

## OPTION INFORMATIQUE

### INTRODUCTION D'UN ENSEIGNEMENT OPTIONNEL COMPLÉMENTAIRE D'INFORMATIQUE EN CLASSES DE SECONDE, PREMIÈRE ET TERMINALES DES LYCÉES

Article premier. - L'annexe III des "enseignements optionnels complémentaires" de l'arrêté du 31 octobre 1980 cité ci-dessus est modifié comme suit :

.....  
- Informatique 1 + (1,5)

Art. 2. - Le tableau n° 4a des "enseignements optionnels complémentaires" annexé à l'arrêté du 29 décembre 1981 cité ci-dessus est modifié comme suit :

.....  
- Informatique 1 + (1,5) pour toutes sections

Art. 3. - L'annexe I du présent arrêté précise et énumère les conditions qui doivent être obligatoirement remplies afin que l'enseignement optionnel complémentaire d'informatique soit ouvert dans un lycée.

Art. 4. - L'enseignement optionnel complémentaire informatique est dispensé pour la classe de seconde, conformément au programme annexé (annexe II) au présent arrêté.

Art. 5. - Les dispositions du présent arrêté entrent en application à la rentrée scolaire 1985 pour les classes de seconde, à la rentrée scolaire 1986 pour les classes de première, à la rentrée scolaire 1987 pour les classes terminales.

Art. 6. - Le directeur général des Enseignements scolaires et le directeur des Lycées sont chargés de l'application du présent arrêté qui sera publié au Journal officiel de la République française.

Pour le ministre et par délégation :

Le directeur des Lycées,

P. ANTONMATTEI

(J.O. du 6 juin 1985)

### ANNEXE I

**Condition d'ouverture d'une option  
informatique en second cycle long en  
1985-1986**

#### 1 - Personnels enseignant l'option

##### *1.1. Constitution de l'équipe enseignante :*

L'enseignement de l'option peut être assuré par des professeurs de toutes disciplines ; ils ont l'informatique comme seconde compétence et travaillent en équipe.

Vu L n° 51-46 du 11-1-1951 mod par D n° 70-620 du 10-7-1970 . L. n° 75-620 du 11-7-1975 ; D. n°59-57 du 6-1-1959 mod not par D n° 65-438 du 10-6-1965 et n° 68-639 du 9-7-1968 ; D. n° 62-1173 du 29-9-1962 mod rot par D. n° 83-359 du 4-5-1983 ; D n° 76-1304 du 28-12-1976 ; A 31-10-1980 A première Arrêtés 29-12-1981 ; A. 9-3-1982 ; avis du conseil de l'enseignement général et technique du 24-1-1985.

R.L.R. : 524-0 d. 0 e. 0 f : 524-6 : 524-7

Arrêté du 31 mai 1985

(Éducation nationale : bureau DL 3)

Pour que l'enseignement puisse être correctement assuré aux différents niveaux, l'équipe enseignante doit comprendre au moins deux professeurs la première année, et être portée à 3 au moins à la rentrée suivante. Ces enseignants n'appartiendront pas tous à la même discipline ; ils auront l'expérience des usages possibles de l'informatique dans les diverses disciplines enseignées aux élèves.

##### *1.2. Service des enseignants*

Compte tenu des objectifs de cet enseignement optionnel, le professeur qui le dispense ne l'assurera que pour moitié, au

plus, de son maximum de service ; pour l'autre moitié il enseignera sa discipline.

### **1.3. Conditions de formation :**

La formation requise est celle assurée par les stages longs d'une année ou une formation universitaire de second cycle en informatique, à laquelle s'ajoute une formation complémentaire orientée vers la didactique de l'informatique.

Cette formation complémentaire doit donc être organisée, dans le cadre des plans académiques de formation continue des personnels enseignants, en fonction des besoins.

Ses objectifs essentiels seront de permettre aux professeurs en ayant bénéficié de pouvoir enseigner les contenus et réaliser les objectifs des programmes définis dans l'annexe II.

Tous les professeurs enseignant l'option dans l'académie devront obligatoirement avoir suivi cette formation complémentaire ou, à défaut, commencer à la suivre la première année d'enseignement de l'informatique.

Les contenus de cette formation prennent comme base ceux qui ont été définis en concertation avec le comité scientifique national et les responsables universitaires du suivi de l'option. Ils seront régulièrement actualisés pour tenir compte du caractère évolutif de l'informatique, tous les enseignants de l'option devant pouvoir bénéficier de ces actualisations.

## **2. Suivi scientifique**

Le centre universitaire assurant ici formation complémentaire au niveau académique doit s'engager à assurer le suivi scientifique de l'enseignement dans les établissements.

## **ANNEXE II**

### **A - Objectifs du programme d'enseignement de l'enseignement optionnel complémentaire d'informatique en second cycle long**

L'option informatique en second cycle long est un enseignement à caractère général. Il n'est pas orienté vers un type particulier

d'application ou de profession. Il vise les trois grands objectifs suivants :

#### **- Apport de connaissances techniques :**

Le matériel informatique (composants, architecture, fonctionnement) et le logiciel (tangage de programmation, systèmes, traducteurs, progiciels) seront présentés de manière précise ; mais les notions acquises seront suffisamment générales pour que les élèves puissent suivre l'évolution des techniques.

#### **- Apprentissage de méthodes de travail:**

L'informatique sera replacée dans le cadre d'un processus général de résolution de problèmes, où l'on distingue et approfondit différentes étapes :

- énoncé d'un problème ;
- découverte et expression d'un procédé de résolution
- codage dans un langage de programmation
- validation du programme ainsi obtenu ;
- utilisation effective pour l'application retenue.

Cette démarche met en jeu des qualités universelles d'analyse, de synthèse, de rigueur d'expression et d'efficacité.

L'intérêt de l'informatique est d'aider à les révéler et à les consolider chez les élèves grâce à son caractère constructif, concret, et à sa rapidité de calcul.

Par ailleurs, la recherche systématique d'une méthode de résolution, l'analyse des causes d'erreur dans ses réalisations, développeront chez l'élève le goût d'une certaine autonomie.

Un autre intérêt de l'informatique est de favoriser le travail en équipe : chaque groupe d'élèves, pour pouvoir réaliser la tâche qui lui incombe dans un projet global, prendra conscience de l'importance de la coordination entre les différentes tâches, de la communication entre les différents acteurs.

#### **- prise de conscience des enjeux économiques, sociaux et culturels :**

L'informatique est désormais bien ancrée dans la vie économique, sociale et

culturelle. Elle touche toutes les professions où elle peut être considérée comme un outil de travail indispensable. L'option informatique permettra d'envisager toutes les dimensions de ce phénomène par une étude de l'évolution de l'informatique, de ses grandes applications et de ses produits, des intérêts mis en jeu, de ses métiers et de sa gestion, ainsi que des importantes questions éthiques et juridiques qu'elle suscite. L'option informatique en second cycle long est un enseignement à caractère général. Il n'est pas orienté vers un type particulier d'application ou de profession. Il vise les trois grands objectifs suivants :

## **B - PROGRAMME D'ENSEIGNEMENT DE LA CLASSE DE SECONDE**

Le programme qui suit comporte des notions relevant de ces trois aspects. L'ordre de leur présentation est laissé à l'appréciation du professeur. Il fixe les points qui doivent être abordés en seconde, mais n'est pas limitatif. Quelques idées de présentations ou d'approfondissements possibles y sont suggérées, ainsi que dans les commentaires.

On gardera présent à l'esprit que l'atteinte des objectifs n'est pas subordonnée à une accumulation de connaissances abstraites ou techniques et qu'il est essentiel que les élèves puissent mobiliser effectivement les connaissances acquises lors de la réalisation de projets, en réalisant le passage progressif du concret vers l'abstrait, et réciproquement, qui est aujourd'hui une dimension essentielle de toute formation. L'informatique est une discipline en pleine évolution. Son enseignement doit être constamment adapté.

### **1 - Informatique générale**

L'enseignement doit permettre à l'élève de comprendre le fonctionnement général d'un micro-ordinateur et d'apprendre à l'utiliser, sans entrer dans le détail technique des

supports, des codages, des composants, de l'architecture et du logiciel.

#### *1.1. - L'information*

Les données simples et leurs représentations internes et externes. Notions de numération. Exemples des nombres, caractères, chaînes.

#### *1.2. - Le matériel*

Les grands composants d'un micro-ordinateur et leurs fonctions : mémoire centrale, mémoires auxiliaires, organes d'entrée et de sortie, organe de calcul, organe de commande.

#### *1.3. - Le logiciel*

Nécessité de programmes pour commander l'ordinateur. Fonctionnement général d'un micro-ordinateur.

Notions de systèmes d'exploitation, d'éditeur de textes, de traducteurs de langages juste nécessaires à la mise en oeuvre et à la sauvegarde des programmes. Apprentissage de quelques commandes.

Pour illustrer le fonctionnement d'un micro-ordinateur, on pourra simuler l'exécution d'un petit programme en "jouant à la machine". On mettra ainsi en évidence les états successifs de la mémoire, les actions des différents organes et les échanges de messages avec l'organe de contrôle.

### **2 - Analyse et programmation :**

L'enseignement doit mettre l'élève à même de :

- lire un texte et spécifier clairement le problème à résoudre ;
- proposer et formuler avec précision une méthode de résolution ;
- la rédiger dans un langage de programmation ;
- la mettre en oeuvre sur un ordinateur.

Pour cela, le professeur développera des exemples typiques, en insistant sur les aspects méthodologiques. En travaux pratiques, les élèves pourront reprendre ces exemples et traiter, de façon analogue, d'autres problèmes de difficultés comparables. On donnera des conseils de rédaction facilitant la construction des énoncés, l'élaboration de méthodes de

résolution, de programmes, et permettant leur bonne lisibilité :

- Structuration en paragraphes ou modules ;
- Spécification de ces unités en français et insertion de commentaires correspondant dans le programme ;
- Choix d'identificateurs significatifs ;
- Construction progressive de lexiques d'identificateurs.

#### 2.1. - *Traitement séquentiel*

- Type et identification des objets élémentaires ;
- Définition d'un objet par une expression simple
- Instruction d'affectation ;
- Lecture des données et écritures des résultats.

#### 2.2. - *Traitement conditionnel*

- Définition conditionnelle d'un objet ;
- Instruction conditionnelle ;
- Expression logique élémentaire ;
- Expression logique générale avec connecteurs logiques.

#### 2.3. - *Traitement itératif simple*

- Définition de listes de données et de résultats ;
- Instruction itérative simple ;
- Notion d'indice contrôlé ;
- Notion de récurrence simple et de variable.

#### 2.4. - *Traitement itératif général*

(première approche)

(on ne sait pas calculer à l'avance le nombre de répétitions, mais on connaît une condition d'arrêt)

- Définition et instruction itératives générales ;
- Condition d'arrêt.

Il est recommandé, quand le matériel le permet, de faire traiter des informations de natures variées telles que nombres, chaînes, dessins, sons.

Les notions de tableau, de fichier de données, de procédures et de fonctions sont au programme de première. Elles pourront

être introduites prudemment pour certaines applications.

### 3 - *L'informatique dans la société*

L'enseignement doit aider l'élève à percevoir la place et l'utilisation passées, présentes et futures de l'Informatique dans la société, ainsi que ses possibilités, ses contraintes et ses limites.

#### 3.1. - *Grandes applications de l'informatique*

Présentation de quelques exemples d'applications de l'informatique dans la vie courante, telles que : gestion, calcul scientifique, banques de données, simulation, bureautique; robotique, enseignement, vidéotex.

#### 3.2. - *L'informatique-outil*

Utilisation concrète par les élèves de logiciels tout faits, tels que programmes de gestion, simulateurs, didacticiels, traitement de taxies, ludiciels, messageries, annuaire électronique. Point de vue critique d'utilisateurs.

#### 3.3. - *Histoire et évolution de l'Informatique*

Les grandes étapes de l'évolution des matériels, des logiciels, des méthodes, des applications et des professions de l'informatique.

#### 3.4. - *Professions de l'informatique*

Pour chaque point abordé dans le programme, on évoquera les professions correspondantes.

Ces éléments peuvent être apportés par une participation des élèves à des activités diversifiées telles que conférences, films, émissions de télévision, analyses d'articles, exposés d'élèves, visites d'entreprises, ateliers sur micro-ordinateurs.

## C - **Commentaire du programme de la classe de seconde**

### 1 - *Informatique général*

Il faut faire comprendre aux élèves que la même information peut être codée de différentes manières selon le support considéré. Mais on évitera les longs et difficiles

développements sur la représentation binaire en se limitant par exemple aux entiers positifs (pour éviter la complémentation à 2), aux réels en format fixe, aux caractères (codes ASCII ou EBCDIC) et aux chaînes.

De même, s'il est important que les élèves comprennent le rôle et le fonctionnement des différents organes d'une machine, on se gardera d'entrer dans les détails techniques de l'architecture ou de l'étude des composants électroniques.

Enfin, il n'est pas souhaitable d'étudier en détail toutes les possibilités et les commandes d'un système d'exploitation ou d'un éditeur particulier. On extraira et présentera seulement un petit noyau de notions et de commandes permettant les manipulations simples que nécessite la mise en oeuvre, sur les micro-ordinateurs de l'établissement, des programmes utilisés en classe de seconde.

### 2 - Analyse et programmation

Un des objectifs essentiels de l'enseignement optionnel de l'informatique est de former l'esprit et d'apporter des méthodes de travail.

On attachera donc la plus grande importance à l'analyse du problème, préalable à toute programmation.. On évitera qu'elle ne soit considérée par les élèves que comme une paraphrase en français courant du programme à, écrire : elle doit évidemment précéder la programmation et non pas la suivre, comme on le voit quelquefois. Ainsi que le recommandent aujourd'hui les professionnels de l'informatique, il faut proscrire l'usage de l'organigramme lors de la phase d'analyse, au profit d'une expression plus structurante. En revanche, un organigramme peut être construit après l'écriture du programme pour suivre les chemins de calculs possibles. Il peut servir aussi à préciser l'effet (la sémantique) d'une instruction, ainsi qu'à débrouiller un programme mal écrit.

Dans le libellé du programme, on entend par objet une donnée, un résultat intermédiaire ou un résultat.

On pourra apprendre aux élèves à reconnaître le type de définition d'un objet (simple, conditionnelle, itérative) avant de préciser davantage son algorithme de calcul.

On notera ensuite que dans les algorithmes séquentiels ou conditionnels, on peut associer biunivoquement un identificateur et une valeur, et il est conseillé de le faire.

Exemple : Lecture de trois notes et affichage de la moyenne pour deux élèves.

Algorithme	Lexique
Lire A1, A2, A3	A1, A2, A3
	Notes du 1er élève
$MA = (A1 + A2 + A3)/3$	MA : Moyenne du 1 <sup>er</sup> élève

Afficher MA.	
Lire B1, B2, B3	B1, B2, B3
	Notes du 2 <sup>o</sup> élève
$MB = (B1 + B2 + B3)/3$	MB : Moyenne du 2 <sup>o</sup> élève

Afficher MB.

Dans les algorithmes itératifs, un identificateur peut être associé successivement à plusieurs valeurs, en fait à une suite de valeurs : on parle de variable.

Exemple : Lecture de trois notes et affichage de la moyenne de ces notes, pour les trente élèves d'une classe.

Algorithme	Lexique
Répéter 30 fois	
Lire N1, N2, N3	N1, N2, N3
Notes d'un élève	
$M = (N1 + N2 + N3)/3$	M : Moyenne d'un élève

Afficher M

Fin répéter.

Les identificateurs N1, N2, N3, M désignent successivement les notes et la moyenne du 1<sup>er</sup>, 2<sup>o</sup>, ..., ième .... 30<sup>e</sup> élève. Si on s'intéresse au ième terme de la suite N1 (l nota N1 du ième élève), on remarque qu'il est obtenu par lecture et qu'il est totalement indépendant des termes précédents de cette suite.

Il y a enfin des cas où ième terme de la suite associée à une telle variable dépend du ou des précédents : on parle alors de récurrence.

Exemple : Lecture de trois notes, affichage de la moyenne de chaque élève, puis de la moyenne générale de la classe.

Algorithme Lexique

SG = 0

Répéter 30 fois

Lire N1, N2, N3

N1, N2, N3

Notes d'un élève

$M = (N1 + N2 + N3)/3$  M : Moyenne d'un élève

Afficher M SG : Somme, générale

$SG = SG + M$  MG : Moyenne générale de la classe

Fin répéter

$MG = SG/30$

Afficher MG.

Pour SG le calcul fait à la ième étape est en fait celui de :

$SG = SG + M$ . La valeur affichée en fin d'algorithme est celle du dernier terme calculé de la suite associée à MG.

Cet enseignement ne doit en aucun cas se limiter à un cours de langage de programmation, les grandes idées qu'il faut faire acquérir étant indépendantes de celui-ci, l'apprentissage, indispensable, d'un langage doit éviter une accumulation ponctuelle de détails syntaxiques ou techniques. Il doit être fractionné et suivre la progression des niveaux d'analyse des problèmes. Il est recommandé que les élèves aient, à la fin de la terminale, utilisé plusieurs langages de programmation. Pour éviter les risques de confusion, il est conseillé de se limiter à un seul langage en seconde.

Pour chaque type de définition, on proposera aux élèves un ou plusieurs schémas de traduction systématique dans le langage étudié, traduction qui sera d'autant plus facile que le langage choisi sera mieux structuré. Cette façon de procéder favorisera l'utilisation ultérieure d'autres langages. On insistera sur la nécessité de constituer des dossiers de programmation clairs et bien documentés retraçant les différentes étapes du processus de résolution des problèmes. On discutera le rôle et la place des commentaires à insérer dans le texte du programme en liaison avec ce dossier. On

veillera à ce que ce dossier puisse être lu et exploité par d'autres personnes que son auteur.

Enfin, s'il est indispensable que les notions nouvelles soient présentées sur des exemples d'école, il est souhaitable, dès la classe de seconde, de faire travailler les élèves sur un projet réaliste, de taille réduite, aboutissant à un petit programme.

### **3 - L'informatique dans la société**

Comme le suggère le programme de l'option, la présentation des implications de l'informatique peut revêtir des formes très variées. Il est tout à fait souhaitable qu'elle soit l'occasion de travaux pluridisciplinaires avec d'autres professeurs. Elle doit être liée aux autres thèmes de l'option et peut, par exemple, servir de point de départ au petit projet mentionné plus haut.

Selon les possibilités locales, les élèves utiliseront au moins un logiciel de traitement de textes, tableur ou système de gestion de fichiers.

En même temps que l'on montrera l'utilité d'une construction raisonnée et précise des programmes, il sera bon d'apprendre aux élèves à évaluer les qualités de ce qu'ils ont construit : la correction du programme est évidemment la qualité essentielle ; on peut essayer de s'en convaincre par l'élaboration de jeux d'essai. Mais il sera utile également de parler aux élèves de problèmes de temps de calcul et de place nécessaire en mémoire, de montrer qu'un problème peut avoir plusieurs solutions n'ayant pas les mêmes performances, sans toutefois se lancer dans un cours sur la complexité des algorithmes. L'option informatique doit s'insérer dans l'établissement et dans le cursus des élèves. Elle doit établir des liens avec toutes les formes d'utilisation de l'informatique, tant à l'intérieur de l'établissement (informatique dans les disciplines de gestion de l'établissement, Centre de Documentation et d'Information), que dans son environnement local.

B-O-n°25 20 juin 1985

## PROGRAMME DE PREMIÈRE

L'arrêté du 31 mai 1985 a introduit dans les lycées un enseignement optionnel complémentaire d'informatique. Publié au B.O. n° 25 du 20 juin 1985, il précisait les conditions d'ouverture d'une option, les objectifs, les programmes et commentaires pour la classe de seconde.

Cet arrêté est complété par l'annexe suivante concernant la classe de première, publiée au *B.O. n° 13 du 3 avril 1986 (pp 1243-1245)* sous le titre Objectifs, programme et commentaires de l'enseignement optionnel complémentaire d'informatique en première.

### I - OBJECTIFS SPÉCIFIQUES

L'enseignement d'option informatique en classe de première fait suite à celui dispensé en classe de seconde. Il s'adresse à des élèves de toutes sections et séries. Les objectifs spécifiques de la classe de première s'inscrivent dans les objectifs généraux définis précédemment.

#### 1 - Apprentissage de méthodes de travail

L'informatique est replacée dans le cadre d'un processus général de résolution de problèmes, où l'on distingue et approfondit différentes étapes :

- construction de l'énoncé précis du problème,
- découverte et expression d'un procédé de résolution, . codage dans un langage de programmation,
- validation du programme ainsi obtenu,
- utilisation effective pour l'application retenue.

Cette démarche met en jeu des capacités d'analyse, de synthèse, de rigueur d'expression et d'efficacité. L'intérêt de l'informatique est d'aider à les révéler et à les consolider chez les élèves.

La recherche systématique d'une méthode de résolution, l'analyse des causes d'erreur dans ses réalisations, développent en outre chez l'élève le goût d'une certaine autonomie.

Simultanément, l'informatique favorise le travail en équipe : chaque groupe d'élèves, pour pouvoir réaliser la tâche qui lui incombe

dans un projet global, prend conscience de l'importance de l'agencement des différentes tâches, de la communication entre les différents acteurs.

## **2 - Apport de connaissances techniques et technologiques**

En effet, un ensemble limité mais solide de connaissances techniques est indispensable pour comprendre le fonctionnement du complexe machine-logiciel. Les notions présentées sont suffisamment générales pour que les élèves puissent suivre l'évolution des techniques.

## **3 - Prise de conscience des conséquences et des enjeux économiques, sociaux et culturels de l'informatique**

Les élèves doivent percevoir les principaux aspects du phénomène social et culturel que représente l'informatique : ses grandes applications, ses métiers, son évolution, ainsi que les importantes questions juridiques - voire éthiques - qu'elle suscite.

## **II - PROGRAMME D'ENSEIGNEMENT ET COMMENTAIRES POUR LA CLASSE DE PREMIÈRE**

Le programme qui suit présente un ensemble de notions qui doivent être abordées avec tous les élèves de première de cette option. L'ordre adopté ne préjuge en rien celui que retient le professeur dans la conduite de son enseignement. La plupart du temps, ces notions se dégagent de la résolution de problèmes liés aux intérêts des élèves et aux programmes des autres enseignements.

L'atteinte des objectifs n'est pas subordonnée à une accumulation de connaissances abstraites ou techniques. Il est indispensable que les élèves puissent mobiliser effectivement lors de la réalisation de projets les connaissances qu'ils ont acquises.

### **1 - Méthodes et outils d'analyse et Commentaires de programmation**

Les objectifs de cette partie sont d'amener les élèves à spécifier, analyser un problème, puis programmer, et mettre en oeuvre la solution retenue.

On reprend la démarche utilisée en classe de seconde en intégrant de nouveaux outils de résolution.

### **1.1. Méthodes**

- Décomposition de problèmes en sous-problèmes, analyse modulaire.
- Choix d'organisation de données, en fonction des opérations qui leur seront appliquées.

### **1.2. Description des traitements**

- Rappel et compléments sur les traitements étudiés en Seconde (séquentiel, conditionnel, itératif).
- Imbrication de ces traitements,
- Sous-programmes : fonctions, procédures. Notion de paramètre donnée, de paramètre résultat.

### **1.3. Structuration de données**

- Les tableaux (1 et 2 dimensions),
- Les fichiers à accès séquentiel et direct : création et suppression d'un fichier, adjonction et suppression d'un article, mise à jour.

Les élèves peuvent continuer à utiliser le langage pratiqué en 2de.

## **Commentaires**

### **1.1. et 1.2.**

L'élève doit être capable :

- de décomposer un problème à solution algorithmique simple en modules et sous-modules qui réalisent chacun une fonction précise ;
- puis de traduire ces modules en programme et sous-programmes lors de la phase de programmation. Le type booléen peut être introduit pour faciliter une analyse : l'utilisation d'une variable booléenne permet de donner un nom à une expression à valeur logique un peu longue et dont l'évaluation se répéterait.

\* On évitera l'excès de technicité à propos de passages de paramètres.

\* L'étude de la récursivité figurera au programme de la classe de terminale.

Il est recommandé de leur faire manier un nouvel outil logiciel au cours de l'année, par exemple un tableur ou un gestionnaire de données.

## **2 - Matériel-logiciel**

L'enseignement vise uniquement à donner aux élèves des références sur le fonctionnement d'un micro-ordinateur et sur la représentation en mémoire de données structurées. Le cours peut donc être bref (de l'ordre de deux heures dans l'année au total). Des compléments éventuels, nécessaires à la réalisation de projets spécifiques peuvent être présentés.

### ***2.1. L'information***

Représentation en mémoire de données structurées : tableaux, fichiers séquentiels et à accès direct

### ***2.2. Le matériel et le logiciel***

Notions sur l'architecture fonctionnelle des ordinateurs, schéma de circulation de l'information, en prenant pour exemple le micro-ordinateur utilisé dans la classe et ses périphériques.

Notion de cycle d'exécution d'une instruction.

Commandes et programmes utilitaires d'un système d'exploitation.

## ***Commentaires***

### ***2.1.***

A l'occasion de l'étude des tableaux et des fichiers, l'élève doit comprendre que la mémorisation et la recherche ultérieure de données structurées demandent une organisation précise du support physique : exemple des descripteurs de tableaux et des catalogues de fichiers sur disquettes.

### ***2.2.***

La décomposition d'une instruction binaire simple en code opération, zones registre et adresse permet de montrer les caractéristiques et les inconvénients de la programmation en binaire, et la nécessité d'un langage d'assemblage.

L'étude d'un tel langage est hors programme.

L'étude systématique d'algorithmes de tri et d'interclassement n'est pas au programme.

Les élèves peuvent proposer eux-mêmes des projets ; ils doivent toujours être associés à leur définition.

### **3 - Aspects sociaux et culturels de l'informatique**

L'enseignement doit faire percevoir à l'élève la place de l'informatique dans la société, la diversité de ses domaines d'application, les problèmes juridiques, voire éthiques posés par son développement.

Les thèmes qui suivent reprennent et complètent ceux qui sont proposés en seconde. Ils sont donnés à titre indicatif. Ils peuvent être abordés à l'occasion d'activités diverses telles que conférences, films, émissions de télévision, analyses d'articles, exposés d'élèves, visites d'entreprises.

#### ***3.1. Applications de l'informatique***

Présentation, selon l'intérêt et la section des élèves, de quelques exemples d'applications de l'informatique telles que : gestion, calcul scientifique, dessin assisté par ordinateur, banque de données, simulation, bureautique, robotique, enseignement, vidéotex. Présentation et utilisation de progiciels

#### ***3.2. Informatique et société***

- Informatique, emploi, qualifications.
- Informatique et liberté ; problèmes de confidentialité et de sécurité liés à l'utilisation de fichiers.

### **4 - Projets**

Le projet fournit une occasion privilégiée de développer et de mettre en pratique certains thèmes du programme, de faire travailler les élèves en équipe. L'intérêt pédagogique d'un projet n'est lié ni à l'originalité ni à l'ampleur des problèmes à résoudre : l'objectif essentiel en est la prise en charge par les élèves. Les projets peuvent s'articuler autour de la conception et de la réalisation de logiciels. Dans ce cas on distingue bien les différentes phases : élaboration d'un cahier des charges, spécification fonctionnelle générale, découpage en plusieurs parties, analyse et programmation, élaboration de la documentation, intégration et tests globaux.

L'outil éventuel de programmation peut être un langage symbolique classique ou un générateur d'applications comme un tableur ou un gestionnaire de données.

### ***Commentaire***

#### **4.**

Les élèves peuvent proposer eux-mêmes des projets ; ils doivent toujours être associés à leur définition.

## **PROJET DE PROGRAMME DE TERMINALE**

Le programme définitif devrait être publié au B.O. avant la fin de l'année scolaire.

### **I - OBJECTIFS**

L'enseignement en classe de terminale doit d'abord renforcer les savoirs et savoir-faire acquis dans les classes précédentes, si nécessaire par des rappels substantiels. Mais il doit comporter aussi une ouverture vers les méthodes et techniques avancées de l'informatique et mettre l'accent sur la conception et la réalisation de projets.

Le programme qui suit présente un ensemble de notions dont certaines devront être abordées avec tous les élèves de Terminale (partie commune) et d'autres seulement de manière facultative. L'ordre adopté ne présume en rien celui que retient le professeur dans la conduite de son enseignement. La plupart du temps, ces notions se dégagent de la résolution de problèmes liés aux intérêts des élèves et aux programmes des autres disciplines.

L'atteinte des objectifs n'est pas subordonnée à une accumulation de connaissances abstraites ou techniques. Il est indispensable que les élèves puissent mobiliser effectivement les notions acquises lors du développement des projets.

### **II - PROGRAMME D'ENSEIGNEMENT ET COMMENTAIRES POUR LA CLASSE DE TERMINALE.**

#### **1. Méthodes et outils d'analyse et programmation.**

L'objectif de cette partie est l'évocation avec les élèves de quelques outils conceptuels puissants pour analyser les problèmes, concevoir des structures de données et programmer. L'utilisation de ces outils doit être cohérente avec la démarche adoptée dans les classes précédentes.

##### ***1.1. Partie commune obligatoire***

- Introduction à l'analyse et à la programmation récursives.

La notion de récursivité facilite l'analyse et la programmation. Mais elle peut être un élément perturbateur pour certains élèves. Elle doit donc être manipulée avec prudence, après révision de la notion de procédure.

- Idée d'une structure de données à choisir parmi : liste, table, file, pile, arbre, graphe.

La présentation d'une structure de données ne peut être comprise qu'à travers un exemple significatif rencontré à l'occasion de projets. Il faut donc éviter l'écueil d'une présentation axiomatique ou encyclopédique.

Il faut faire sentir aux élèves que le choix d'une organisation de données dépend essentiellement des opérations effectuées sur ces données.

- Notion de bibliothèque de logiciels.

Un projet se découpe généralement en modules analysés, programmés, testés séparément, et réutilisables. Il faut donner aux élèves le savoir-faire qui les aidera à se constituer eux-mêmes leur "boite à outils".

### ***1.2. Questions au choix (en choisir une au maximum)***

- Compléments sur la récursivité et l'itération : exécution d'un programme récursif, suppression de la récursivité dans des cas simples.

Pour mieux en faire comprendre le caractère opératoire, on peut ici démontrer le mécanisme d'exécution des procédures et fonctions récursives, avec la gestion de pile.

Dans le cas de la récursivité terminale, on peut montrer comment se ramener à une itération.

- Notion de programmation par objets.

- Notion de programmation fonctionnelle.

- Notion de programmation logique.

A côté de techniques traditionnelles de programmation où l'on pense à la manière dont s'exécute un programme, il est intéressant que les élèves aient vu une technique de programmation "déclarative", où l'on cherche plutôt à définir des objets, fonctions ou relations.

C'est dans ces voies nouvelles que s'engage aujourd'hui l'informatique. Mais il est toujours préférable de concrétiser ces notions par l'usage d'un langage adapté au matériel disponible.

## **2. Matériel et logiciel de base**

Il est indispensable de dépasser le micro-ordinateur isolé pour donner aux élèves quelques points de repère sur les grosses installations (celles de beaucoup d'entreprises), la télématique (de plus en plus

répandue dans la vie courante) et les réseaux locaux (existant dans certains établissements scolaires).

### ***2.1. Partie commune obligatoire***

- Composants d'une grosse installation.
- Réseaux locaux et télématique.

On peut ici se contenter de schémas généraux explicatifs afin de situer les grands composants, leurs fonctions et leurs liaisons : unités centrales, périphériques de différents types, câbles, liaisons téléphoniques, modems.

Il ne s'agit pas d'un cours de technologie, mais plutôt d'une leçon de vocabulaire d'une ou deux heures, éventuellement complétée d'une visite.

### ***2.2. Question facultative***

- Fonctions d'un système d'exploitation dans le cas de partage de ressources.

Ici encore, l'évocation des problèmes à résoudre est suffisante : quand plusieurs personnes veulent simultanément utiliser la même ressource limitée, il est souvent nécessaire d'ordonnancer son accès grâce au système d'exploitation.

Ceci peut être observé simplement sur un réseau local pour des demandes simultanées simples sorties sur une même imprimante ou accès à un même fichier.

## **3. Aspects sociaux et culturels de l'informatique.**

L'enseignement doit faire percevoir à l'élève la place de l'informatique dans la société, la diversité de ses domaines d'application, les problèmes juridiques, voire éthiques posés par son développement.

Les thèmes qui suivent, donnés à titre indicatif, reprennent et complètent ceux qui ont été préparés dans les classes précédentes. Ils peuvent être abordés à l'occasion d'activités diverses telles que conférences, films, émissions de télévision, analyse d'articles, exposés d'élèves, visites d'entreprises.

**3.1. Applications de l'informatique. En choisir au moins une dans la liste non limitative suivante :**

- gestion, calcul scientifique, dessin assisté par ordinateur, banque de données, simulation, bureautique, robotique, enseignement, vidéotex, Intelligence Artificielle.

**3.2. Informatique et société. Choisir au moins un thème parmi :**

- Histoire et épistémologie de l'informatique.
- Économie et informatique.
- L'informatique, phénomène culturel.
- L'informatique et la communication.
- Les métiers et les qualifications de l'informatique.
- Les aspects juridiques et éthiques de l'informatique (Loi "Informatique et liberté").

**4. Projets : Partie commune obligatoire**

Le projet fournit une occasion privilégiée de développer et de mettre en pratique certains thèmes du programme, de faire travailler les élèves en équipe. L'intérêt pédagogique d'un projet n'est lié ni à l'originalité, ni à l'ampleur des problèmes à résoudre : l'objectif essentiel en est la prise en charge par les élèves.

Les projets peuvent s'articuler autour de la conception et de la réalisation de logiciels. Dans ce cas on distingue bien les différentes phases : élaboration d'un cahier des charges, spécification fonctionnelle générale, découpage en plusieurs parties, analyse et programmation, élaboration de la documentation, intégration et tests globaux.