

B u l l e t i n

Enseignement de spécialité d'informatique et sciences du numérique de la série scientifique - classe terminale

NOR : MENE1119484A
arrêté du 12-7-2011 - J.O. du 20-9-2011
MEN - DGESCO A3-1

Vu code de l'éducation ; arrêté du 27-1-2010 modifié ; avis du CSE du 9-6-2011

Article 1 - Le programme de l'enseignement de spécialité d'informatique et sciences du numérique en classe terminale de la série scientifique est fixé conformément à l'annexe du présent arrêté.

Article 2 - Les dispositions du présent arrêté entrent en application à la rentrée de l'année scolaire 2012-2013.

Article 3 - Le directeur général de l'enseignement scolaire est chargé de l'exécution du présent arrêté qui sera publié au Journal officiel de la République française.

Fait le 12 juillet 2011

Pour le ministre de l'éducation nationale, de la jeunesse et de la vie associative
et par délégation,
Le directeur général de l'enseignement scolaire,
Jean-Michel Blanquer

Annexe

Programme de l'enseignement de spécialité d'informatique et sciences du numérique Classe terminale de la série S

1 - Préambule

Les sciences informatiques, et plus généralement les sciences du numérique, ont aujourd'hui envahi nos vies professionnelles et personnelles. Elles ont entraîné des mutations profondes dans nos sociétés (culture, sciences, économie, politique, etc.). Pourtant, seule une faible partie de la population maîtrise les mécanismes fondamentaux qui régissent ces mutations et est en mesure d'apprécier les enjeux sociétaux qui en découlent. L'enseignement de l'informatique au lycée peut contribuer à réduire cette fracture.

L'objectif de l'enseignement de spécialité ISN en classe terminale de la série S n'est pas de former des experts en informatique, mais plutôt de fournir aux élèves quelques notions fondamentales et de les sensibiliser aux questions de société induites.

Il s'agit d'un enseignement d'ouverture et de découverte des problématiques actuelles, adapté à la société d'aujourd'hui, qui valorise la créativité et contribue à l'orientation.

2 - Mise en activité de l'élève

Afin de refléter le caractère scientifique et technique propre à la discipline et de développer l'appétence des élèves en faveur de cet enseignement nouveau pour eux, il convient de les mettre en situation d'activité aussi souvent que possible. Une pédagogie de projet est à privilégier pour favoriser l'émergence d'une dynamique de groupe. Dans ce cadre, le professeur joue un rôle central : il impulse et coordonne les projets, anime les débats et met en place l'évaluation et ses modalités. L'informatique étant connexe à de nombreux domaines, il est utile d'envisager un travail pluridisciplinaire : la complémentarité des approches, associée à la richesse d'un travail collaboratif, joue un rôle stimulant pour les élèves et les équipes pédagogiques. Le professeur peut s'appuyer sur la mise en place d'exposés suivis de débats au sein de la classe pour introduire des questions sociétales liées à la généralisation du numérique. Enfin, lors de la préparation des exposés, comme lors du développement des projets, le professeur guide les élèves dans leurs recherches documentaires s'appuyant sur des livres ou des ressources présentes sur le Web.

La progression peut suivre un rythme annuel construit autour de périodes spécifiques favorisant une alternance entre différents types d'activités (acquisition de nouveaux savoirs, exposés, projets) permettant d'entretenir l'intérêt des élèves et de développer leur autonomie.

Les activités pratiques et la réalisation de projets sont organisées dans une salle qui permet d'enseigner les bases et notions théoriques fondamentales, avec un recours possible aux outils numériques de présentation adaptés. Un environnement numérique suffisamment ouvert est choisi pour favoriser cette dynamique de projet.

3 - Les projets

Les projets réalisés par les élèves, sous la conduite du professeur, constituent un apprentissage fondamental tant pour la compréhension de l'informatique et des sciences du numérique que pour l'acquisition de compétences variées. Ils peuvent porter sur des problématiques issues d'autres disciplines et ont essentiellement pour but d'imaginer des solutions répondant à l'expression d'un besoin.

Les compétences mises en jeu au cours du développement d'un projet peuvent être regroupées ainsi :

- proposer une approche fonctionnelle qui réponde aux besoins ;
- conduire des recherches documentaires ;
- concevoir des programmes en autonomie ;
- gérer les étapes de l'avancement du projet en dialogue et en interaction avec le professeur.

Les activités des élèves sont organisées autour d'une équipe de projet formée de deux ou trois élèves et dont le fonctionnement est guidé par une démarche incluant des points d'étape pour faire un bilan avec le professeur, valider des éléments, contrôler ou modifier l'avancement du projet, voire le redéfinir partiellement.

L'enseignant veille à ce que les projets choisis par les élèves restent d'une ambition raisonnable afin de ne pas empiéter sur le temps consacré aux autres disciplines. Un projet mené durant la seconde partie d'année permet de mettre en œuvre les savoirs et capacités acquises et donne lieu à un rapport écrit d'une dizaine de pages au maximum.

Au long de l'année, l'évaluation du travail de l'élève s'appuie sur les capacités mentionnées ci-dessous, sans nécessairement les mettre toutes en jeu.

4 - Éléments de programme

Le programme est construit autour de quatre parties : **représentation de l'information, algorithmique, langages et programmation, architectures matérielles**. Les séquences pédagogiques ont vocation à être construites en combinant des savoirs et capacités extraits des quatre parties du programme.

Organisation : les éléments du programme sont présentés à l'aide d'un tableau en trois colonnes : **Savoirs, Capacités, Observations**.

Une partie des savoirs et capacités, repérés par le signe distinctif ♦, sont optionnels et seront traités en fonction des équipements disponibles ainsi que des orientations pédagogiques choisies par les enseignants.

4.1 Représentation de l'information

Dans un contexte informatique, l'information est représentée par des suites de nombres. La numérisation est l'opération qui associe à un objet réel du monde physique une description à l'aide d'un ensemble d'informations exploitables par un ordinateur ou, plus généralement, une machine numérique. À cause de l'échantillonnage sous-jacent, la numérisation induit des effets importants sur la qualité de l'information numérique. Elle entraîne des conditions spécifiques de création, de stockage, de traitement et de circulation de l'information.

Les capacités de traitement et de stockage des ordinateurs croissent de façon continue depuis leur apparition. Il est donc crucial d'organiser ces flux d'informations en local sur une machine ou de façon distribuée sur un réseau.

L'intégration croissante du numérique dans les activités humaines et la numérisation de l'information suscitent des transformations culturelles, socio-économiques, juridiques et politiques profondes qui font apparaître de nouvelles opportunités, de nouveaux risques et de nouvelles contraintes qu'il convient d'étudier.

Savoirs

Représentation binaire

Un ordinateur est une machine qui manipule des valeurs numériques représentées sous forme binaire.

Opérations booléennes

Présentation des opérations booléennes de base (et, ou, non, ou-exclusif).

Numérisation

L'ordinateur manipule uniquement des valeurs numériques. Une étape de numérisation des objets du monde physique est donc indispensable.

Capacités

Manipuler à l'aide d'opérations

élémentaires les trois unités de base : bit, octet, mot.

Exprimer des opérations logiques

simples par combinaison d'opérateurs de base.

Coder un nombre, un caractère au travers d'un code standard, un texte sous forme d'une liste de valeurs numériques.

Numériser une image ou un son sous forme d'un tableau de valeurs numériques.

♦ **Modifier** format, taille, contraste ou luminance d'images numériques.

♦ **Filtrer** et détecter des informations spécifiques.

♦ **Créer** une image à l'aide d'un logiciel de modélisation.

Observations

On met en évidence, sous forme de questionnement, la présence du numérique dans la vie personnelle et professionnelle, au travers d'exemples.

On découvre les opérations logiques de base à l'aide d'exercices simples et on met en évidence ces opérations dans les mécanismes de recherche.

En parallèle avec les séances d'algorithmique, on peut expliquer le principe d'addition de deux octets.

Il est ici utile de faire référence à des notions technologiques introduites à propos des architectures matérielles. Les images et les sons sont choisis comme contexte applicatif et sont manipulés via des logiciels de traitement ou de synthèse.

Le traitement numérique de la lumière et du son est en lien avec les principes physiques sous-jacents, qu'il est utile d'évoquer au moment voulu.

Formats

Les données numériques sont agencées de manière à en faciliter le stockage et le traitement. L'organisation des données numériques respecte des formats qui sont soit des standards de fait, soit des normes.

Compression

- ◆ Notion de compression de données.
- ◆ Compression avec et sans perte d'information.

Structuration et organisation de l'information

On manipule de grandes quantités d'informations. Il est nécessaire de les organiser.

Persistence de l'information

Les données, notamment personnelles, sont susceptibles d'être mémorisées pour de longues périodes sans maîtrise par les personnes concernées.

Non-rivalité de l'information

Existence de lois régissant la détention et la circulation de données numériques.

Identifier quelques formats de documents, d'images, de données sonores.

Choisir un format approprié par rapport à un usage ou un besoin, à une qualité, à des limites.

◆ **Utiliser** un logiciel de compression.

Classer des informations, notamment sous forme d'une arborescence.

Prendre conscience de la persistance de l'information sur les espaces numériques interconnectés.

Comprendre les principes généraux permettant de se comporter de façon responsable par rapport au droit des personnes dans les espaces numériques.

Prendre conscience de la non-rivalité des biens immatériels.

Distinguer différents types de licences (libres, propriétaires).

Le choix d'un format approprié pose le problème de l'interopérabilité, qui est le fait d'assurer un usage sans restriction des mêmes données sur un système différent.

On met en évidence l'effet de la compression d'une image ou d'un son en comparant deux systèmes de compression (avec ou sans perte).

On peut ici étudier le système d'organisation de fichiers en dossiers. Un ensemble de documents unis par des liens hypertextes fournit un exemple de classement de type graphe.

La persistance de l'information se manifeste tout particulièrement au sein des disques durs mais aussi des mémoires caches. Elle interagit avec le droit à la vie privée et fait naître une revendication du « droit à l'oubli ».

La non-rivalité d'un bien se définit par le fait que son usage par une personne n'en limite pas l'usage par d'autres (ainsi, le poste de radio est rival mais l'émission ne l'est pas).

À l'occasion d'exposés suivis de débats, on sensibilise les élèves à l'évolution des valeurs et du droit (en France et ailleurs) induite par l'émergence de biens immatériels.

4.2 Algorithmique

Un algorithme se définit comme une méthode opérationnelle permettant de résoudre, en un nombre fini d'étapes clairement spécifiées, toutes les instances d'un problème donné. Cette méthode peut être exécutée par une machine ou par une personne.

Les élèves ont été confrontés aux algorithmes très tôt dans leur parcours scolaire (avec les quatre opérations arithmétiques) et régulièrement de nouvelles situations de nature algorithmique leur ont été proposées ; ainsi, la construction de figures en géométrie euclidienne, la transcription des « formules » moléculaires en chimie, le code génétique ou encore l'analyse fonctionnelle en technologie sont autant de situations évoquant des algorithmes. Les programmes de mathématiques des classes de seconde et première contiennent une initiation à l'algorithmique sur laquelle il convient de s'appuyer.

À travers l'étude de quelques algorithmes, on développe la faculté de lire et comprendre un algorithme conçu par d'autres, puis d'en concevoir de nouveaux. Ces algorithmes sont exprimés dans un langage de programmation et exécutés sur une machine ou bien définis de manière informelle.

Savoirs

Algorithmes simples

- rechercher un élément dans un tableau trié par une méthode dichotomique ;
- trier un tableau par sélection ;
- ajouter deux entiers exprimés en binaire.

Algorithmes plus avancés

- ◆ tri par fusion ;
- ◆ recherche d'un chemin dans un graphe par un parcours en profondeur (DFS) ;
- ◆ recherche d'un plus court chemin par

Capacités

Comprendre un algorithme et **expliquer** ce qu'il fait.

Modifier un algorithme existant pour obtenir un résultat différent.

Concevoir un algorithme.

Programmer un algorithme.

◆ **S'interroger** sur l'efficacité d'un algorithme.

Comprendre et expliquer (oralement ou par écrit) ce que fait un algorithme.

◆ **S'interroger** sur l'efficacité d'un algorithme.

Observations

On présente simultanément les notions d'algorithme et de programme, puis on les distingue.

L'objectif est une compréhension de ces algorithmes et la capacité à les mettre en œuvre.

Les situations produisant une erreur (division par zéro, dépassement de capacité) sont mises en évidence.

L'objectif se limite à une compréhension des principes fondamentaux sans exiger leur programmation.

un parcours en largeur (BFS).

4.3 Langages et programmation

La programmation est l'expression d'un algorithme dans un langage exécutable par une machine et joue un rôle central dans le développement des systèmes et produits informatiques.

L'apprentissage de la programmation vise d'une part à savoir **programmer un algorithme** décrit en langue naturelle et d'autre part à **comprendre un programme** et exprimer en langue naturelle l'algorithme sous-jacent.

On commence par rappeler les éléments de base de tout langage de programmation (affectation, séquence, test et boucle) tels qu'ils ont été présentés en mathématiques en classe de seconde et consolidés en classe de première. On introduit alors la notion de fonction qui permet d'éviter des redondances, de structurer les programmes et d'organiser leur conception. Enfin, on met en évidence la qualité des programmes en les testant sur différents jeux de données.

On insiste sur la clarté et la documentation qui facilitent la reprise du code par d'autres programmeurs. On montre enfin l'universalité de la notion de langage au-delà de la programmation.

L'enseignant choisit un langage de programmation selon les critères suivants : simplicité d'utilisation, liberté d'installation, présence d'outils associés, existence d'une communauté d'utilisateurs et de bibliothèques facilitant le développement.

Savoirs

Types de données

- nombre entier ;
- virgule flottante ;
- booléen ;
- caractère ;
- tableau ;
- chaîne de caractères.

Fonctions

- notion de fonction ;
- portée des variables et passage d'arguments ;
- définition récursive de fonctions.

Correction d'un programme

- test ;
- instrumentation ;
- situations d'erreur ou *bugs*.

Langages de description

Présentation du langage HTML et du principe de séparation du contenu et de la mise en forme.

Capacités

Choisir un type de donnée en fonction d'un problème à résoudre.

Concevoir l'entête (ou l'interface) d'une fonction, puis la fonction elle-même.

Mettre un programme au point en le testant, en l'instrumentant.

♦ **Utiliser** un outil de mise au point.

Créer et analyser une page web en langage HTML.

Observations

On adapte la présentation de ces notions en fonction du langage de programmation retenu.

On adapte la présentation de ces notions en fonction du langage de programmation retenu.

On évoque les risques issus des programmes incorrects et des *bugs* qui en résultent, aux conséquences parfois graves.

On met en valeur le double usage du langage, lisible par un humain et interprétable par une machine. On utilise HTML pour écrire une page « à la main », puis on insiste sur le fait que ce langage sert aussi de cible à des générateurs de pages. On évalue la qualité des pages du point de vue de la correction syntaxique et de l'efficacité du message.

4.4 Architectures matérielles

Exprimer un algorithme dans un langage de programmation a pour but de le rendre exécutable par une machine numérique.

La découverte de l'architecture de ces machines constitue une étape essentielle d'une initiation à l'informatique. De plus, mieux comprendre cette organisation est nécessaire pour programmer de manière efficace, en tenant compte des capacités et limitations des machines numériques.

La progression pédagogique suit la chronologie du développement des systèmes informatiques : d'abord centralisés autour des machines à accès direct, ensuite connectés par l'intermédiaire d'une liaison série point à point et enfin répartis grâce aux réseaux où le transport des informations repose sur des méthodes de routage. Le développement de ces réseaux et leur utilisation massive ont induit des questions sociétales majeures qu'il est préférable d'aborder sous forme d'activités pluridisciplinaires. Finalement, l'étude d'un minirobot permet de découvrir les mécanismes de pilotage et de communication dans l'exécution de tâches complexes, interférant directement avec le monde physique.

Architecture des ordinateurs

Savoirs

Éléments d'architecture

Composants de base (unité centrale, mémoires, périphériques).

Jeu d'instructions

Instructions simples (chargement, stockage, opérations arithmétiques et logiques, saut conditionnel).

Capacités

Expliquer le rôle des constituants d'un ordinateur.

♦ **Savoir dérouler** l'exécution d'une séquence d'instructions simples de type langage machine.

Observations

On se limite à une présentation générale de ces concepts autour d'une machine à accès direct (Random Access Machine).

On propose des activités sous forme d'exercices sur papier sans utiliser d'ordinateur.

Réseaux

Savoirs

Transmission point à point

Principes de base d'une transmission d'informations numériques entre un émetteur et un récepteur.

Adressage sur un réseau

Mécanismes d'adressage pour identifier des machines distantes.

Routage

Mécanismes induits par la communication sur un réseau dont la structure est de type graphe. Notions de paquets, de chemins, de routage.

Supranationalité des réseaux

Capacités

♦ **Établir** une communication sérielle entre deux machines.

Décrire une situation d'adressage sur un type de réseau particulier.

♦ **Analyser** le trafic (trames) sur un réseau et mettre ainsi en évidence la notion de protocole.

♦ **Analyser** les entêtes de messages électroniques, pour décrire le chemin suivi par l'information.

Prendre conscience du caractère supranational des réseaux et des conséquences sociales, économiques et politiques qui en découlent.

Observations

On s'interroge sur la qualité d'une liaison série point à point.

On se limite à l'analyse d'un trafic de type « chat » (échange de caractères codés).

On introduit la notion de protocole (règles, formats et conventions, sur lesquels il est nécessaire de s'accorder pour communiquer).

Au-delà de deux machines, le modèle de la liaison point à point ne convient plus.

On introduit ces notions en comparant différents types d'adressages existants (téléphone, courrier postal).

On fait appel à un outil d'analyse pour visualiser la transmission des trames nécessaires au dialogue entre machines numériques.

On se limite à la mise en œuvre d'une séance de travaux pratiques, avec analyse d'entêtes de courriels prédéfinis reçus (aspect distribué et non fiable des réseaux de grande taille, difficulté du passage à l'échelle).

On explique la différence entre les réseaux de type arborescent et de type graphe.

On met en évidence le fait que certains pays autorisent la mise en ligne d'informations, services ou contenus numériques dont la consultation n'est pas permise dans d'autres pays.

Initiation à la robotique

Savoirs

Découverte d'un système robotique et de sa programmation

Capacités

♦ **Identifier** les différents composants d'un minirobot et comprendre leurs rôles respectifs.

♦ **Décrire** un système à événements simple à l'aide d'une machine à états finis.

♦ **Programmer** (dans un langage de haut niveau) un minirobot pour lui faire exécuter une tâche complexe.

Observations

On propose des activités adaptées aux équipements et logiciels disponibles dans l'établissement.