

PROGRAMME ET INSTRUCTIONS OFFICIELLES

POUR L'ENSEIGNEMENT DE

L'INFORMATIQUE

MPSI première et deuxième années

PREAMBULE

Le Ministère de l'Education Nationale a entrepris un important effort pour réviser les programmes dans les classes préparatoires marocaines.

Afin de contribuer aux efforts de rénovation du curriculum d'informatique, le présent guide se veut un outil qui permet de:

- Délimiter la nature et les caractéristiques de l'informatique en tant qu'objet d'enseignement ;
- Préciser les finalités et les objectifs généraux de l'enseignement de l'informatique dans les classes préparatoires ;
- Clarifier les approches pédagogiques servant de référence pour élaborer les activités d'apprentissage relatives à la discipline informatique ;
- Présenter le programme ainsi que la progression qui lui est attachée ;
- Suggérer quelques méthodes et moyens permettant à l'enseignant de planifier, d'animer et d'évaluer les apprentissages des apprenants.

Sommaire

- I. Contexte de la réforme de l'informatique en C.P.G.E
- II. Objectifs de la formation
- III. Moyens et matériel d'encadrement
- IV. Programme de la classe MPSI (première année)
- V. Programme de la classe MP (deuxième année)

I. Contexte de la réforme de l'informatique en C.P.G.E

Aujourd'hui, l'informatique compte parmi les indices de développement des pays, l'acquisition de l'information en temps réel est l'enjeu de tous les actes politiques, économiques, sociaux ...etc. Notre pays est conscient de ce facteur, et le Ministère de l'Education Nationale a entrepris depuis quelques années des démarches pour que l'informatique soit une matière d'enseignement de base, puisqu'elle est introduite dans notre enseignement secondaire voire fondamental. Entre les années 1985 (date de lancement des C.P.G.E au Maroc) et 1995, l'enseignement de l'informatique en C.P.G.E était omniprésent sous sa forme algorithmique mais de manière peu officielle, puisque les élèves ne subissaient pas d'épreuves d'évaluation en cette matière. A partir de l'année 1996, l'aspect algorithmique sera abandonné et remplacé par le calcul formel. L'évaluation de l'informatique au concours national n'entrera en vigueur sous forme d'épreuves orales qu'à partir de l'année 2002. Tenant compte de l'évolution de l'enseignement d'informatique au secondaire et de la nature diversifiée de l'enseignement aux grandes écoles d'ingénieurs Marocaines et étrangères, une nouvelle entreprise de l'enseignement d'informatique dans nos classes préparatoires s'impose, avec un double objectif :

1. Fournir des *méthodes* et des *outils* pour l'enseignement des mathématiques, de la physique, de la chimie et des sciences industrielles notamment en exploitant les possibilités offertes par un logiciel de calcul symbolique et formel comportant à la fois de puissantes facilités graphiques et éditoriales et des éléments de programmation performants. Le but est d'habituer les élèves à se servir d'outils logiciels, qui fournissent un support au *raisonnement* par la *confrontation* rapide et commode des hypothèses et des résultats permettant :

a. d'enrichir la compréhension des phénomènes mathématiques et des modèles physiques par la simulation de leurs comportements en fonction de divers paramètres ;

b. de mieux cerner la notion de domaine de validité d'une hypothèse ou d'une méthode par l'étude de cas limites ;

c. d'étudier certains problèmes par la mise en œuvre de modèles dont la résolution numérique manuelle serait lourde ou complexe ;

d. d'alléger la part de calcul systématique au profit de *l'intuition mathématique* et du *sens physique*.

2. Introduire l'informatique en tant que discipline par l'utilisation des différents outils de traitement informatique de l'information, à travers l'algorithmique et la programmation structurée. En effet, l'ambition de la nouvelle réforme est de présenter l'informatique comme une science opérant sur des représentations rigoureuses de concepts bien définis. La programmation consiste avant tout à déterminer la démarche d'obtenir, à l'aide d'un ordinateur, la solution d'un problème, c'est la recherche d'un algorithme. En ce sens, la programmation est de même nature philosophique qu'une démonstration mathématique.

II. Objectifs de la formation

A. Développement de compétences et d'aptitudes chez les élèves

L'enseignement de l'informatique aux C.P.G.E vise à développer chez nos élèves les compétences et les aptitudes suivantes :

1. L'exploitation des fonctionnalités et des performances qu'offre un logiciel de calcul Formel ou de calcul scientifique;
2. Une méthode d'analyse et une démarche de travail;
3. L'apprentissage des concepts de base de l'algorithmique, de la programmation et des structures de données;
4. La traduction d'un algorithme dans un langage de programmation.

Le programme de l'informatique aux CPGE, dans les deux années, est découpé en deux parties :

- Algorithmique et programmation
- Utilisation d'un logiciel de calcul symbolique et formel

B. Algorithmique et programmation

Les algorithmes sont aussi anciens que les mathématiques, certains très simples **agencent** la **succession** désirée de **tâches élémentaires** (calcul du périmètre d'un rectangle, calcul de la surface d'un disque, résolution des problèmes d'arithmétique élémentaire, ...), d'autres plus complexes mettent en jeu des **structures de données** comme les **arbres** ou les **graphes** qui représentent les différents états du système (optimisation de la représentation d'un lexique, liaisons ferroviaires, ...).

Dans l'enseignement, l'esprit algorithmique accompagne les résolutions et les démonstrations à tous les niveaux. **Décomposer** une tâche complexe en tâches élémentaires, reconnaître les tâches qui se **répètent**, **estimer la durée** du processus, sans oublier de vérifier que la succession d'opérations élémentaires produit bien le **résultat escompté** : c'est la démarche de celui qui écrit un programme, c'est aussi celle de tout être rationnel. Ces algorithmes sont traduits par la suite en **langage** compréhensible par la **machine**.

L'écriture d'un algorithme est l'occasion d'appliquer des règles de **logique** absolues dans un univers clairement défini et limité. Il permet aussi de distinguer entre **calculs théoriquement possibles** et **calculs effectivement réalisables ou approchés**, puisqu'un ordinateur ne peut manipuler que des **objets** complètement **explicites**. Ecrire un programme qui fonctionne récompense le programmeur de ses efforts de **réflexion**, d'**analyse** et de **synthèse**. Mais cela ne dispense pas de s'assurer que l'algorithme termine dans tous les cas envisagés et qu'il le fait en **temps raisonnable (cas limites et complexité)**.

Le programme de ce module se limite aux points suivants :

- Apprentissage des concepts de base de l'algorithmique, de la programmation et des structures de données en tirant les exemples des disciplines enseignées.
- Structures de données et objets.

1. Objectifs :

Le programme d'informatique doit permettre la présentation des principes de l'algorithmique et de la programmation.

Par ailleurs, un enseignement d'informatique doit être confronté à un principe de réalité : les élèves doivent écrire des programmes clairs, courts et précis.

La première partie du programme de première année est consacrée aux notions de base de l'algorithmique et les méthodes de programmation. Le but étant d'amener progressivement les élèves à avoir les compétences suivantes :

- Maîtrise de la méthode de programmation descendante.
- Programmation modulaire.
- Jeu de test pour validation des solutions et des documentations.
- Maîtrise de quelques méthodes de programmation : itérative, récursive,...

La deuxième partie du programme de première année et le programme de la deuxième année sont consacrés aux structures des données qui représentent une partie fondamentale dans l'étude des algorithmes afin d'atteindre les compétences suivantes :

- Représentation optimale des données afin de les traiter dans un algorithme.
- Réalisation d'algorithmes en fonction des structures de données utilisées (influence des structures sur l'algorithme).
- Gestion de la mémoire dynamique.
- Utilisation des traitements des structures des données dans la résolution des problèmes théoriques et réels (réservation, file d'attente,...)

2. Didactique :

- Nécessité d'organisation de données et de traitement.
- Présentation et étude des différents types de structures des données.
- Choix des données en fonction de la problématique.
- Les exemples sont issus des matières enseignées en classes prépas.
- Une activité algorithmique intégrée dans les autres disciplines enseignées, pas en termes de principe mais en termes d'applications (ex. analyse numérique).
- Les algorithmes, écrits dans le cours d'algorithmique, doivent être transcrits en langage de programmation.
- Il est recommandé de mettre l'accent sur l'aspect algorithmique beaucoup plus que sur le langage de programmation lui-même.

C. Utilisation d'un logiciel de calcul symbolique et formel

L'utilisation d'un logiciel de calcul formel qui, en exploitant la puissance de calcul de l'ordinateur, permettra :

- d'explorer profondément certains éléments (Algèbre linéaire, étude de fonctions, suites, résolution d'équations différentielles, simulations numériques du comportement des solutions d'équations, tracé de courbes, ...).
- de montrer, faire découvrir, susciter des réflexions, émettre des conjectures sur les notions acquises en classes prépas.
- de traiter des données en vraie grandeur.

1. Objectifs

Les premiers logiciels de calcul formel (computer algebra) datent de l'année 1953 (Nolan). La nécessité de ces logiciels dans le développement de la recherche scientifique ne

fait aucun doute. L'introduction de l'emploi d'un logiciel de calcul formel dans le programme d'informatique des C.P.G.E a deux objectifs :

- Mettre à la disposition des élèves des C.P.G.E un outil de calcul et de manipulation des formules algébriques leur permettant de vérifier un bon nombre de résultats obtenus lors des séances théoriques de Mathématiques ou de Sciences physiques : par une présentation simplifiée du logiciel, les élèves doivent acquérir, assez vite, une autonomie d'exploration et d'utilisation des différentes fonctionnalités de ce type de logiciels.
- Reconnaître les spécificités (précision infinie et manipulation d'expressions algébriques), la puissance (possession d'une grande bibliothèque extensible) ainsi que les difficultés (croissance en exponentielle de la durée d'exécution et de la place mémoire) d'un logiciel de calcul formel.

2. *Mise en œuvre*

Prenant place au *sein* des enseignements des mathématiques, de la physique, de la chimie et des sciences industrielles, l'enseignement du calcul formel s'effectue à travers l'emploi d'outils logiciels. En physique, en chimie et en sciences industrielles, cet emploi permet notamment la modélisation, la simulation et/ou le pilotage des dispositifs expérimentaux et des systèmes industriels.

La répartition des responsabilités entre les différents intervenants dans cette formation ainsi que la liste des différents logiciels et matériels au programme seront fixées dans une note de service de l'autorité gouvernementale chargée des classes préparatoires aux grandes écoles.

3. *Didactique*

- Un logiciel de calcul formel (se référer à la note ministérielle).
- Proposer tout le long des deux années un bon nombre de travaux pratiques portant sur :
 - Des exemples d'application du logiciel de calcul formel à la résolution de problèmes de Mathématiques
 - Des comparaisons du point de vue performance entre un logiciel de calcul formel et un logiciel de calcul numérique
 - Des exemples d'application du logiciel de calcul formel à la résolution de problèmes de Sciences physiques
 - Utilisation de logiciels de simulation en Sciences physiques
 - Acquisition automatique et traitement de données.
 - L'outil informatique pour le calcul formel doit être aussi intégré dans les autres disciplines enseignées, pas en termes de principe mais en termes d'applications.

III. Moyens et matériels d'encadrement

A. Volume horaire en classes prépas MPSI première année

Première période (16 semaines) : deux heures par semaine et par élève en groupe.

- 18h de cours et de travaux dirigés d'algorithmique.
- 08h de travaux pratiques de programmation.

- 06h de travaux pratiques : Introduction à l’outil de calcul formel.

Deuxième période (16 semaines) : deux heures par semaine et par élève en groupe.

- 16h de cours et de travaux dirigés d’algorithmique.
- 8h de travaux pratiques de programmation.
- 8h de travaux pratiques de calcul formel.

Total :

- Cours et de travaux dirigés d’algorithmique : 34h.
- Travaux pratiques de programmation : 16h.
- Travaux pratiques de calcul formel : 14h.

B. Volume horaire en classes prépas MP deuxième année

Première période (13 semaines) : deux heures par semaine et par élève en groupe.

- 12h de cours et de travaux dirigés d’algorithmique.
- 08h de travaux pratiques de programmation.
- 06h de travaux pratiques de calcul formel.

Deuxième période (13 semaines) : deux heures par semaine et par élève en groupe.

- 08h de cours et de travaux dirigés d’algorithmique.
- 06h de travaux pratiques de programmation.
- 12h de travaux pratiques de calcul formel.

Total :

- Cours et de travaux dirigés d’algorithmique : 20h.
- Travaux pratiques de programmation : 14h.
- Travaux pratiques de calcul formel : 18h.

C. Evaluation :

Afin d’évaluer le travail et la progression des élèves dans les différents aspects de la matière, les enseignants de l’informatique doivent envisager plusieurs types de contrôles de leurs élèves dont

- des devoirs surveillés,
- des colles,
- des travaux pratiques notés,
- un travail d’initiative personnel encadré utilisant l’outil informatique.

D. Moyens matériels et humains:

- Documentation : (livres, supports de cours,...).
- Matériels informatiques et logiciels : (salles équipées d’ordinateurs et de logiciels).
- Langages et logiciels : Une liste des langages et des logiciels doit faire l’objet d’une note actualisée.
- Enseignants : L’enseignement d’informatique doit être dispensé par un

d. Exemples.

4. Les exemples d'applications

a. Les exemples sont issus du programme de mathématiques (arithmétique, algèbre linéaire, analyse,...) :

Algorithme d'exponentiation rapide, Pivot de Gauss, Algorithme d'Euclide, Calcul matriciel, résolution d'équations numériques, calcul polynômial, ...

b. Des algorithmes de Tri et de recherche

Recherche séquentielle, Tri simple d'un tableau, Tri par sélection, Tri par insertion, Tri à bulles, ...

5. La récursivité simple

a. Principe de la récursivité simple.

b. Exemples d'utilisation de la récursivité simple.

c. Occupation de mémoire (Pile).

d. Terminaison d'une fonction récursive.

e. Récursivité et Itération (champs d'utilisation).

Exemples d'algorithmes récursifs et itératifs (factorielle, puissances entières, suites récurrentes, pgcd,...).

B. Calcul formel

Un Logiciel de calcul formel doit permettre un ensemble de traitements comprenant :

1. Les manipulations élémentaires

a. Type d'une variable.

b. Affectation.

c. Evaluation d'une variable.

d. Réinitialisation d'une variable.

e. Formation d'expressions et simplifications possibles.

2. Les calculs usuels de type arithmétique ou flottant

a. Calculs exacts dans \mathbf{N} , \mathbf{Z} , \mathbf{Q} et sur les expressions.

b. Calculs approchés dans \mathbf{R} et \mathbf{C} .

c. Calcul exact et approché des solutions d'une équation.

d. Utilisation des opérateurs, fonctions et constantes mathématiques usuelles.

e. Calcul arithmétique : pgcd, ppcm, division euclidienne, nombres premiers, ...

3. Le calcul polynomial

a. Manipulation des sommes et des produits.

b. Calcul arithmétique sur les polynômes.

c. Développement et factorisation des polynômes.

d. Décomposition en éléments simples des fractions rationnelles.

4. Les différentes structures

a. Liste, Séquence, Ensemble, Table et Tableau.

b. Manipulation et conversion de ces structures.

5. **Les opérations sur des fonctions**
 - a. Définitions de fonctions.
 - b. Dérivation et Intégration formelles de fonctions.
6. **Les limites et développements asymptotiques**
 - a. Limite d'une expression : d'une suite, d'une fonction. Somme d'une série...
 - b. Suites récurrentes.
 - c. Développements limités et asymptotiques.
7. **Les commandes graphiques 2D**
 - a. Représentation en 2D de courbes en coordonnées cartésiennes, polaires et paramétriques.
 - b. Éléments de géométrie plane : Tracés et manipulation d'objets géométriques tels que les lignes, les triangles, les cercles...
8. **L'algèbre linéaire**
 - a. Représentation d'une matrice et d'un vecteur.
 - b. Opérations courantes sur les matrices : Somme, produit, inverse, rang, déterminant...
 - c. Résolution formelle ou numérique d'équations ou de systèmes d'équations linéaires.
 - d. Opérations sur les vecteurs : Produit scalaire, produit vectoriel...
9. **Les équations différentielles**
 - a. Résolution d'équations différentielles linéaires du premier et du deuxième ordre.
 - b. Résolution numérique d'équations différentielles.

VI. Programme de la classe MP (deuxième année)

A. Algorithmique et programmation : Structures de données

1. **Définition et rôle des structures de données en programmation**
2. **Les enregistrements**
 - a. Définition et manipulation d'enregistrement. (Manipulation globale et par champ).
 - b. Exemples d'application.
3. **L'allocation dynamique de la mémoire**
 - a. Notion d'adresse mémoire.
 - b. Allocation et libération de mémoire.
 - c. Exemples.
4. **Les listes chaînées**
 - a. Déclaration d'une liste chaînée.
 - b. Création d'une liste chaînée.
 - c. Fonctions de manipulation de listes : insertion, suppression, recherche, tri, MAJ, ... (utilisation de la récursivité et de l'itération).

5. **Les Piles et les Files**
 - a. Définition d'une Pile.
 - b. Fonctions de manipulation d'une Pile.
 - c. Exemples : Evaluation d'expressions arithmétiques, ...
 - d. Définition d'une File.
 - e. Fonctions de manipulation d'une File.
 - f. Exemples d'application.

L'étude des structures de données (enregistrement, liste chaînée, etc.) doit se faire à l'aide d'exemples illustratifs accompagnés de schémas.

B. Calcul formel

1. **Expressions algébriques (Complément)**
 - a. Notion d'opérande et classement des objets par type.
 - b. Représentation arborescente d'une expression.
 - c. Opérations sur les opérandes.
2. **Programmation**
 - a. Définition d'une procédure.
 - b. Variables globales et variables locales.
 - c. Entrées et sorties dans une procédure.
 - d. Structures conditionnelles et itératives.
 - e. Débogage d'une procédure.
 - f. Notion de package.
3. **Compléments d'algèbre linéaire et bilinéaire**
 - a. Eléments propres d'une matrice.
 - b. Polynôme caractéristique et polynôme minimal.
 - c. Diagonalisation et triangulation.
4. **Opérateurs vectoriels**
 - a. Gradient, Divergence, Rotationnel et Laplacien...
 - b. Exemples de résolution d'équations aux dérivées partielles.
5. **Commandes graphiques 3D**
 - a. Représentation en 3D de courbes et de surfaces en coordonnées cartésiennes, cylindriques et sphériques.
 - b. Animation de ces tracés.