé en plusieurs étapes, être parcouru séquentiellement, ponctuellement ou bien selon une progression déterminée par la personne qui l'a paramétré (en général l'enseignant).

Le langage employé dans le didacticiel s'adapte plutôt à des élèves qu'à des adultes. Ce choix peut être modifié facilement si l'on désire adapter le style des interventions à un autre public.

1.4 Ressources disponibles

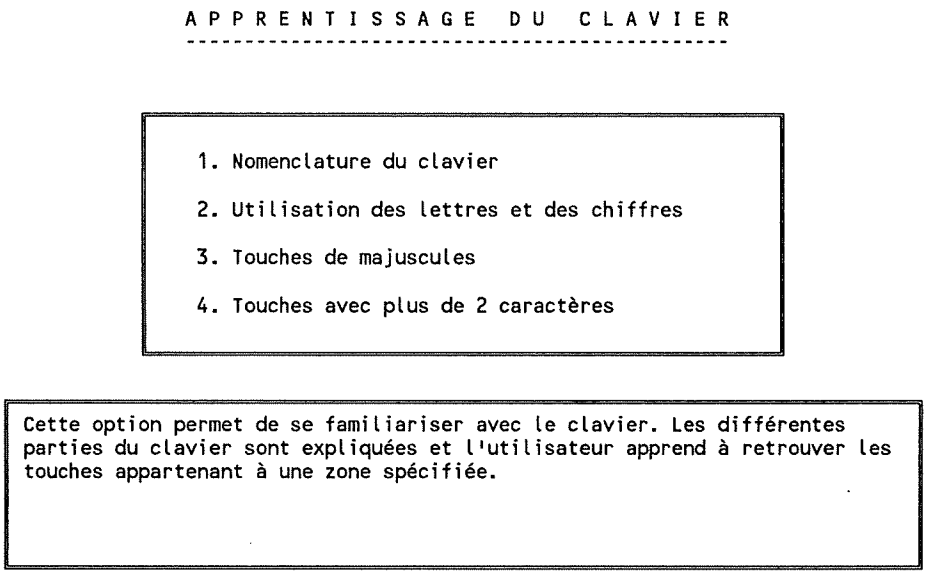
Les moyens à disposition pour la réalisation de ce projet sont essentiellement des heures de dégrèvement obtenues par l'intermédiaire du Dispositif de recherche. Le développement proprement dit a été effectué sur les ordinateurs personnels des auteurs. Le langage, les utilitaires et autres produits nécessaires étaient également personnels. Au cours des dernières étapes, le travail a pu bénéficier de l'utilisation d'ordinateurs très rapides.

2. DESCRIPTION DU DIDACTICIEL

2.1 Organisation

Le didacticiel est organisé autour d'un menu proposant l'accès à onze modules. Un module traite d'une ou de plusieurs notions logiquement apparentées. Chaque module forme un tout, mais dépend généralement des modules précédents. Dans l'apprentissage du clavier il est cependant difficile d'ordonner exactement la chronologie des différentes notions à acquérir. Certaines devraient être abordées simultanément, d'autres peuvent l'être de manière indépendante.

Le contenu des modules a été déterminé à partir d'une liste des notions qui devraient figurer dans le didacticiel. Par regroupement des sujets, les onze modules ont été spécifiés. Nous les avons ensuite ordonnés d'une manière qui nous a semblé logique. Il convient de préciser que d'autres approches pourraient se révéler tout aussi logiques. Nous verrons par la suite que l'ordre des modules est une caractéristique qui perd de son importance du fait de la paramétrisation.



Menu du didacticiel

Voici la liste des modules telle qu'elle apparaît dans le menu du didacticiel:

1. Nomenclature du clavier
2. Utilisation des lettres et des chiffres
3. Touches de majuscules
4. Touches avec plus de 2 caractères
5. Touches mortes (accents)
6. La touche "return"
7. Touches de déplacement du curseur
8. Touches de fonction
9. Le pavé numérique et la touche "funct lock"
10. Utilisation des menus
11. Test général

Parmi ces modules, on aura remarqué que les deux derniers ne concernent pas directement l'apprentissage du clavier.

Il nous a pourtant semblé nécessaire d'inclure l'utilisation des menus, car elle est omniprésente dans les applications informatiques, que ce soient des langages de

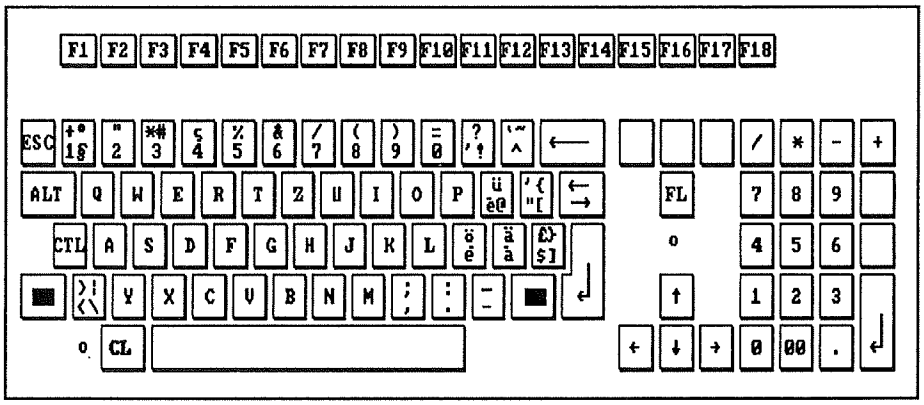
programmation, des logiciels d'application, des traitements de texte... Or, un néophyte est souvent perdu devant un menu, ne serait-ce que le plus simple.

Cet avant-dernier module place donc l'utilisateur devant toutes sortes de menus dont l'utilisation fait intervenir des touches ou des groupes de touches étudiés précédemment.

Quant au dernier module, c'est un test qui permet de reprendre différentes notions abordées dans le didacticiel.

2.2 Présentation sommaire du contenu des modules

Module 1. Nomenclature du clavier



Ce module présente les différentes parties ou groupes de touches du clavier. La reproduction du clavier sur l'écran permet de visualiser les touches concernées.

Module 2. Utilisation des lettres et des chiffres

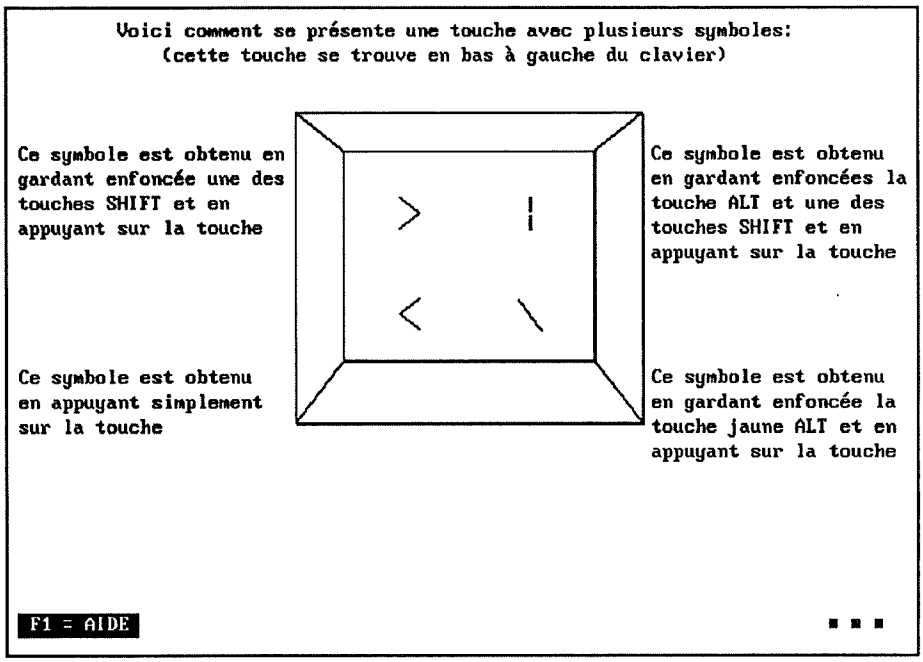
La distinction entre les lettres et les chiffres n'est pas forcément évidente pour le néophyte (différentiation entre la lettre 'l' et le chiffre '1', par exemple).

Module 3. Touches de majuscules

Ce module explique l'utilisation des différentes touches de majuscules que l'on

trouve sur un clavier informatique. Des tests placent l'utilisateur dans différentes situations.

Module 4. Touches avec plus de 2 caractères



Les claviers informatiques utilisés avec les ordinateurs OLIVETTI possèdent plusieurs touches comportant plus de deux signes, ce qui ne facilite pas l'approche du débutant. Dans ce module, l'utilisateur peut, entre autres, s'exercer à utiliser ces touches un peu spéciales.

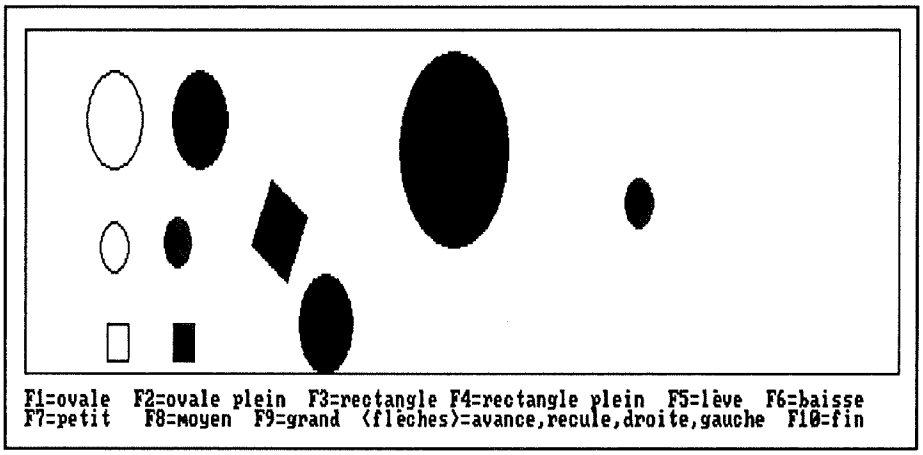
Module 5. Touches mortes (accents)

Certaines touches des claviers sont "mortes", c'est-à-dire qu'elles s'utilisent en combinaison avec d'autres touches. Il est essentiel de savoir les employer, par exemple, pour le traitement de texte.

Module 6. La touche "return"

Ce module explique l'utilité et la fonction de cette touche propre aux claviers informatiques.

Module 7. Touches de déplacement du curseur



Exercice faisant appel aux touches de fonction

Par quelques tests d'adresse, entre autres, ce module permet de se familiariser avec cet ensemble de touches.

Module 8. Touches de fonction

Dans la plupart des logiciels les touches de fonction sont largement employées. Ce module permet à l'utilisateur de saisir le concept de touche de fonction à l'aide de mises en situation.

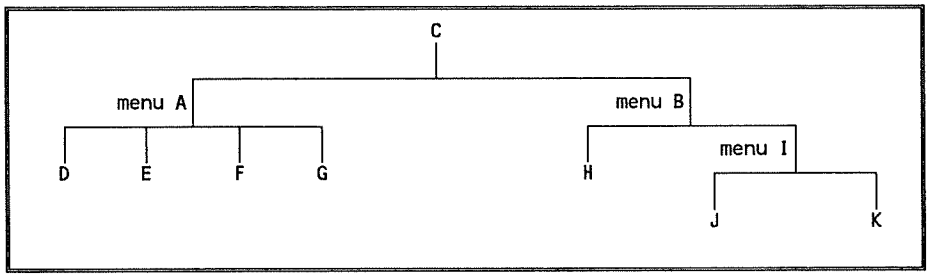
Module 9. Le pavé numérique et la touche "funct lock"

Cette partie du clavier est souvent négligée, elle se révèle pourtant très pratique.

Module 10. Utilisation des menus

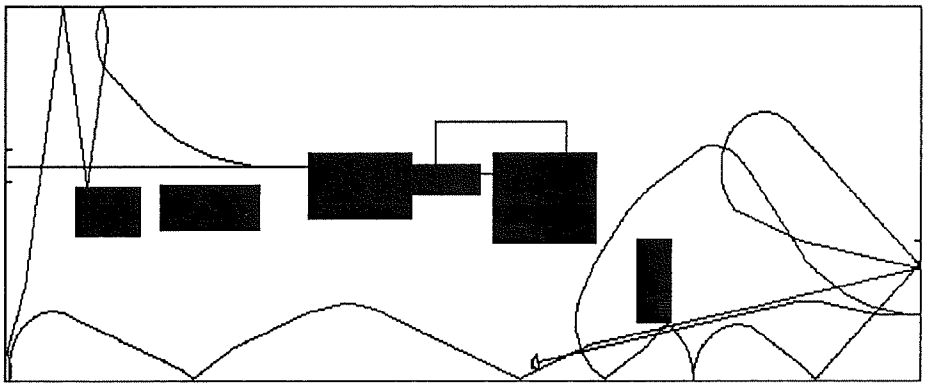
Ce module permet à l'utilisateur d'être confronté à plusieurs types de menus que

l'on retrouve dans de nombreux logiciels. Il met également en évidence la structure arborescente des menus.




```
Voici un exemple de menu qui comporte des options et des "sous-menus".
Il a une forme d'arbre renversé. Le menu principal se trouve en haut,
représenté par la lettre C.
```

Module 11. Test général



Exercice d'adresse: pilotage d'un mobile

Ce module permet de vérifier la bonne utilisation de quelques groupes de touches expliqués auparavant.

2.3 Options prises au cours du développement

Au départ, notre ligne de conduite était de réaliser un didacticiel entièrement paramétrable, mais nous savions pertinemment que cette ambition ne pouvait être approchée que de manière asymptotique. Le didacticiel devait également être varié, pour ne pas lasser, et utiliser les ressources à disposition sur un micro-ordinateur: son, graphisme, animation... Il fallait également une certaine rigueur dans la manière de présenter à l'utilisateur les messages d'aide ou d'erreur.

Dans le didacticiel tel que nous l'avons réalisé, l'utilisateur peut, à tout moment, abandonner le module en cours en appuyant sur une touche (<Esc>). Il peut également demander de l'aide lorsqu'une indication à l'écran signifie que cela est possible. Les abandons et les demandes d'aide sont comptabilisés.

Nous avons développé une infrastructure permettant de prendre en compte le facteur temps: temps de réponse de l'utilisateur, temps de parcours des modules... Le paramétrage et la configuration du didacticiel ont également fait partie de nos préoccupations.

2.4 Paramétrage et configuration du didacticiel

Programme de configuration du didacticiel

Nom du fichier : CLAVIER.PRM (valeur par défaut)

ordre	no. module	si pas réussi, suite	nbre répétitions	retour/suite
1	1	2	1	retour
2	2	4	1	retour
3	4	5	2	retour
4	5	6	1	retour
5	6	9	1	retour
6	9	10	1	retour
7	10	11	1	retour
8 ▸	11	12	1	retour
9	12	12	1	retour
10				
11				
12				

F1=enregistrer un fichier F2=lire cohérence F3=vérif. F4=insérer une ligne F5=effacer <return> modif. ↑ ↓ <ESC>

Paramétrage du didacticiel

Le paramétrage du didacticiel intervient à plusieurs niveaux. D'abord au niveau de la programmation. Certains courts-circuits d'utilisation du didacticiel étaient nécessaires durant la phase de développement pour des raisons de gain de temps. Par exemple, l'adjonction d'un mot ou d'un signe après le nom du programme permet, au moment du démarrage, d'éviter la partie introductive du didacticiel. Un certain nombre de ces facilités sont restées, mais ne sont accessibles qu'au niveau de la programmation ou grâce à des indications que seul le programmeur peut spécifier. De plus, certains calculs de performances ont été réalisés et essayés; ces dispositions subsistent, mais elles ont été désactivées pour des raisons qui transparaissent dans les points 6 à 8.

Il existe aussi pour l'enseignant la possibilité de modifier les textes d'aide et les messages d'erreur apparaissant lors de l'utilisation du didacticiel. En appelant le didacticiel par la commande **clavier prof**, les numéros des messages d'aide et d'erreur apparaissent. Ces numéros sont repris dans les programmes **makehelp** et **makemsg** permettant de modifier les textes correspondants. La généralisation, relativement simple, de cette facilité pourrait permettre d'adapter le didacticiel à des publics différents.

Enfin, nous avons réalisé un **pogramme de paramétrage** permettant à l'enseignant de choisir comment l'élève va parcourir les différents modules.

Première option: **parcours libre**

L'utilisateur choisit quels modules il veut utiliser, répéter ou éviter.

Deuxième option: **parcours séquentiel**

L'utilisateur doit passer successivement par tous les modules, dans l'ordre où ils apparaissent dans le menu.

Troisième option: **parcours paramétré**

C'est sans doute l'option la plus intéressante. Le parcours est pré-sélectionné par l'enseignant. Il lui est possible de choisir quels modules il veut faire parcourir et dans quel ordre. Il peut également choisir, en cas de performances non satisfaisantes dans un module, de renvoyer l'élève à un autre module (pour revoir une notion non assimilée) et, à choix, le faire revenir au module non réussi ou le faire poursuivre depuis le module où il a été envoyé. Cette dernière possibilité existe, mais n'est pas exploitée dans la forme actuelle du didacticiel.

Nous n'avons trouvé une telle approche dans aucun des didacticiels que nous avons eu l'occasion d'utiliser, et il nous semble que ce genre de paramétrage offre beaucoup d'avantages.

Le plus souvent possible, nous avons essayé d'introduire le hasard dans le didacticiel. Par exemple, nous avons utilisé un mini-dictionnaire de mots de plusieurs catégories (avec accents, sans accents, en majuscules...), ainsi que des phrases, des textes et des calculs choisis aléatoirement. Ce procédé évite l'accoutumance de l'utilisateur et le force à rester attentif; il ne peut pas réagir de manière machinale et répétitive. C'est, à notre avis, une des caractéristiques que devrait également présenter un didacticiel.

Dans certains cas, nous avons prévu des structures de programmation qui permettent de paramétrer les résultats obtenus, la diminution du nombre de questions en cas de réussite... Parfois des améliorations ont été réalisées, puis abandonnées; dans d'autres cas, elles sont uniquement aménagées en vue d'un développement futur.

3. FONCTIONNEMENT DU DIDACTICIEL

3.1 Point de vue de l'enseignant

Il est conseillé d'utiliser le didacticiel soi-même avant de le proposer à des utilisateurs (par exemple des élèves). Pour éviter un parcours désordonné de la part de l'utilisateur, il est préférable de faire appel au programme PARAM qui permet le paramétrage du parcours des différents modules.

3.2 Fichiers nécessaires

Voici les noms des fichiers nécessaires au bon fonctionnement du didacticiel:

Nom du fichier	description
COMMAND.COM	fichier DOS
AUTOEXEC.BAT	fichier de démarrage

Nom du fichier	description
KEYBSF.COM	driver de clavier suisse romand
HIRES.COM	programme résident de primitives graphiques
CLAVIER.COM	didacticiel proprement dit
CLAVIER.000	idem
CLAVIER.001	idem
PARAM.COM*	programme de configuration
CLAVIER.PRM	fichier de configuration du didacticiel
IMAGES	petite base de données d'images
MINUS.DIC	dictionnaire contenant des mots
TEXTE1.UDO	textes utilisés dans le menu
PHR.DIC	dictionnaire contenant des phrases
MAKEHELP.COM*	programme d'édition des message d'aide
SOS.SOS	fichier contenant les messages d'aide
MAKEMSG.COM*	programme d'édition des messages d'erreur
MSG.MSG	fichier contenant les messages d'erreur
MAKETXT.COM*	programme d'édition de textes divers
TXT.TXT	fichier contenant divers textes

* Programmes de paramétrisation du parcours et des messages

> Le programme HIRES.COM doit obligatoirement être chargé avant d'exécuter le didacticiel proprement dit. Le fichier KEYBSF.COM doit également être chargé pour que le didacticiel fonctionne correctement.

3.3 Installation sans réseau

Le bon fonctionnement du didacticiel sous-entend le respect des points suivants:

- tous les fichiers présents sur la disquette doivent être accessibles;
- le driver de clavier KEYBSF.COM doit être chargé avant l'entrée dans le didacticiel;

- le programme résident HIRES.COM doit être chargé avant l'entrée dans le didacticiel; en principe il ne doit être chargé qu'avant la **pemière** utilisation du didacticiel, il doit se trouver en mémoire vive;

- certains programmes résidents annexes peuvent interférer de manière négative sur les programmes chargés ultérieurement (entre autres le didacticiel);

- utiliser exclusivement un ordinateur OLIVETTI M24 (ou compatible) avec un clavier 102 touches; un clavier différent n'empêche pas le didacticiel de fonctionner, mais la correspondance de certaines touches n'est plus exacte. La résolution graphique doit être compatible avec le mode 640 x 400 points. Pour les versions du DOS supérieures à 3.0, GRAFTABL.COM doit être chargé.

La manière la plus simple d'utiliser le didacticiel est de le faire fonctionner indépendamment d'un réseau local. Après que l'ordinateur ait démarré, placer la disquette contenant le didacticiel dans le lecteur A:. Taper ensuite:

```
a:
cd clavier
hires
clavier
```

Dans le cas où le didacticiel doit être installé sur disque dur, il suffit de créer un répertoire sur le disque dur (C:):

```
c:
cd\
md clavier
cd clavier
```

et de copier le contenu de la disquette:

```
copy A:\clavier\*.*
```

Pour utiliser le didacticiel il suffit de se placer dans le répertoire clavier et de taper:

```
hires
clavier
```

Le programme HIRES ne doit être appelé que lors de la première utilisation du didacticiel durant une session.

3.4 Installation en réseau

L'installation du didacticiel sur un réseau local de type 10-NET peut être très facilement effectuée à l'aide de la disquette de reconfiguration du SIES. Il suffit de:

- placer la disquette dans le lecteur A: d'un poste de travail;
- taper **a:install** (le fichier INSTALL.BAT se trouve sur la disquette) et attendre la fin des opérations.

L'installation est automatique, les fichiers concernant le didacticiel sont placés dans le répertoire c:\clavier (existant ou créé automatiquement). De plus, le fichier de commande CLAVIER.BAT est placé dans le répertoire c:\bat. Pour faire démarrer le didacticiel, il suffit alors de taper **clavier**.

Il est également possible de désinstaller le didacticiel en exécutant la même séquence d'instructions que pour l'installation (voir ci-dessus), en remplaçant "install" par "desinsta".

4. PHASE DE DEVELOPPEMENT

4.1 Introduction

Voyons un peu plus précisément quelle a été la méthode suivie au cours de ce projet, quels ont été les obstacles rencontrés et quels sont les acquis.

4.2 Elaboration du scénario

L'élaboration du scénario du didacticiel s'est effectuée en plusieurs étapes. Une première esquisse a été établie au début du projet.

Puis, à la suite d'un séminaire de plusieurs jours donné, entre autres par Monsieur Fargette, nous avons affiné notre scénario afin de tenir compte des

une touche pour passer à la suite, il doit pouvoir le faire de la même manière dans tout le didacticiel.

5.3 Enchaînement des modules

En ce qui concerne l'enchaînement des différents modules, la difficulté majeure a été le choix de la chronologie des différentes parties. Dans notre cas, où le parcours peut être pré-établi d'une manière tout à fait libre, ce problème est légèrement atténué. Toutefois, l'ordre des modules dans le menu principal devait suivre une certaine logique. Or, est-il plus logique d'aborder les touches préfixes avant ou après les touches de fonction ou les touches accentuées ? Comme nous l'avons déjà dit, il faut parfois trancher, car un compromis est impossible.

Un autre problème s'est présenté: comment conclure un module ? Nous avons, dans un premier temps, mis en place une infra-structure statistique permettant d'afficher divers renseignements à la fin de chaque module. Ces renseignements comprenaient notamment le temps passé dans le module, le nombre de demandes d'aide, le numéro du module conseillé pour la suite (dans le cas d'un parcours libre), un score reflétant le taux de réussite du module. Dans un second temps nous avons décidé de supprimer cet écran récapitulatif à la fin de chaque module, estimant que, par exemple, le mode de détermination du score n'était pas élaboré de manière assez précise. Il nous paraît difficile, en l'absence d'une évaluation réelle et précise, d'extrapoler un score représentatif du taux de réussite ou d'échec d'un utilisateur. La plus grande difficulté réside dans des questions telles que: "Comment pondérer ?", "Comment tenir compte de l'historique du parcours et de l'utilisateur ?", "Comment faire intervenir le contexte ?"

5.4 Problèmes de "timing"

Il est difficile, en cours de développement du didacticiel, de se rendre compte du temps de réaction ou de réflexion nécessaire à un utilisateur dans différentes situations. Par exemple: vitesse de défilement d'un texte explicatif, temps maximum imparti pour terminer un test basé sur la vitesse d'exécution, etc. Dans la plupart des cas, nous basant sur notre dextérité, nous avions tendance à laisser trop peu de temps. A la suite d'expérimentations et de remarques nous avons pu ajuster les différentes temporisations.

5.5 Problèmes techniques

Les problèmes techniques les plus ardus ont été le contrôle intégral du clavier et la taille finale du didacticiel. Ces problèmes, connus bien avant la phase de programmation, ont été résolus, parfois au prix de restructurations.

La résolution du premier problème était une condition sine qua non de la réalisation du projet. A ce propos, notons un point essentiel pour la conception de didacticiels: certaines personnes préconisent de s'attacher, au début, exclusivement à la phase d'analyse et de conception, repoussant la phase de programmation en dernière priorité. Cette approche nous paraît dangereuse; elle peut constituer une cause de gaspillage d'énergie, de temps et donc d'argent. Nous reviendrons plus loin sur ce point.

La difficulté qu'il fallait surmonter, peu courante en matière de création de didacticiels, était d'**apprendre à utiliser un clavier d'ordinateur à l'aide précisément d'un clavier d'ordinateur**. Ce problème de récursivité, bien connu des informaticiens, devenait ici un cauchemar. Il fallut donc partir de l'hypothèse que l'utilisateur du didacticiel ignorait le fonctionnement d'un ordinateur et surtout de son clavier. Dans ces conditions, il devenait nécessaire de le guider progressivement et de trouver des subterfuges permettant de faciliter la communication.

Comme certaines touches des claviers d'ordinateurs servent à le mettre dans des états très différents, nous avons choisi de prendre directement et intégralement le contrôle de toute touche frappée. Cela peut paraître relativement simple, mais cette contrainte a nécessité toute une série de développements et de tests. Il n'est pas toujours aisé de savoir si l'utilisateur est, par exemple, en train d'appuyer sur la touche <Ctrl> ou sur une touche de majuscules, et ceci dans des contextes de programmation très différents.

Le second problème important qu'il s'agissait de résoudre était la taille relativement grande du didacticiel une fois terminé (environ 160 kilobytes sans aucun des fichiers annexes). Le langage utilisé pour le développement (Turbo Pascal version 3) ne permet de produire, d'un bloc, que des programmes d'environ 50 kilobytes. La technique de segmentation permit de pallier cette limitation. Toutefois, dans les cas où tous les segments utilisent une grande partie de code commun, le découpage est devenu très pointu. Plusieurs fois le didacticiel a dû être restructuré et redécoupé. Ce travail de restructuration n'est pas productif, mais prend beaucoup de temps; et il est très difficile de prévoir à priori la bonne organisation.

6. PHASE D'EVALUATION

6.1 Tests du didacticiel

Ce didacticiel a été testé par les auteurs dans des cours d'initiation à l'informatique (10ème degré), ainsi que par quelques autres personnes avec des élèves de niveaux très différents. Plusieurs questionnaires destinés aux personnes intéressées par une évaluation ont été réalisés et diffusés.

Le but de ces questionnaires était de recueillir des critiques, remarques ou commentaires, aussi bien en ce qui concerne le déroulement et le contenu du didacticiel qu'en ce qui concerne son fonctionnement (mise en évidence des défauts).

6.2 Collaboration externe

La collaboration externe s'est limitée à des contacts personnels pris par les auteurs, à une rencontre avec Monsieur Giordan (Université de Genève), à un entretien téléphonique avec un responsable du SRP et à des discussions avec des collègues enseignants. De toute évidence, il n'est pas toujours facile de recueillir des critiques concrètes. Par exemple la réalisation technique du didacticiel intéresse vivement le SRP, qui, par ailleurs, ne pense pas que ce sujet méritait d'être traité sous forme de didacticiel.

6.3 Comment tenir compte des résultats des tests

Pour tenir compte des résultats des tests, ceux-ci doivent être effectués dans la phase initiale de développement. De cette manière, en cas de fausse manoeuvre, il n'est pas nécessaire de tout recommencer. C'est précisément dans cette phase que nous avons mis en circulation une douzaine de disquettes de test à la demande de collègues (pour la plupart membres de la sous-commission UDO du GIDES). Or, seules deux ou trois réponses nous sont parvenues, après plusieurs semaines. Il ne nous était pas possible d'interrompre le développement dans l'attente des autres, qui ne nous ont d'ailleurs jamais été renvoyées.

Nous avons répété cette opération quelque temps plus tard, lorsque le didacticiel était quasiment terminé. Le taux de réponses fut cette fois un peu plus élevé.

6.4 Validité des tests

La validité des tests doit être établie sur une base statistique, or cette base nous a fait défaut, en partie à cause des problèmes exposés au paragraphe précédent. Ce sont donc plutôt, à proprement parler, des pré-tests que notre didacticiel a subis.

6.5 Diffusion du didacticiel en test

Nous pensons que le seul moyen de recueillir des remarques et critiques constructives pendant la réalisation d'un didacticiel est de faire appel à des personnes ayant vraiment le temps de le tester et de faire part de leur avis. De plus, s'agissant d'un véritable travail, ces personnes doivent impérativement être rétribuées. Un document, sous forme de questionnaire ouvert, doit accompagner le didacticiel en test. Des rencontres et discussions avec les personnes se prêtant à l'évaluation sont également indispensables.

Depuis peu de temps, la mise en place du Centre EAO, pourrait remplir de manière efficace ce rôle d'évaluation.

On n'oubliera pas, cependant, que l'évaluation d'un didacticiel fait naître une difficulté majeure. Idéalement, cette évaluation devrait s'effectuer de manière continue et parallèle à l'avancement du projet. Or, dans la pratique, il est difficile d'évaluer un produit non fini, et de faire abstraction des dysfonctionnements dûs à l'état de prototype du projet en se focalisant uniquement sur son fonctionnement correct présumé. En d'autres termes, il faudrait pouvoir évaluer un produit fonctionnel, donc terminé; mais l'évaluation sert, paradoxalement, à améliorer le produit, donc éventuellement à le modifier.

7. ECHOS, QUESTIONS ET REMARQUES

7.1 Diversité des réactions

Les réactions que nous avons recueillies concernant le didacticiel étaient très inégales quant à la forme, mais aussi quant au fond. Certaines remarques étaient très précises et circonstanciées. D'autres plus vagues ou trop utopistes.

Entre autres anecdotes intéressantes, nous avons reçu, par l'intermédiaire de leur maître, des impressions d'élèves ayant utilisé le didacticiel. Parmi les réponses on trouvait des phrases telles que: "C'était très intéressant", "On peut très bien apprendre à utiliser un ordinateur grâce à ce programme", "Oui, j'ai bien aimé ce programme", "Je n'ai rien compris à la partie avec la tortue"...

7.2 Quelques exemples de critiques

Voici, en vrac et à titre d'information, quelques unes des critiques reçues (à gauche les remarques, à droite nos réponses):

La démonstration de l'utilisation du menu au début du didacticiel est trop rapide.	*Nous l'avons ralentie.*
Le test consistant à guider une tortue autour d'obstacles est trop difficile et n'amène pas grand chose.	*Certains élèves apprécient ce test et il nous semble que l'utilisation des flèches de déplacement devient, grâce à lui, plus naturelle.*
Il est dommage de ne pas pouvoir revenir en arrière d'un caractère lors d'une faute de frappe dans le module où l'utilisateur doit reproduire des séquences de lettres et/ou de chiffres.	*Effectivement, mais c'est un choix que nous avons fait.*
Avec un écran couleur, certains textes (aide par exemple) ne sont pas très lisibles (texte blanc sur fond bleu ciel).	*Ayant développé ce didacticiel sur nos ordinateurs personnels (sans écran couleur) nous n'avions pas prévu ce problème, mais nous y avons remédié.*
La présentation animée au début du didacticiel trouble les élèves et n'est pas utile.	*Nous l'avons supprimée, bien que nous ayons passé pas mal de temps à la réaliser.*
Certains élèves sont bloqués pour le déplacement du petit bonhomme à amener dans sa maison.	*Ce point a été amélioré.*
Le logiciel "se plante" lorsqu'on déroule le menu plus loin que le dernier module prévu.	*Problème résolu.*

Le logiciel devrait s'assurer que KEYBSF.COM a bien été chargé ou devrait le charger automatiquement.

Il suffit d'appeler KEYBSF.COM dans le fichier .BAT qui lance le didacticiel.

Dans le module 1, le défilement des différentes parties du clavier est un peu trop rapide pour les élèves qui essaient de voir sur leur clavier ce qu'on leur présente à l'écran.

Nous avons généralisé l'emploi des trois petits carrés placés dans le coin inférieur droit de l'écran, indiquant à l'utilisateur qu'il doit appuyer sur une touche.

On pourrait mentionner "escape" lorsqu'on parle de la touche <Esc>.

Nous ne l'avons pas fait: <Esc> n'a pas toujours le sens de "escape".

Lorsque l'on doit taper sur une touche du pavé numérique, le logiciel accepte la touche 1, mais pas la touche 5.

Cette touche n'envoie aucun code en mode "Funct lock".

Pourquoi, dans la présentation des différentes touches du clavier, ne parle-t-on pas des touches spécifiques telles que <Clear>, <Break>, <Scroll on>, <Funct Lock>, <Scr prt>, <Help>, <Ins>, <Del>....

Les touches <Clear>, <Scroll on> et <Help> n'existent que sur les claviers OLIVETTI (et encore, uniquement M24). La touche <Help>, provoquant un faux espoir n'est utilisée dans quasiment aucun logiciel (si c'était le cas, il serait spécifiquement OLIVETTI M24). Les touches <Ins>, <Del> et <Funct Lock> sont expliquées dans les modules suivants.

Pourquoi ne pas présenter la deuxième touche <Return> ?

Nous avons pris en compte la seconde touche <Return>.

Lorsque le programme nous demande de taper sur une touche majuscule, il n'y a aucune interprétation de la touche <Caps Lock>. Pourquoi ne serait-elle pas acceptée ?

La modification permettant d'interpréter cette touche a été effectuée.

A plusieurs endroits dans le déroulement du didacticiel on ne sait pas exactement ce qu'on doit faire. La plupart du temps il faut faire <Return>, mais les trois petits carrés n'apparaissent pas en bas à droite. Par exemple, à la sortie du

L'utilisation des trois petits carrés a été généralisée.

labyrinthe, du billard ou lorsque le dessin du clavier est très bas (cf. 2ème module).	
Les trois petits carrés en bas à droite apparaissent parfois avec le curseur, parfois sans le curseur.	*Cette situation a été rendue cohérente.*
A un endroit, l'utilisateur doit écrire quelque chose dans une zone marquée par des soulignements; il n'est précisé nulle part qu'il faut taper des caractères jusqu'à la fin du soulignement. De ce fait, les élèves tapent <Return> et tournent en rond.	*Actuellement, le <Return> est accepté.*
La barre de division "/" n'est pas forcément connue des élèves; on pourrait rappeler brièvement le symbole des quatre opérations.	*Nous ne l'avons pas fait, ce symbole n'est pas si inconnu que ça des élèves.*
Pourquoi le pavé numérique est-il parfois appelé pad ?	*La terminologie est maintenant cohérente.*
Module 1: comment corriger le nom si un caractère faux est tapé ?	*On peut maintenant revenir en arrière.*
Il n'est pas expliqué comment déplacer le petit bonhomme.	*Explication ajoutée.*
Utilisation de mots inconnus (didacticiel, autorepeat, module...).	*C'est vrai, mais où est la frontière ? Les mots "préfixe", "alphanumérique", "simultanément"... sont-ils mieux connus ?*
Exercice avec calcul: faut-il recopier ou calculer ?	*Des explications ont été ajoutées.*
Comment sortir du deuxième jeu-test si on se "plante" ?	*La possibilité d'appuyer sur la touche <Esc> a été ajoutée.*
Comment un maître peut-il modifier les exercices et proposer des exercices plus longs et répétitifs ?	*Bonne proposition, mais trop complexe à mettre en oeuvre. Cela équivaudrait à proposer un vrai langage auteur.*
Allonger le temps d'arrêt entre certaines actions.	*Dans la mesure du possible cela a été fait.*

Module 4: c'est le symbole "obtenu" et non "choisi" qui clignote; dans le titre remplacer <Alt> + <Ctrl> par <Alt> + <Shift> ou <Maj>, mentionner que les touches de préfixe doivent être pressées.	*Nous avons corrigé le programme pour tenir compte de cette remarque.*
L'exercice d'évitement d'obstacles paraît un peu long.	*Nous avons ajouté la possibilité d'abandonner avec <Esc>.*
Les touches <F11> à <F18> et <CapsLock> sont restées sans effet.	*Problème résolu.*
Exercice de recopie de mots: la séquence de caractères reste trop longtemps avant de disparaître dans "juste" ou "faux".	*Nous avons accéléré cette partie.*
La comparaison entre un touche morte et un cadavre semble des plus douteuses.	*Eh oui! Sans fantaisie, que serait un didacticiel ?*
Jeu des proverbes: amusant, mais un peu long.	*Il a été écourté.*
Le didacticiel est prévu pour un élève, alors qu'on trouve généralement deux élèves par machine. Il faudrait une version pour deux élèves, avec gestion adéquate et compétition entre les deux.	*Cas typique de proposition impossible à implémenter. L'ordinateur n'a aucun moyen de savoir qui a réellement appuyé sur une touche. De plus, le clavier de quasiment tous les ordinateurs ne capte qu'une touche significative à la fois.*
Autorepeat: quoi que l'on fasse le message "Vous êtes un cas désespéré" apparaît.	*Problème non résolu, lié à la différence de la fréquence de l'autorepeat entre ordinateurs du même modèle.*
Module 6: il faudrait préciser dans le texte que gauche et droite font tourner de 45 degrés.	*Nous pensons que l'élève le remarque assez vite. De plus, il ne connaît pas forcément la notion d'angle.*
Il manque malheureusement le son.	*Certains maîtres apprécieront fortement ce manque lorsqu'une dizaine de didacticiels "clavier" fonctionneront dans la même salle.*
Vive le labyrinthe.	*Nous sommes de cet avis.*

7.3 Répercussions sur le didacticiel

Comme nous l'avons déjà mentionné, il est plus difficile de modifier un didacticiel achevé qu'un didacticiel en cours de développement. Ceci explique que nous n'avons pas pu intégrer toutes les remarques qui nous ont été transmises. Certaines, fort intéressantes, auraient demandé plusieurs mois de travail ou une restructuration trop importante. Il n'en reste pas moins que de nombreuses suggestions et critiques ont été prises en compte, ainsi qu'on a pu le constater au paragraphe précédent.

8. BUTS ATTEINTS ET BUTS NON ATTEINTS

8.1 Points négatifs

Le didacticiel n'est pas aussi indépendant du matériel que nous l'aurions souhaité. En particulier, le mode graphique utilisé est propre aux ordinateurs OLIVETTI. De plus le didacticiel concerne exclusivement le clavier suisse romand 102 touches OLIVETTI.

Nous n'avons pas trouvé de moyen efficace permettant d'apprécier la réussite d'un module, à l'aide d'une quantification sous forme de score. La structure pouvant aider à cette quantification est présente dans le programme, mais n'apparaît pas, car elle nous semble imparfaite. Il est difficile de quantifier les difficultés réelles surmontées ou non. Les fautes de frappe se comptent très facilement, mais elles ne reflètent en rien la compréhension. Enfin nous n'avons pas eu le temps, au moment opportun, de réaliser un ensemble d'outils statistiques de monitoring utilisables dans la phase de test.

8.2 Points positifs

L'objectif général du didacticiel nous semble avoir été atteint. Nous avons pu mettre en oeuvre et en pratique un certain nombre des caractéristiques visées. D'abord, le didacticiel peut effectivement être mis entre les mains de personnes qui abordent l'informatique et n'ont, de ce fait, pas une expérience de l'utilisation d'un clavier d'ordinateur. Ensuite, le fonctionnement est assuré aussi bien en réseau qu'en

mode indépendant (avec ou sans disque dur). Nous avons pu maîtriser la plupart des difficultés techniques spécifiques au sujet traité. La paramétrisation du didacticiel a pu être en partie réalisée, notamment pour ce qui concerne le contrôle du parcours. Sur ce point nous avons imaginé une méthode inédite, simple dans le concept qu'elle recouvre, mais que délicate à mettre en oeuvre. Nous pensons que la plupart des didacticiels devraient intégrer un tel contrôle de parcours. Le paramétrage du texte (aide, messages, etc.) affiché par le didacticiel a été en partie réalisé. Sa généralisation pose plus des problèmes matériels (taille et lourdeur du programme) que de conception. Enfin, nous avons pu incorporer à ce didacticiel des ressources, essentiellement logicielles, que nous avions développées hors du cadre de ce projet de recherche. Nous sommes heureux d'avoir réalisé un produit réellement utilisable et pas simplement une maquette ou autre recherche plus ou moins éloignée de la réalité. En effet, trop souvent, la phase de matérialisation d'un projet est occultée, oubliée, minimisée ou considérée comme intellectuellement dégradante.

8.3 Suggestions

Pour la réalisation d'un didacticiel, il vaudrait mieux disposer de plus de temps de manière continue, que de temps de travail très fractionné. La collaboration avec des personnes ayant des compétences dans le domaine traité est fortement souhaitable et un grand effort doit être fait en ce qui concerne l'évaluation du didacticiel.

9. REMARQUES GENERALES

9.1 Importance de la phase d'étude

Dans un projet de réalisation d'un didacticiel, la phase d'étude et d'analyse revêt une grande importance. Il convient d'y englober le plus grand nombre de problèmes à traiter, tant de conception que d'implémentation. C'est sans doute la phase la plus importante et de sa réussite dépend la qualité du résultat final. La conception des différents scénarios doit absolument être prise en considération dans les moindres détails.

A notre avis, il ne faut surtout pas exclure de cette phase les problèmes d'implémentation et de programmation. Un obstacle inattendu peut bloquer la

réalisation pratique et remettre en cause l'analyse elle-même en raison d'une impossibilité technique. Et un obstacle est d'autant plus difficile à surmonter qu'on ne l'a pas envisagé dans la phase d'analyse. Nous pensons donc qu'il est essentiel, pendant l'analyse, d'avoir constamment à l'esprit les problèmes techniques possibles. Cela signifie concrètement que des programmeurs expérimentés doivent participer à cette première phase de conception. Il n'est pas réaliste de développer un scénario en partant de l'a priori que tous les problèmes trouveront une solution lors de la réalisation pratique, cet a priori est faux.

9.2 Importance de la phase d'implémentation

Cette phase est la mise en pratique de l'analyse effectuée auparavant. Plus l'analyse et la description des scénarios sont précises et complètes, plus la structuration du didacticiel est aisée. Mais une analyse aussi parfaite soit-elle n'est pas d'une grande utilité pour la manière de programmer et pour le choix des algorithmes. En fait l'intersection entre les deux phases est relativement réduite.

9.3 Limites techniques

La réalisation d'un didacticiel est une tâche bien trop importante pour être confiée à des non-spécialistes de l'informatique ou à des personnes complètement étrangères à l'enseignement. Notre avis est qu'il n'existe pas (encore) d'assez bons environnements d'aide à la réalisation de didacticiels, grâce auxquels les non-informaticiens (c'est-à-dire des personnes ne connaissant pas la programmation et les différents problèmes liés à l'informatique) pourraient réaliser un didacticiel performant, efficace et atteignant tous les objectifs fixés.

Il est essentiel d'être conscient ou de subodorer dès le début les difficultés techniques potentielles. Parmi celles-ci, certaines sont surmontables, tandis que d'autres obligent parfois à revoir certains objectifs généraux ou même le scénario du didacticiel.

9.4 Adaptabilité et durabilité du didacticiel

Le problème de la maintenance d'un logiciel est une constante en informatique. Un didacticiel n'échappe pas à la contrainte d'offrir une maintenance aisée. Un

logiciel souple et de qualité devrait être dans la plus large mesure possible:

- indépendant de ses concepteurs;
- indépendant du type d'ordinateur sur lequel il est utilisé;
- indépendant des versions du système d'exploitation;
- indépendant des types d'écrans et de claviers;
- transposable dans une autre langue;
- indépendant des modes...

Notre didacticiel présente quelques lacunes sur ces différents points. Notamment il est très dépendant du type d'ordinateur et de clavier utilisé. Le graphisme en haute résolution (640 x 400 points) qu'il utilise est caractéristique des ordinateurs OLIVETTI, et le clavier physique qu'il simule est un clavier 102 touches. Son adaptation à d'autres types d'ordinateurs et/ou de claviers demanderait un travail important. Nous avons utilisé un programme résident spécifiquement conçu pour accéder au mode graphique haute résolution des OLIVETTI M24 depuis le TURBO Pascal version 3. Lorsque le didacticiel affiche un écran en mode graphique, toutes les coordonnées sont exprimées en valeurs absolues. L'adaptation à un autre type d'écran graphique impliquerait une révision complète de ces passages (nombreux). De plus, certaines images, par exemple la représentation complète des touches du clavier, ne peuvent pas être conçues et dessinées de la même manière avec un écran de résolution plus faible.

L'adaptation du didacticiel à un autre type de clavier pose également des problèmes, essentiellement à deux niveaux. D'abord, l'analyse de la lecture des touches (au niveau des "scan codes" et non des "code ASCII") devrait être revue et modifiée. Ensuite, la corrélation entre le clavier physique utilisé et sa représentation graphique à l'écran devrait être garantie.

Un détail met en évidence la spécificité du didacticiel sur un point particulier. Dans un des modules la connaissance de l'"autorepeat" est testée. Or, il se trouve que sur les modèles récents d'OLIVETTI M24 la fréquence de répétition des touches est légèrement différente. Remédier à ce problème, apparemment anodin, n'est pas immédiat et demanderait plusieurs heures de travail. Cet exemple montre que la maintenance est souvent rendue difficile par des détails.

10. DESCRIPTION TECHNIQUE

10.1 Choix du langage

Pour développer un didacticiel à l'aide d'un langage de programmation habituel, il paraît indispensable de choisir un langage offrant les éléments suivants:

- procédures et/ou fonctions;
- passage de paramètres (échange d'information entre différentes parties d'un programme);
- langage structuré (ce point est essentiel à tout développement informatique un tant soit peu cohérent);
- possibilité d'utiliser le graphisme (si possible avec une résolution acceptable);
- phase de compilation rapide permettant de gagner beaucoup de temps dans la phase de programmation;
- une certaine richesse de types de données;
- une bonne intégration dans l'environnement MS-DOS.

Parmi les langages potentiels, nous avons choisi le Turbo Pascal version 3, surtout parce que nous le connaissions déjà (avec ses qualité et ses points faibles). De plus, nous connaissions aussi quelques outils de développement prévus pour ce langage. Enfin nous avions déjà développé (à titre privé et indépendamment de ce travail de recherche) quelques utilitaires que nous avons pu reprendre. Un autre très bon langage auquel nous aurions pu faire appel est le Modula-2, le Turbo Pascal version 4.0 étant arrivé trop tard.

Pour des raisons dont certaines sont exposées au point 5.5, ce didacticiel ne pouvait être réalisé qu'à l'aide d'un langage de programmation traditionnel. A notre connaissance aucun langage auteur ne permet de résoudre complètement et efficacement, en plus des problèmes banals, des problèmes comme le contrôle intégral du clavier. Nous avons, à l'occasion de ce travail de recherche ou en d'autres occasions, assisté à des démonstrations de langages auteurs divers, et nous avons été étonnés de constater la pauvreté de la plupart d'entre eux. Il est dommage de développer des didacticiels à l'aide de langages auteurs n'offrant pas toutes les facilités que l'on trouve couramment dans les progiciels du commerce. S'il est un

domaine de la programmation où les plus grandes précautions doivent être prises, c'est, entre autres, l'EAO. La qualité à verser à l'actif des langages et systèmes auteurs est leur facilité d'emploi.

10.2 Choix des outils

La plupart des outils entrant dans la composition du didacticiel ont été développés par nous-mêmes. Nous avons cependant fait appel à des parties de produits commercialisés, notamment pour la gestion de fenêtres superposables.

10.3 Organisation et modularité

Le didacticiel est organisé autour d'un module principal incluant tous les autres. Chaque module est une "overlay procedure". Dans chaque module, l'entrée et la sortie sont prises en charge par des procédures qui déterminent, entre autres, le temps d'utilisation du module courant, le nombre de demandes d'aide, le taux de réussite du module. Le module principal détermine, selon le mode de paramétrage du parcours (voir 2.4), la manière dont le menu principal doit être utilisé, et gère le parcours.

La modularité est intégrée de manière à permettre facilement l'adjonction de nouveaux modules ou le remplacement de modules existants.

10.4 Quelques chiffres

Pour les amateurs de chiffres, en voici quelques uns permettant de cerner le produit de manière technique:

Taille du programme compilé:
environ 160'000 bytes.

Taille totale des fichiers utilisés par le didacticiel:
environ 300'000 bytes.

Taille totale des fichiers de travail durant le développement:
approximativement 1'500'000 bytes.

Nombre de lignes total du didacticiel (sans les programmes de test et de développement):

environ 12'000 lignes.

Nombre de fichiers (contenant des programmes) utilisés dans la phase de développement:

plusieurs centaines.

Nombre de modules:

11.

Nombre d'overlays (didacticiel seul):

2 grandes zones d'overlays,
17 grandes procédures en overlay.

Nombre de procédures et fonctions (didacticiel seul):

environ 450.

Temps de compilation du didacticiel seul (sans les programmes annexes):

env. 35 secondes sur un ALR/386 (20 Mhz) avec disque rapide;
env. 10 à 15 fois plus sur un OLIVETTI M24.

Il convient de préciser que ces chiffres sont purement indicatifs et n'ont aucune autre valeur.

11. PARTIES REUTILISABLES DU DIDACTICIEL

Ce didacticiel contient beaucoup de parties de code que l'on peut qualifier d'"utilitaires". Nous donnons ci-dessous une liste des principaux outils qui peuvent être repris tels quels ou adaptés à d'autres programmes ou didacticiels.

Affichage d'images

- à partir d'images (bitmap) stockées sur disque, nous avons utilisé des procédures permettant de les afficher à un emplacement donné de l'écran.

Menu

- le menu général du didacticiel comporte la possibilité de défilement vertical permettant de présenter plus d'options que la place disponible ne l'autorise;
- chaque option est expliquée par un texte au bas de l'écran.

Gestion des messages

- les programmes "makehelp" et "makemsg" permettent de créer des textes d'aide, des messages d'erreur ou autres;
- l'utilisateur peut créer et modifier les différents textes (repérés par un numéro) qui seront ensuite affichés depuis un programme;
- les textes sont placés dans un fichier.

Autres menus

- nous avons "imité" le fonctionnement de plusieurs menus que l'on retrouve dans des logiciels existants;
- ces menus peuvent être repris et utilisés dans n'importe quel programme.

Tortue

- un des modules présente un jeu basé sur le pilotage d'une tortue;
- les routines nécessaires à la gestion des déplacements sont disponibles, et peuvent être reprises dans d'autres programmes ou situations.

Paramétrage

- le programme de paramétrage "param" contient les routines de base permettant différents modes de paramétrage du didacticiel;
- ce paramétrage peut être utilisé dans n'importe quel programme dans lequel l'utilisateur doit ou peut être guidé.

Saisie d'information

- la saisie d'information à l'écran pose souvent des problèmes de sécurité et donc de programmation;
- notre module de saisie permet une saisie de plusieurs types de données (entiers, réels, chaînes de caractères...) en toute sécurité;

- ce module passe-partout peut rendre des services dans tous les programmes.

Editeur

- nous avons développé un petit éditeur de texte dont l'avantage est d'être très compact et relativement facile à paramétrer.

Outils divers

- routines de saisie du temps du système et de calcul de durées;
- création de sons divers;
- routines sophistiquées permettant de déterminer les touches appuyées (toutes);
- sauvegarde et reprise d'une partie de l'écran (graphique ou texte);
- routines de déplacement de chaînes de caractères (dans deux directions);
- ...

REMERCIEMENTS

Nous tenons à remercier Monsieur Bugniet, responsable du Dispositif de recherche de l'enseignement secondaire, ainsi que toutes les personnes ayant contribué à la réalisation de ce travail de recherche.