

2008

RÉVISÉ

Le curriculum de l'Ontario
de la 10^e à la 12^e année

Études informatiques



appuyer chaque élève

 Ontario

TABLE DES MATIÈRES

INTRODUCTION	3
Les écoles secondaires au XXI ^e siècle	3
L'école de langue française	3
La place du programme-cadre d'études informatiques dans le curriculum	5
Les objectifs du programme-cadre d'études informatiques	6
Le rôle de l'élève	7
Le rôle des parents	7
Le rôle de l'enseignante ou l'enseignant	8
Le rôle de la directrice ou du directeur d'école	8
ORGANISATION DU PROGRAMME-CADRE D'ÉTUDES INFORMATIQUES	11
Les cours offerts	11
Les domaines d'étude	13
Les attentes et les contenus d'apprentissage	15
ÉVALUATION DU RENDEMENT DE L'ÉLÈVE	17
Le processus d'évaluation du rendement de l'élève	17
La grille d'évaluation du rendement	18
La communication du rendement	22
CONSIDÉRATIONS CONCERNANT LA PLANIFICATION DU PROGRAMME	23
Les stratégies d'enseignement et d'apprentissage	23
L'importance de l'actualité dans les cours d'études informatiques	24
La place des technologies dans les cours d'études informatiques	24
La planification des cours d'études informatiques destinés aux élèves en difficulté	24

*An equivalent publication is available in English under the title
The Ontario Curriculum, Grades 10-12, Computer Studies.*

Cette publication est affichée sur le site Web du ministère de l'Éducation
au www.edu.gov.on.ca.

L'élève bénéficiant des programmes d'actualisation linguistique en français ou de perfectionnement du français	27
L'éducation antidiscriminatoire et les études informatiques	28
L'éducation environnementale et les études informatiques	30
Les habiletés de la pensée et de la recherche	31
La littératie et la numératie	31
La Majeure Haute Spécialisation	32
La planification de carrière	33
Le Passeport-compétences de l'Ontario et les compétences essentielles	33
L'éducation coopérative et les autres expériences en milieu de travail	34
La santé et la sécurité	35

COURS **37**

Introduction à l'informatique, 10 ^e année, cours ouvert (ICS2O)	39
Introduction au génie informatique, 11 ^e année, cours préuniversitaire (ICS3U)	47
Introduction à la programmation informatique, 11 ^e année, cours précollégial (ICS3C)	53
Génie informatique, 12 ^e année, cours préuniversitaire (ICS4U)	59
Programmation informatique, 12 ^e année, cours précollégial (ICS4C)	65

GLOSSAIRE **71**

INTRODUCTION

Le présent document *Le curriculum de l'Ontario – Études informatiques, de la 10^e à la 12^e année, édition révisée, 2008* est destiné aux écoles de langue française; il remplace le cours d'études informatiques du document intitulé *Le curriculum de l'Ontario – Éducation technologique, 9^e et 10^e année, 1999* et les deux cours d'études informatiques en partie B du document *Le curriculum de l'Ontario – Éducation technologique, 11^e et 12^e année, 2000*. À compter de septembre 2009, le programme d'études informatiques, de la 10^e à la 12^e année, sera fondé sur les attentes et les contenus d'apprentissage énoncés dans les pages suivantes.

LES ÉCOLES SECONDAIRES AU XXI^e SIÈCLE

Les écoles secondaires de l'Ontario offrent à tous les élèves un programme d'études varié et planifié de grande qualité. Ce programme vise la réussite de tous les élèves dans la destination de leur choix. La mise à jour du curriculum de l'Ontario, de pair avec un élargissement des options d'apprentissage offertes à l'extérieur de la salle de classe, intègre l'apprentissage des compétences essentielles pour réussir au XXI^e siècle et respecte les champs d'intérêt, les forces ainsi que les besoins des élèves.

L'ÉCOLE DE LANGUE FRANÇAISE

À l'école secondaire de langue française, un apprentissage de qualité se déroule dans un environnement propice à la construction de l'identité francophone. En effet, s'éveiller et s'ouvrir à la francophonie, prendre conscience de ses enjeux, identifier ses caractéristiques, s'y engager avec fierté et contribuer à la vitalité de ses institutions représentent sans aucun doute la plus-value de l'apprentissage proposé.

À l'appui du mandat de l'école de langue française, la *Politique d'aménagement linguistique de l'Ontario pour l'éducation en langue française, 2004* définit la nature et la portée des interventions en aménagement linguistique ainsi que les résultats escomptés. Ces résultats sont de trois ordres :

- Pour les élèves : capacité accrue à acquérir les compétences en communication orale afin de maximiser l'apprentissage et la construction identitaire.
- Pour le personnel scolaire : capacité accrue à œuvrer en milieu minoritaire afin d'appuyer les apprentissages scolaires et le développement identitaire de chaque élève.
- Pour les conseils scolaires : capacité accrue à maintenir et à augmenter l'effectif scolaire afin de contribuer à la vitalité des écoles de langue française et de la communauté francophone.

Pour parvenir à ces résultats, le personnel enseignant tient compte des attentes génériques suivantes :

- L'élève utilise sa connaissance de la langue française et sa capacité à communiquer oralement en français pour interpréter de l'information, exprimer ses idées et interagir avec les autres.
- L'élève manifeste son engagement pour la culture francophone en s'informant sur les référents culturels de la francophonie, en les faisant connaître, en en discutant et en les utilisant dans diverses situations.

Dans sa planification des activités d'enseignement et d'apprentissage, le personnel enseignant de l'école conçoit des interventions en aménagement linguistique qui réunissent les conditions favorables à la création d'un espace francophone respectueux du dynamisme et du pluralisme de la communauté et qui contrent les effets négatifs de l'assimilation sur la réussite des élèves. L'école de langue française, milieu de bilinguisme additif, permet aux élèves d'acquérir d'abord de solides compétences langagières en français à l'oral et à l'écrit et d'assurer un apprentissage de l'anglais langue seconde. De plus, elle invite les élèves à prendre conscience des avantages de maîtriser les deux langues officielles du Canada. Les élèves utilisent leur capacité à communiquer oralement en français pour apprendre à se connaître, à construire leur identité, à apprendre avec les autres et à faire état de leurs apprentissages.

La politique d'aménagement linguistique de l'Ontario (PAL) comporte, entre autres, deux axes d'intervention qui ciblent la réussite scolaire et le développement de la personne.

L'axe de l'apprentissage. Cet axe d'intervention porte sur l'appropriation des savoirs et le choix de carrière. Le curriculum de l'Ontario définit les compétences transdisciplinaires que tous les élèves doivent acquérir pour évoluer comme francophones dans la vie et dans la société, c'est-à-dire savoir communiquer oralement, savoir lire, savoir écrire, savoir rechercher l'information, savoir se servir des technologies de l'interaction et savoir exercer une pensée critique. Garante de la réussite scolaire, l'acquisition de ces compétences de base se fait graduellement et en parallèle avec la découverte des champs d'intérêt et des talents individuels, ce qui amènera chaque élève à définir son rôle dans la société et à choisir son domaine d'activité professionnelle.

L'axe de la construction identitaire. Cet axe d'intervention porte sur l'appropriation de la culture et le développement de l'identité. En approfondissant sa connaissance de la culture de langue française, l'élève acquiert un ensemble de repères culturels qui lui permettent d'interpréter le monde et de découvrir les traits distinctifs et les manifestations de la francophonie sur le plan matériel, culturel et intellectuel. Chez l'élève, ce cheminement culturel vient encadrer sa démarche de construction identitaire qui s'opère en trois étapes : *l'ouverture et le constat* où l'élève s'éveille au milieu environnant et à la réalité culturelle francophone, *l'expérience* où l'élève prend contact de façon approfondie et plus active avec les contextes socioculturels, et *l'affirmation* où l'élève fait des choix déterminants pour s'engager et affirmer son identité.

L'école de langue française doit aussi s'assurer de créer des situations d'apprentissage qui permettent aux élèves d'affirmer leur identité comme francophones. Les attentes génériques de même que les attentes et les contenus d'apprentissage propres à chaque matière ou discipline visent le cheminement de l'élève sur les plans personnel, interpersonnel et professionnel. En incitant les élèves à discuter de leurs apprentissages par

rapport à leurs émotions, leurs valeurs et leurs connaissances antérieures, on développe simultanément chez eux l'expression de la pensée et le courage d'exposer un point de vue et de le confronter à d'autres avec confiance et respect. Ainsi, toutes les attentes et tous les contenus d'apprentissage du curriculum de l'Ontario constituent un tremplin à partir duquel l'élève peut, en perfectionnant ses compétences linguistiques, construire son identité et s'engager envers la culture francophone.

En instaurant dans la salle de classe une ambiance collégiale et respectueuse des divers niveaux d'habiletés linguistiques et des différences culturelles, on contribue à rehausser chez les élèves l'estime de soi, à développer des relations individuelles et de groupe avec les personnes de culture perçue différente de la leur et à construire une identité forte et engagée.

Finalement, les expériences vécues dans le milieu communautaire et les expériences de travail prévues dans les cours d'études informatiques offrent d'excellentes occasions pour que les élèves s'engagent dans des activités sociales, communautaires ou culturelles et consolident leurs liens avec la communauté francophone de l'Ontario.

LA PLACE DU PROGRAMME-CADRE D'ÉTUDES INFORMATIQUES DANS LE CURRICULUM

Au cours des dernières décennies, notre société a connu des transformations aussi rapides que profondes. Les progrès technologiques ont considérablement amélioré les moyens de communication et les outils de production. Les distances qui se réduisent, la quantité d'information partagée en ligne qui augmente et les logiciels qui deviennent de plus en plus performants et accessibles ont contribué à la vulgarisation et à l'appropriation de l'informatique par des utilisatrices et utilisateurs de plus en plus expérimentés, à l'émergence de nouvelles pratiques de travail telles que les équipes virtuelles ou le travail à distance, et au phénomène de mondialisation particulièrement bien illustré par l'externalisation des services informatiques.

Pour réussir dans le monde professionnel, l'élève doit non seulement utiliser efficacement les outils technologiques et traiter un volume important d'information, mais aussi être en mesure de continuellement adapter et de mettre à jour ses connaissances pour faire face aux changements de la société. Les cours d'études informatiques offrent à l'élève une opportunité unique de comprendre le fonctionnement d'un ordinateur de bureau, d'un téléphone mobile ou d'une calculatrice graphique, ce qui le conduira naturellement à démystifier les technologies de l'information et des communications (TIC) et à les utiliser dans d'autres disciplines, ainsi qu'à jeter les bases d'un perfectionnement professionnel enrichissant prenant en compte l'évolution des technologies informatiques.

Les études informatiques sont pertinentes pour tous les élèves puisqu'elles traitent de la programmation d'outils informatiques (p. ex., tableur, traitement de texte, calculatrice graphique programmable) et de communication (p. ex., téléphone mobile, wiki, page Web) largement répandus, tant dans les entreprises que chez les particuliers. De plus, elles touchent une vaste gamme de compétences transférables en matière de résolution de problèmes, parmi lesquelles on retrouve la définition des besoins, le raisonnement, la conception, la mise en application et l'évaluation, et des habitudes de travail telles que l'initiative, l'organisation et le travail en équipe.

LES OBJECTIFS DU PROGRAMME-CADRE D'ÉTUDES INFORMATIQUES

Le but fondamental du programme-cadre d'études informatiques est d'offrir à l'élève l'occasion d'acquérir des connaissances et des compétences en programmation qui lui permettront d'utiliser les outils informatiques de façon plus efficace dans les autres matières et de se préparer aux programmes d'études postsecondaires ainsi qu'à la vie professionnelle. Ces cours seront aussi une occasion de découverte et d'enrichissement personnel pouvant orienter des choix de carrière ou de loisirs.

Les études informatiques abordent le traitement de l'information par ordinateur. Il ne s'agit pas seulement de bien comprendre le fonctionnement des composants matériels et logiciels d'un ordinateur, mais aussi de développer des logiciels originaux répondant à des problèmes authentiques. Partant de la définition précise des besoins menant à l'élaboration d'un cahier des charges, puis à la modélisation des objets et des processus mis en œuvre et enfin à la programmation et à la maintenance, le processus de développement d'un logiciel fait appel à des compétences très diverses. Pour les besoins du présent document, le terme *études informatiques* comprend l'étude du développement de logiciels, y compris l'étude d'un langage de programmation, l'algorithmique, les pratiques méthodologiques, le fonctionnement du matériel et des logiciels pertinents à cette discipline, ainsi que leur impact sur la société.

Le programme-cadre d'études informatiques vise à préparer l'élève à relever les défis de son avenir en lui permettant :

- d'acquérir une compréhension des concepts informatiques;
- de développer sa capacité à raisonner, à résoudre des problèmes et à communiquer les résultats avec exactitude, éthique et efficacité;
- d'appliquer les connaissances, les compétences et les attitudes acquises en études informatiques à une variété de tâches d'apprentissage et à d'autres matières, et de les mettre en relation avec l'actualité informatique aux niveaux local, national et mondial;
- de développer des habitudes d'apprentissage tout au long de sa vie qui aideront à s'adapter aux nouvelles technologies informatiques dans un monde et dans des milieux professionnels en mutation;
- d'établir des contacts qui aideront à tirer parti des options d'études postsecondaires et de choix de carrière.

Un bon apprentissage dans tous les aspects des études informatiques dépend de l'acquisition de connaissances et d'habiletés dans les quatre grands secteurs suivants :

- langage de programmation
- algorithmes et structures de données
- méthodologie de développement de logiciels
- responsabilité professionnelle et éthique

Le programme-cadre d'études informatiques de la 10^e à la 12^e année offre un ensemble de cours qui sont structurés pour donner aux élèves des bases solides dans ces quatre grands secteurs.

LE RÔLE DE L'ÉLÈVE

Face à la diversité des possibilités d'apprentissage que l'école propose, l'élève a la responsabilité de s'engager résolument et de faire les efforts nécessaires pour réussir. C'est en prenant conscience de ses progrès et du développement de ses habiletés que l'élève sera amené à croire en sa réussite et trouvera la motivation pour assumer cette responsabilité et persévérer dans ses apprentissages. Tous les élèves doivent pouvoir compter sur l'appui et la sollicitude du personnel enseignant et, dans certains cas, sur un soutien supplémentaire.

La maîtrise des connaissances et des habiletés propres au programme d'études informatiques requiert de la part de l'élève un engagement sincère. L'élève devrait saisir toutes les occasions possibles en dehors de la classe pour mieux maîtriser les processus de communication. Ses connaissances et ses habiletés croîtront au fur et à mesure qu'elle ou il explore son environnement et s'engage dans des activités telles que la participation à des forums, la lecture de magazines informatiques et la visite de salons de l'informatique. Les activités d'apprentissage qui lui sont proposées permettent à l'élève de s'engager activement dans sa construction identitaire, dont l'épanouissement culturel constitue une dimension importante. Il importe donc d'amener l'élève à réaliser que la culture comporte de nombreux aspects qui concourent à la richesse de son identité et qu'il lui appartient d'assumer une part de responsabilité à cet égard.

LE RÔLE DES PARENTS

Le rôle des parents¹ dans l'éducation de leur enfant s'articule principalement autour des axes suivants : connaître le curriculum, accompagner leur enfant dans son apprentissage, faire du foyer un milieu d'apprentissage et un lieu d'épanouissement sur le plan culturel.

Connaître le curriculum. L'élève fournit généralement un meilleur rendement scolaire lorsque ses parents s'intéressent à ses études. En se familiarisant avec les programmes-cadres du curriculum, les parents sauront quelles sont les connaissances, les habiletés et les compétences que leur enfant doit acquérir dans chaque cours. En outre, ils pourront mieux suivre le progrès scolaire de leur enfant et en discuter en connaissance de cause. Cela leur permettra aussi de collaborer plus étroitement avec l'enseignante ou l'enseignant en vue d'améliorer le rendement scolaire de leur enfant.

Accompagner leur enfant dans son apprentissage. Les parents peuvent manifester leur intérêt pour l'apprentissage de leur enfant de bien des façons, par exemple, en l'invitant à parler de sa journée à l'école, en l'encourageant à faire ses devoirs, en prenant part aux réunions de parents et en s'assurant que leur enfant dispose d'un endroit approprié à la maison pour effectuer ses travaux. Comme l'apprentissage de leur enfant se fait en français, il est important que les parents valorisent l'acquisition d'une bonne compétence en français en faisant du foyer un milieu stimulant pour l'apprentissage du français. Ils peuvent aussi l'encourager à se tailler une place dans la communauté francophone en Ontario et à assumer ses responsabilités en tant que citoyenne ou citoyen.

1. Dans le présent document, le terme « parents » désigne aussi les tutrices et tuteurs.

Faire du foyer un milieu d'apprentissage. Les parents peuvent encourager leur enfant à participer à des activités qui élargiront ses horizons, enrichiront sa compréhension du monde et développeront son esprit critique, qu'il s'agisse de discuter de nouvelles technologies informatiques, de consulter des ressources en ligne ou de l'accompagner à un salon de l'informatique.

Faire du foyer un lieu d'épanouissement culturel. L'appui des parents est essentiel pour favoriser chez leur enfant le développement de l'identité francophone. Parler français à la maison, prévoir des activités culturelles et récréatives en français et offrir à leur enfant des ressources en français renforce le travail éducatif fait à l'école de langue française et permet à l'enfant de mieux réussir à l'école et de s'identifier à la culture d'expression française dans toute la diversité de ses manifestations.

LE RÔLE DE L'ENSEIGNANTE OU L'ENSEIGNANT

Le rôle de l'enseignante ou l'enseignant, qui consiste à appuyer chaque élève dans sa réussite, s'articule autour de trois axes : créer un milieu d'apprentissage convivial pour l'élève, proposer des activités pertinentes et faire de l'aménagement linguistique en français une priorité.

Créer un milieu d'apprentissage convivial pour l'élève. L'enseignante ou l'enseignant a pour tâche d'élaborer une gamme de stratégies d'enseignement et d'évaluation fondées sur une pédagogie éprouvée. Il lui faut concevoir des stratégies qui tiennent compte des différents styles d'apprentissage et les adapter pour répondre aux divers besoins des élèves. Ces stratégies devraient aussi viser à insuffler à chaque élève le désir d'apprendre et l'inciter à donner son plein rendement.

Proposer des activités pertinentes. Il incombe à l'enseignante ou l'enseignant de saisir toutes les occasions de tisser des liens entre la théorie et la pratique, et de concevoir des activités fondées sur des projets authentiques qui placent l'élève dans une situation d'apprentissage actif. Miser sur les acquis et le concret amène l'élève à découvrir les concepts à l'étude et à les intégrer par l'entremise du questionnement, de la recherche, de l'expérimentation, de l'observation, de la réflexion et de la mise en application. L'enseignante ou l'enseignant l'encouragera à mettre ces concepts en contexte pour lui permettre d'en comprendre la pertinence et l'application dans le monde qui l'entoure.

Faire de l'aménagement linguistique en français une priorité. La qualité de la langue utilisée est garante de la qualité des apprentissages. Il est donc primordial d'accorder la plus grande importance à la qualité de la communication orale et écrite en classe, quelle que soit l'activité d'apprentissage. Il ne s'agit pas de tout corriger et encore moins de culpabiliser l'élève, mais de l'encadrer dans sa prise de parole en situation d'échange. On l'aidera ainsi à développer ses capacités d'expression et à se familiariser avec les techniques de présentation orale et le processus de production écrite. Il faut offrir à l'élève un environnement linguistique cohérent, où tout contribue à enrichir ses compétences en français. En outre, il est essentiel que l'élève dispose de diverses ressources d'apprentissage en français.

LE RÔLE DE LA DIRECTRICE OU DU DIRECTEUR D'ÉCOLE

De concert avec tous les intervenants, la directrice ou le directeur d'école prendra les mesures nécessaires pour fournir la meilleure expérience scolaire possible à tous les élèves et leur donner les moyens de connaître le succès et d'assumer leurs responsabilités sur le plan personnel, civique et professionnel. Il lui incombe de veiller à la mise en œuvre du curriculum dans sa totalité et dans le respect des différents styles d'apprentissage des élèves et, pour ce faire, de s'assurer que les élèves et le personnel enseignant disposent des ressources nécessaires, y compris le perfectionnement professionnel pour favoriser l'excellence de l'enseignement.

La directrice ou le directeur d'école doit valoriser et favoriser l'apprentissage sous toutes ses formes, à l'école comme dans le milieu communautaire. Il lui appartient aussi de concevoir des mesures pour appuyer l'épanouissement d'une culture d'expression française, en conformité avec la politique d'aménagement linguistique du conseil scolaire. À cet égard, la directrice ou le directeur d'école travaille en collaboration avec divers intervenants pour créer une communauté apprenante, laquelle constituera un milieu communautaire où il fait bon vivre et apprendre en français.

La directrice ou le directeur d'école est responsable de s'assurer que l'élève qui a un plan d'enseignement individualisé (PEI) obtient les adaptations et les changements décrits dans son PEI. Il lui incombe aussi de voir à l'élaboration, à la mise en œuvre et au suivi du PEI.

ORGANISATION DU PROGRAMME- CADRE D'ÉTUDES INFORMATIQUES

LES COURS OFFERTS

Le programme-cadre d'études informatiques comprend des cours allant de la 10^e à la 12^e année. Trois types de cours sont offerts : cours préuniversitaires, précollégiaux et ouverts. Les élèves choisissent le type de cours en fonction de leurs champs d'intérêt, de leurs acquis et de leurs objectifs en matière d'études postsecondaires. Les types de cours se définissent comme suit :

- Les *cours préuniversitaires* sont conçus pour permettre à l'élève d'acquérir les connaissances et les habiletés qu'il lui faut pour satisfaire aux critères d'admission des programmes d'études universitaires, et pour préparer sa réussite scolaire et professionnelle.
- Les *cours précollégiaux* sont conçus pour permettre à l'élève d'acquérir les connaissances et les habiletés qu'il lui faut pour satisfaire aux critères d'admission de la plupart des programmes d'études collégiales ou de formation professionnelle.
- Les *cours ouverts* sont conçus pour permettre à l'élève d'élargir ses connaissances et ses habiletés dans des matières qui l'intéressent et qui le préparent à participer de façon active et enrichissante à la société. Ils ne visent pas particulièrement à satisfaire aux exigences des universités, des collèges ni des milieux de travail.

Les cours d'études informatiques proposent aux élèves d'acquérir une meilleure compréhension du monde de l'informatique et de la programmation informatique. Tous les cours offrent également l'occasion de réfléchir sur les enjeux éthiques, environnementaux, économiques, sociaux et professionnels de l'usage du matériel informatique.

Le cours ouvert de 10^e année s'adresse aux élèves désirant acquérir une compréhension générale du fonctionnement d'un ordinateur et de ses composants matériels et logiciels. Il fournit aux élèves la possibilité de développer leurs aptitudes et de résoudre des problèmes concrets en créant des logiciels, des macros, des scripts et des applications Web.

Les cours précollégiaux portent principalement sur l'application des techniques de programmation et la résolution de problèmes concrets tirés du monde professionnel. L'élève développe ainsi une bonne maîtrise d'un langage de programmation orientée objet et d'outils de développement de logiciels et acquiert des habiletés méthodologiques permettant de développer, seul ou en équipe, des logiciels relativement complexes tels qu'un logiciel muni d'une interface utilisateur graphique ou une application Web exploitant des bases de données.

Les cours préuniversitaires offrent une approche plus théorique permettant à l'élève d'explorer les diverses ramifications de l'informatique telles que l'intelligence artificielle et les interfaces homme-machine, tout en bâtissant des bases solides pour des études postsecondaires. Le choix du paradigme de programmation est laissé à l'enseignante ou l'enseignant. De plus, la pratique ne sera pas négligée puisque l'élève mettra en œuvre une méthodologie de développement de logiciels dans le cadre d'un projet en équipe.

Le programme d'études informatiques propose donc aux élèves une sélection de cours qui correspondent à leurs champs d'intérêt et les préparent à suivre des études plus avancées ou à travailler dans le secteur de leur choix. Ces cours sont particulièrement indiqués pour les élèves qui suivent un programme de la Majeure Haute Spécialisation. Que les élèves travaillent par la suite dans le domaine de l'informatique ou tout simplement utilisent des ordinateurs, le programme d'études informatiques leur fournira les bases nécessaires pour prendre des décisions éclairées concernant leurs projets d'avenir.

Cours d'études informatiques, de la 10^e à la 12^e année

Année	Titre du cours	Type	Code	Préalable
10 ^e	Introduction à l'informatique	Ouvert	ICS2O	Aucun
11 ^e	Introduction au génie informatique	Préuniversitaire	ICS3U	Aucun
11 ^e	Introduction à la programmation informatique	Précollégial	ICS3C	Aucun
12 ^e	Génie informatique	Préuniversitaire	ICS4U	Introduction au génie informatique, 11 ^e année, cours préuniversitaire
12 ^e	Programmation informatique	Précollégial	ICS4C	Introduction à la programmation informatique, 11 ^e année, cours précollégial

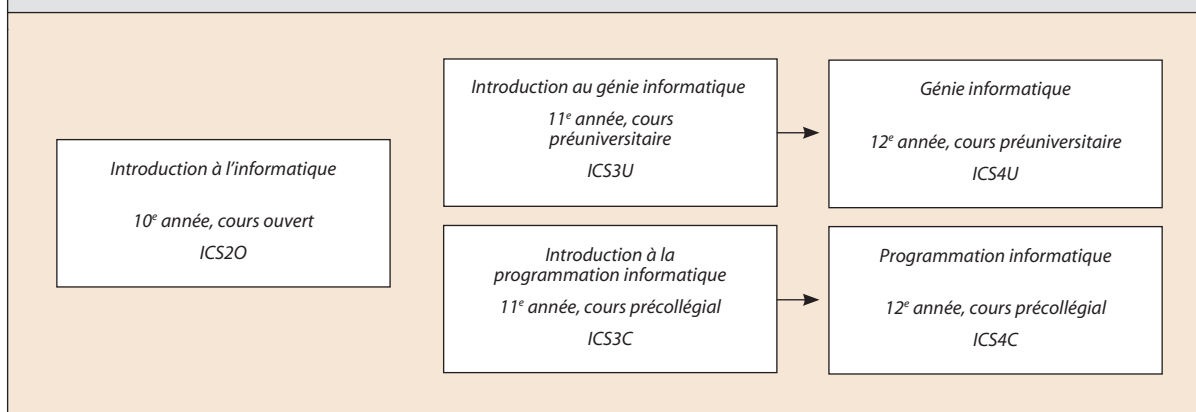
N.B. : Tous les cours décrits dans ce tableau donnent droit à un (1) crédit.

Bien que les cours d'études informatiques soient facultatifs, les élèves ne devraient pas perdre de vue qu'un cours du programme-cadre d'études informatiques, offert de la 10^e à la 12^e année, peut leur servir à satisfaire aux exigences relatives aux crédits obligatoires supplémentaires du Groupe 3 du diplôme d'études secondaires de l'Ontario².

2. Pour satisfaire aux exigences du diplôme d'études secondaires de l'Ontario portant sur le crédit obligatoire supplémentaire du Groupe 3, les élèves peuvent choisir un cours de un (1) crédit d'études informatiques (de la 10^e à la 12^e année), d'éducation technologique (de la 9^e à la 12^e année), de sciences (de 11^e ou 12^e année) ou d'éducation coopérative.

Organigramme des préalables pour les cours d'études informatiques de la 10^e à la 12^e année

Cet organigramme présente l'organisation des cours en fonction des préalables.



Les cours donnant droit à des demi-crédits. Les cours d'études informatiques décrits dans le présent document ont été conçus comme des cours donnant droit à un (1) plein crédit.

Toutefois, à l'exception du cours préuniversitaire de 12^e année, on pourra offrir les cours décrits dans le présent document sous forme de demi-cours valant chacun un demi-crédit (0,5). Les demi-cours exigent un minimum de cinquante-cinq (55) heures d'enseignement et doivent satisfaire aux conditions suivantes :

- Les deux (2) demi-cours élaborés à partir d'un cours donnant droit à un (1) plein crédit doivent ensemble inclure toutes les attentes et les contenus d'apprentissage du cours dont ils sont tirés. Les attentes et les contenus d'apprentissage doivent être répartis entre les deux (2) demi-cours de manière à permettre à l'élève d'acquérir le plus efficacement possible les connaissances et les habiletés dans le temps alloué.
- Un cours préalable à un autre cours du palier secondaire peut aussi être offert sous forme de deux (2) demi-cours. Cependant, l'élève doit réussir les deux (2) demi-cours pour obtenir ce préalable. Si le cours original ne constitue pas un préalable à un cours qu'elle ou il a l'intention de suivre, l'élève n'a pas à suivre les deux (2) demi-cours.
- Le titre de chaque demi-cours doit préciser s'il s'agit de la *Partie 1* ou de la *Partie 2* du cours, selon le cas. La reconnaissance d'un demi-crédit (0,5) sera inscrite dans la colonne de la valeur des crédits du bulletin scolaire et du relevé de notes de l'Ontario.

Les conseils scolaires s'assureront que tous les demi-cours respectent les conditions ci-dessus et signaleront tous les demi-cours au ministère de l'Éducation, dans les rapports d'octobre des écoles.

LES DOMAINES D'ÉTUDE

Chaque cours du programme d'études informatiques est organisé selon des domaines d'étude. En 10^e et 12^e année, la matière est répartie en trois domaines, et en quatre domaines en 11^e année. Ces domaines d'étude sont présentés aux tableaux des pages 15 et 16. Un domaine est commun à tous les cours, soit Enjeux sociétaux et perspectives professionnelles. L'élève y découvrira, entre autres, l'impact de l'industrie de l'informatique sur l'environnement et la santé, et des options de carrière dans ce secteur. Le schéma à la page suivante illustre la présentation des cours d'études informatiques.

La matière à l'étude dans les cours d'études informatiques se répartit en trois ou quatre domaines d'étude (voir pages 15-16).

Les *attentes* décrivent en termes généraux les connaissances et les habiletés que l'élève doit avoir acquises à la fin de chaque cours.

Les contenus d'apprentissage se rattachant à une même attente sont regroupés sous une *rubrique*.

Les *contenus d'apprentissage* décrivent en détail les connaissances et les habiletés que l'élève doit acquérir pour satisfaire aux attentes auxquelles ces contenus se rattachent.

Cours préscoléglial, 11^e année

A. ENVIRONNEMENT INFORMATIQUE DE TRAVAIL

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- A1.** expliquer le fonctionnement d'un ordinateur personnel en utilisant les termes justes en français.
- A2.** appliquer des techniques de gestion de fichiers.
- A3.** utiliser des outils appropriés pour développer des programmes.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Fonctionnement d'un ordinateur personnel

- A1.1** expliquer les fonctions des composants matériels internes d'un ordinateur personnel (p. ex., *carte mère, microprocesseur, mémoire vive, carte vidéo, carte son*).
- A1.2** expliquer les fonctions des périphériques externes qui sont couramment utilisés (p. ex., *souris, clavier, moniteur, imprimante, appareil photo numérique, caméra, clé de mémoire*).
- A1.3** comparer les performances du matériel informatique de divers ordinateurs personnels en utilisant des mesures objectives (p. ex., *fréquence du microprocesseur [en gigahertz], capacité de la mémoire vive [en gigaoctets], consommation électrique [en watts]*).
- A1.4** comparer les caractéristiques majeures des différents types d'accès à Internet (p. ex., *câble de radiotélévision, ligne d'abonné numérique, ligne commutée, accès sans fil*).

Gestion de fichiers

- A2.1** répertorier les fichiers d'un programme en utilisant les fonctions du système d'exploitation.
- A2.2** appliquer une procédure systématique de sauvegarde des fichiers (p. ex., *archivage programmé de fichiers, copie manuelle sur une clé de mémoire ou en ligne, utilisation d'un système de contrôle de versions*).

A2.3 décrire plusieurs types de virus (p. ex., *attaque du système d'amorçage, macro dans un document de bureautique*) et autres programmes malveillants (p. ex., *cheval de Troie, logiciel espion*), leur cycle de vie ainsi que les contre-mesures de sécurité informatique connues (p. ex., *antivirus, coupe-feu, blocage de fenêtres, filtre antipourriel*).

A2.4 expliquer des raisons motivant l'adoption d'une politique d'utilisation des ordinateurs et du réseau local (p. ex., *respect de la vie privée, équité, maintenance informatique*).

A2.5 utiliser les services du réseau local (p. ex., *partage de fichier et d'imprimante, centralisation du code et de l'échéancier*) pour faciliter la gestion et la sauvegarde des fichiers lors du développement d'un programme.

Outils de développement

- A3.1** expliquer les fonctions respectives des systèmes d'exploitation (p. ex., *exécution de programmes, archivage de fichiers, contrôle d'accès*) et des logiciels d'application (p. ex., *traitement de texte, lecture de courriels*), notamment des outils de développement (p. ex., *compilateur, interpréteur*).
- A3.2** expliquer les caractéristiques et les avantages de divers environnements de développement intégrés (EDI) pour développer des programmes.
- A3.3** utiliser les commandes d'un EDI requises pour éditer, exécuter et déboguer des programmes.
- A3.4** utiliser l'aide disponible (p. ex., *fonction d'aide de l'EDI, tutoriel, aide en ligne*) pour développer des programmes.

LE CURRICULUM DE L'ONTARIO, DE LA 10^e À LA 12^e ANNÉE | Études informatiques

54

Bon nombre de contenus d'apprentissage proposent à titre indicatif des *exemples* présentés entre parenthèses et en italique. Ces exemples illustrent des habiletés, des situations ou des outils.

LES ATTENTES ET LES CONTENUS D'APPRENTISSAGE

Les attentes et les contenus d'apprentissage de chaque cours décrivent les connaissances et les habiletés que l'élève démontrera dans son travail de classe, dans ses recherches ainsi que dans ses travaux, ses examens ou toute autre activité qui sert à évaluer son rendement.

À chaque domaine d'étude correspondent des attentes et des contenus d'apprentissage.

- Les *attentes* décrivent en termes généraux les connaissances et les habiletés que l'élève doit avoir acquises à la fin de chaque cours.
- Les *contenus d'apprentissage* décrivent en détail ces connaissances et ces habiletés que l'élève doit maîtriser pour satisfaire aux attentes. Les contenus d'apprentissage se rattachant à une même attente sont groupés sous une même rubrique et numérotés (p. ex., « B2.1 », pour désigner le premier contenu d'apprentissage se rapportant à l'attente 2 du domaine d'étude B).

Les contenus d'apprentissage sont répartis en plusieurs rubriques qui portent chacune sur des aspects particuliers des connaissances et des habiletés mentionnées dans les attentes. Cette répartition pourra aider le personnel enseignant à planifier les activités d'apprentissage. Cependant, le fait d'organiser les cours selon des domaines d'étude et des rubriques ne signifie pas que les attentes et les contenus d'apprentissage d'un domaine ou d'une rubrique doivent être abordés séparément. Au contraire, le personnel enseignant devrait intégrer des attentes et des contenus d'apprentissage de divers domaines d'étude et rubriques lorsque cela s'applique.

Le tableau ci-dessous illustre les domaines d'étude et les rubriques correspondantes pour le cours de 10^e année.

	Environnement informatique de travail	Introduction à la programmation	Enjeux sociétaux et perspectives professionnelles
Rubriques	Matériel informatique	Techniques de programmation	Environnement et santé
	Logiciels		
	Système d'exploitation	Développement de programmes	Enjeux sociétaux
	Réseaux locaux		
	Maintenance informatique	Assurance-qualité logicielle	Carrière et formation

Le tableau ci-dessous illustre les domaines d'étude et les rubriques correspondantes pour les cours de 11^e et de 12^e année.

Domaines d'étude :		Environnement informatique de travail	Techniques de programmation		Développement de logiciels	Enjeux sociétaux et perspectives professionnelles
Rubriques			Filière préuniversitaire	Filière précollégiale		
	11 ^e année	Fonctionnement d'un ordinateur personnel	Syntaxe et sémantique	Traitement des données	Techniques de développement de logiciels	Environnement et santé
		Gestion de fichiers	Algorithmes et structures de données	Structures de contrôle	Conception d'algorithmes	Carrière et formation
		Outils de développement	Assurance-qualité logicielle	Assurance-qualité logicielle	Développement de programmes	
	12 ^e année		Syntaxe et sémantique	Programmation orientée objet	Méthodologie de développement de logiciels	Enjeux sociétaux
			Algorithmes et structures de données	Traitement des données	Conception d'algorithmes	Progrès en informatique
			Documentation d'un logiciel	Documentation d'un logiciel	Développement de logiciels	Carrière et formation

Bon nombre de contenus d'apprentissage proposent à titre indicatif des exemples entre parenthèses. Ces exemples illustrent la nature ou le type des connaissances ou des habiletés précisées dans les contenus d'apprentissage, de même que le degré de complexité visé. Les enseignantes et enseignants peuvent s'en inspirer, choisir d'utiliser des exemples adaptés à leurs groupes classes ou encore élaborer leur propre stratégie en adoptant un niveau de complexité similaire. Quelles que soient les méthodes utilisées pour mettre en œuvre les exigences énoncées dans les contenus d'apprentissage, elles doivent, autant que possible, être inclusives et tenir compte de la diversité au sein du groupe classe.

ÉVALUATION DU RENDEMENT DE L'ÉLÈVE

LE PROCESSUS D'ÉVALUATION DU RENDEMENT DE L'ÉLÈVE

L'objectif premier de l'évaluation consiste à améliorer l'apprentissage de l'élève. Les données recueillies au moyen de l'évaluation aident l'enseignante ou l'enseignant à cerner les points forts et les points à améliorer de l'élève par rapport aux attentes visées. Ces données permettent aussi à l'enseignante ou l'enseignant d'adapter le programme et les approches pédagogiques aux besoins de l'élève et d'en évaluer l'efficacité globale.

Le processus d'évaluation consiste d'abord à recueillir des données provenant de diverses sources, notamment les présentations, les projets, les activités et les tests qui témoignent jusqu'à quel point l'élève satisfait aux attentes. L'enseignante ou l'enseignant peut donner à l'élève une rétroaction descriptive qui la ou le guidera dans ses efforts pour s'améliorer. Il s'agit ensuite de juger de la qualité du travail de l'élève en fonction des critères établis et d'y attribuer une valeur.

L'enseignante ou l'enseignant fondera l'évaluation sur les attentes du curriculum en se servant de la grille d'évaluation du programme-cadre, conformément aux consignes énoncées dans le présent document.

Pour assurer la validité et la fiabilité de l'évaluation ainsi que pour favoriser l'amélioration du rendement scolaire, l'enseignante ou l'enseignant doit utiliser des stratégies d'évaluation qui :

- portent sur la matière enseignée et sur la qualité de l'apprentissage de l'élève;
- sont fondées sur la grille d'évaluation du rendement (p. 20-21) mettant en relation quatre grandes compétences et les descriptions des niveaux de rendement;
- sont diversifiées et échelonnées tout au long du cours pour donner à l'élève de multiples possibilités de démontrer l'étendue de son apprentissage;
- conviennent aux activités d'apprentissage, aux attentes et aux contenus d'apprentissage de même qu'aux besoins et aux expériences de l'élève;
- sont justes pour tous les élèves;
- tiennent compte des besoins de l'élève en difficulté, conformément aux stratégies décrites dans son plan d'enseignement individualisé (PEI);
- tiennent compte des besoins de l'élève inscrit au programme d'actualisation linguistique en français (ALF) ou de perfectionnement du français (PDF);
- favorisent la capacité de l'élève à s'autoévaluer et à se fixer des objectifs précis;
- reposent sur un échantillonnage de travaux de l'élève illustrant bien son niveau de rendement;

- servent à communiquer à l'élève la direction à prendre pour améliorer son rendement;
- sont communiquées clairement à l'élève et aux parents au début du cours et à tout autre moment approprié durant l'année scolaire.

Les critères indiqués au niveau 3 de la grille de rendement (p. 20-21) représentent la « norme provinciale » de rendement par rapport aux attentes d'un cours. On peut avoir une idée bien détaillée du rendement général au niveau 3 pour un cours d'études informatiques en examinant de haut en bas la colonne ombrée du tableau du rendement ayant pour titre « 70-79 % (Niveau 3) ». Le personnel enseignant et les parents peuvent considérer que l'élève ayant un rendement de niveau 3 sera bien préparé pour le cours suivant.

Le niveau 1, bien qu'il indique une réussite, signifie que l'élève a démontré un rendement inférieur à la norme provinciale. Le niveau 2 indique un rendement moyen qui se rapproche de la norme provinciale. Le niveau 4 signifie que le rendement de l'élève est supérieur à la norme provinciale. Cependant, cela ne veut pas dire que l'élève dépasse les attentes du cours, mais plutôt qu'elle ou il démontre une compréhension plus approfondie de la matière que l'élève dont le rendement se situe au niveau 3.

LA GRILLE D'ÉVALUATION DU RENDEMENT

La grille d'évaluation du rendement en études informatiques sera utilisée par le personnel enseignant de toute la province. Elle lui permettra de porter un jugement sur le rendement de l'élève basé sur des niveaux de rendement clairs et précis et sur des données recueillies sur une période prolongée.

La grille d'évaluation du rendement vise à :

- fournir un cadre commun qui couvre les attentes pour tous les cours du présent programme-cadre;
- guider l'enseignante ou l'enseignant lors de l'élaboration d'instruments de mesure, y compris des grilles adaptées;
- guider l'enseignante ou l'enseignant dans la planification de son enseignement;
- communiquer à l'élève ses points forts et ceux à améliorer;
- préciser les compétences et les critères d'après lesquels sera évalué le rendement de l'élève.

La grille porte sur les quatre *compétences* suivantes : Connaissance et compréhension, Habiletés de la pensée, Communication et Mise en application. Ces compétences couvrent l'ensemble des éléments à l'étude et des habiletés visés par les attentes et les contenus d'apprentissage. Elles sont précisées par des critères clairs et sont complémentaires les unes des autres. L'enseignante ou l'enseignant doit déterminer quelles compétences utiliser pour évaluer la satisfaction des attentes. Les compétences doivent être mesurées et évaluées de manière équilibrée tout au long du cours. De plus, il est essentiel de donner à l'élève des occasions multiples et diverses de démontrer jusqu'à quel point elle ou il a satisfait aux attentes et ce, pour chacune des quatre compétences.

Les compétences sont définies comme suit :

- La compétence *Connaissance et compréhension* est la construction du savoir propre à la discipline, soit la connaissance des éléments à l'étude et la compréhension de leur signification et de leur portée.

- La compétence **Habiletés de la pensée** est l'utilisation d'un ensemble d'habiletés liées aux processus de la pensée critique et de la pensée créative (p. ex., conception d'un algorithme, test d'un programme, analyse des enjeux éthiques, création d'une interface utilisateur graphique). Elle comprend aussi les habiletés liées à la planification (p. ex., identification d'un besoin, définition d'une tâche, élaboration d'un calendrier de soumission des travaux) et au traitement de l'information (p. ex., analyse d'un problème, interprétation d'un diagramme, analyse des besoins).
- La compétence **Communication** est la transmission des idées et de l'information selon différentes formes et divers moyens. L'information et les idées peuvent être transmises de façon orale (p. ex., exposé), de façon écrite (p. ex., code source documenté) ou visuelle (p. ex., organigramme).
- La compétence **Mise en application** est l'application des éléments à l'étude et des habiletés dans des contextes familiers (p. ex., utilisation des concepts de programmation), leur transfert à de nouveaux contextes (p. ex., développement de programmes pour résoudre différents problèmes) et l'établissement de liens (p. ex., impact de l'industrie de l'informatique sur l'environnement).

Dans la grille d'évaluation du rendement, une série de **critères** viennent préciser davantage chaque compétence et définissent les dimensions du rendement de l'élève qui sont évaluées. Par exemple, le premier critère sous la compétence Connaissance et compréhension est la « connaissance des éléments à l'étude » (p. ex., composants matériels de l'ordinateur, types de données).

Les **descripteurs** permettent à l'enseignante ou l'enseignant de poser un jugement professionnel sur la qualité du rendement de l'élève et de lui donner une rétroaction descriptive. Dans la grille d'évaluation du rendement, le type de descripteur utilisé pour tous les critères des trois dernières compétences de la grille est l'efficacité. On définit l'**efficacité** comme la capacité de réaliser entièrement le résultat attendu. L'enseignante ou l'enseignant pourra se servir d'autres types de descripteur (p. ex., la *convenance*, la *clarté*, l'*exactitude*, la *précision*, la *logique*, la *pertinence*, la *cohérence*, la *souplesse*, la *profondeur*, l'*envergure*) en fonction de la compétence et du critère visés lorsqu'elle ou il élaborera des grilles adaptées. Par exemple, l'enseignante ou l'enseignant pourrait déterminer le niveau d'efficacité pour la compétence Habiletés de la pensée en évaluant le niveau logique d'une analyse; pour la compétence Communication, elle ou il pourrait évaluer le niveau de clarté de la communication des idées; pour la compétence Mise en application, elle ou il pourrait évaluer la convenance et l'envergure des liens établis. De la même façon pour la compétence Connaissance et compréhension, l'évaluation de la connaissance des éléments à l'étude pourrait porter sur l'exactitude des faits, tandis que celle de la compréhension des éléments à l'étude pourrait porter sur la profondeur d'une explication.

L'**échelle de progression** (p. ex., avec une efficacité limitée, avec une certaine efficacité, avec efficacité ou avec beaucoup d'efficacité) qualifie le rendement de l'élève à chacun des niveaux de la grille. Par exemple, pour un élève dont le rendement se situe au niveau 3 par rapport au premier critère de la compétence Habiletés de la pensée, on dirait qu'elle ou il « utilise les habiletés de planification avec efficacité ».

GRILLE D'ÉVALUATION DU RENDEMENT EN ÉTUDES INFORMATIQUES

Compétences	50-59 % (Niveau 1)	60-69 % (Niveau 2)	70-79 % (Niveau 3)	80-100 % (Niveau 4)
Connaissance et compréhension – La construction du savoir propre à la discipline, soit la connaissance des éléments à l'étude et la compréhension de leur signification et de leur portée.				
	L'élève :			
Connaissance des éléments à l'étude (p. ex., composants matériels de l'ordinateur, types de données).	démontre une connaissance limitée des éléments à l'étude.	démontre une connaissance partielle des éléments à l'étude.	démontre une bonne connaissance des éléments à l'étude.	démontre une connaissance approfondie des éléments à l'étude.
Compréhension des éléments à l'étude (p. ex., conception modulaire, algorithme).	démontre une compréhension limitée des éléments à l'étude.	démontre une compréhension partielle des éléments à l'étude.	démontre une bonne compréhension des éléments à l'étude.	démontre une compréhension approfondie des éléments à l'étude.
Habiletés de la pensée – L'utilisation d'un ensemble d'habiletés liées aux processus de la pensée critique et de la pensée créative.				
	L'élève :			
Utilisation des habiletés de planification (p. ex., identification d'un besoin, définition d'une tâche, élaboration d'un calendrier de soumission des travaux).	utilise les habiletés de planification avec une efficacité limitée.	utilise les habiletés de planification avec une certaine efficacité.	utilise les habiletés de planification avec efficacité.	utilise les habiletés de planification avec beaucoup d'efficacité.
Utilisation des habiletés de traitement de l'information (p. ex., analyse d'un problème, interprétation d'un diagramme, analyse des besoins).	utilise les habiletés de traitement de l'information avec une efficacité limitée.	utilise les habiletés de traitement de l'information avec une certaine efficacité.	utilise les habiletés de traitement de l'information avec efficacité.	utilise les habiletés de traitement de l'information avec beaucoup d'efficacité.
Utilisation des processus de la pensée critique et de la pensée créative (p. ex., conception d'un algorithme, test d'un programme, analyse des enjeux éthiques, création d'une interface utilisateur graphique).	utilise les processus de la pensée critique et de la pensée créative avec une efficacité limitée.	utilise les processus de la pensée critique et de la pensée créative avec une certaine efficacité.	utilise les processus de la pensée critique et de la pensée créative avec efficacité.	utilise les processus de la pensée critique et de la pensée créative avec beaucoup d'efficacité.

GRILLE D'ÉVALUATION DU RENDEMENT EN ÉTUDES INFORMATIQUES (suite)

Compétences	50-59 % (Niveau 1)	60-69 % (Niveau 2)	70-79 % (Niveau 3)	80-100 % (Niveau 4)
Communication – La transmission des idées et de l'information selon différentes formes et divers moyens.				
	L'élève :			
Expression et organisation des idées et de l'information (p. ex., documentation d'un logiciel, découpage d'un programme en plusieurs fichiers, catalogue de fonctions).	exprime et organise les idées et l'information avec une efficacité limitée.	exprime et organise les idées et l'information avec une certaine efficacité.	exprime et organise les idées et l'information avec efficacité.	exprime et organise les idées et l'information avec beaucoup d'efficacité.
Communication des idées et de l'information de façon orale (p. ex., exposé), écrite (p. ex., code source documenté) et visuelle (p. ex., organigramme), à des fins précises et pour des auditoires spécifiques.	communique les idées et l'information à des fins précises et pour des auditoires spécifiques avec une efficacité limitée.	communique les idées et l'information à des fins précises et pour des auditoires spécifiques avec une certaine efficacité.	communique les idées et l'information à des fins précises et pour des auditoires spécifiques avec efficacité.	communique les idées et l'information à des fins précises et pour des auditoires spécifiques avec beaucoup d'efficacité.
Utilisation des conventions (p. ex., syntaxe d'un langage de programmation, règle de mise en page, règle de nomenclature des noms de variables) et de la terminologie à l'étude.	utilise les conventions et la terminologie à l'étude avec une efficacité limitée.	utilise les conventions et la terminologie à l'étude avec une certaine efficacité.	utilise les conventions et la terminologie à l'étude avec efficacité.	utilise les conventions et la terminologie à l'étude avec beaucoup d'efficacité.
Mise en application – L'application des éléments à l'étude et des habiletés dans des contextes familiers, leur transfert à de nouveaux contextes et l'établissement de liens.				
	L'élève :			
Application des connaissances et des habiletés dans des contextes familiers (p. ex., utilisation des concepts de programmation).	applique les connaissances et les habiletés dans des contextes familiers avec une efficacité limitée.	applique les connaissances et les habiletés dans des contextes familiers avec une certaine efficacité.	applique les connaissances et les habiletés dans des contextes familiers avec efficacité.	applique les connaissances et les habiletés dans des contextes familiers avec beaucoup d'efficacité.
Transfert des connaissances et des habiletés dans de nouveaux contextes (p. ex., développement de programmes pour résoudre différents problèmes).	transfère les connaissances et les habiletés à de nouveaux contextes avec une efficacité limitée.	transfère les connaissances et les habiletés à de nouveaux contextes avec une certaine efficacité.	transfère les connaissances et les habiletés à de nouveaux contextes avec efficacité.	transfère les connaissances et les habiletés à de nouveaux contextes avec beaucoup d'efficacité.
Établissement de liens (p. ex., impact de l'industrie de l'informatique sur l'environnement).	établit des liens avec une efficacité limitée.	établit des liens avec une certaine efficacité.	établit des liens avec efficacité.	établit des liens avec beaucoup d'efficacité.

LA COMMUNICATION DU RENDEMENT

Le bulletin scolaire de l'Ontario de la 9^e à la 12^e année doit servir à communiquer officiellement à l'élève et à ses parents le rendement scolaire fourni.

Compte rendu de la satisfaction des attentes. Le bulletin scolaire dresse un bilan du rendement que l'élève a fourni par rapport aux attentes des cours suivis, pendant une période déterminée du semestre ou de l'année scolaire, sous forme de notes exprimées en pourcentage. La note en pourcentage représente la qualité du rendement global de l'élève en fonction des attentes du cours et indique le niveau de rendement correspondant dans la grille d'évaluation de la discipline.

Une note finale est inscrite à la fin de chaque cours et le crédit correspondant est accordé si l'élève a obtenu une note de 50 % ou plus. Pour chaque cours de la 9^e à la 12^e année, la note finale sera déterminée comme suit :

- Soixante-dix pour cent (70 %) de la note de chaque cours sera fondé sur les évaluations effectuées tout au long du cours. Cette portion de la note devrait refléter le niveau de rendement le plus fréquent durant le cours, bien qu'il faille accorder une attention particulière aux niveaux de rendement les plus récents.
- Trente pour cent (30 %) de la note sera fondé sur l'évaluation finale, sous forme d'examen, de travail, de recherche ou de tout autre mode d'évaluation approprié. Cette évaluation aura lieu vers la fin du cours.

Compte rendu sur les habiletés d'apprentissage à développer. Le bulletin scolaire rend compte des habiletés d'apprentissage démontrées par l'élève dans chacun des cours, dans les six catégories suivantes : l'utilisation du français parlé, l'autonomie, la collaboration en équipe, l'organisation, les habitudes de travail/devoirs et l'initiative. Ces habiletés d'apprentissage sont évaluées au moyen d'une échelle à quatre degrés (E – excellent, T – très bien, S – satisfaisant, N – amélioration nécessaire). La décision d'évaluer et de rendre compte de façon distincte des habiletés d'apprentissage dans ces six catégories est fondée sur leur rôle essentiel dans la capacité des élèves de réaliser les attentes des cours. L'évaluation des habiletés d'apprentissage, sauf celles qui peuvent faire partie intégrante des attentes du cours, ne doit pas être prise en considération dans la détermination des notes en pourcentage, car celles-ci devraient uniquement représenter la mesure dans laquelle l'élève a satisfait aux attentes du cours. Les politiques relatives à ce sujet sont tracées dans le *Guide du bulletin scolaire de l'Ontario de la 9^e à la 12^e année, 1999*. Ce document est affiché sur le site Web du ministère de l'Éducation au www.edu.gov.on.ca.

CONSIDÉRATIONS CONCERNANT LA PLANIFICATION DU PROGRAMME

L'enseignante ou l'enseignant doit planifier son enseignement et l'apprentissage des élèves en études informatiques en prêtant une attention toute particulière à la différenciation pédagogique et en tenant compte des diverses considérations présentées aux pages suivantes.

La *différenciation pédagogique* est une approche souple et proactive qui place l'élève au cœur de son apprentissage et crée un environnement propice à la quête de son identité francophone. Cette approche offre des pistes de soutien au modèle francophone de l'école de la réussite en permettant à l'enseignante ou l'enseignant de faire un choix judicieux de stratégies pédagogiques et d'y apporter des ajustements en fonction des niveaux de préparation et des champs d'intérêt des élèves et de leurs préférences en matière d'apprentissage.

LES STRATÉGIES D'ENSEIGNEMENT ET D'APPRENTISSAGE

Le personnel enseignant choisit les stratégies pédagogiques qu'il juge les plus appropriées dans le but d'optimiser le rendement des élèves. Les stratégies pédagogiques utilisées dans les cours d'études informatiques mettent généralement l'accent sur la réalisation de projets et sur l'exploration des technologies informatiques. Les élèves apprennent davantage lorsqu'ils sont placés dans un rôle d'acteur, mis au défi de résoudre un problème à leur mesure. Par conséquent, les attentes qui suivent concernent principalement l'application de techniques, l'utilisation d'outils, la conception et le développement de logiciels. Il est entendu que l'élève qui satisfait à ces attentes est également capable d'expliquer et de justifier son travail. D'ailleurs, les productions de l'élève comprennent les produits livrables d'une méthodologie de développement de logiciels et le code source qui se doit d'être documenté dans les règles de l'art. L'élève expérimentera donc les pratiques modèles de l'industrie de l'informatique avec l'aide des enseignantes et enseignants.

L'élève apprend mieux lorsqu'on lui offre un éventail d'activités d'apprentissage. Il est donc souhaitable d'encourager l'élève à faire des recherches, à développer son esprit critique, à proposer des solutions technologiques originales, à partager ses connaissances et à travailler en équipe. En outre, le travail d'équipe devrait revêtir une part particulièrement importante des activités d'une ou d'un élève de 12^e année puisque les logiciels sont

de plus en plus rarement le fruit d'un travail isolé. En effet, les effets de la mondialisation tels que le recours à la pratique de l'externalisation ainsi que la popularisation des communautés de logiciels libres, a fait que les programmeuses et programmeurs sont souvent amenés à travailler au moyen d'Internet avec des collègues qui résident dans des fuseaux horaires éloignés.

L'IMPORTANCE DE L'ACTUALITÉ DANS LES COURS D'ÉTUDES INFORMATIQUES

L'actualité informatique, en particulier celle qui est véhiculée par des médias francophones, permet de mettre en contexte et de valoriser les notions apprises en classe. Les technologies émergentes, les produits commerciaux tels que les derniers appareils mobiles et les nouveaux médias constituent autant de raisons de se passionner pour l'informatique. Ce sont aussi des opportunités idéales pour vivre des discussions animées et ainsi permettre à l'élève de tisser des liens avec le monde professionnel et la société.

LA PLACE DES TECHNOLOGIES DANS LES COURS D'ÉTUDES INFORMATIQUES

Les technologies de l'information et des communications (TIC) offrent une gamme d'outils qui peuvent grandement élargir et enrichir les stratégies d'enseignement du personnel enseignant et appuyer l'apprentissage des élèves. Ces outils comprennent, entre autres, des ressources multimédias, des bases de données, des sites Web, des caméras numériques et des logiciels de traitement de texte. Ils peuvent aider les élèves à collecter, organiser et classer l'information recueillie et puis à écrire, réviser et présenter leurs travaux. Les TIC peuvent aussi être utilisées pour permettre aux élèves de communiquer avec des élèves d'autres écoles et pour accéder virtuellement à la communauté mondiale à partir de la salle de classe.

Dans les cours d'études informatiques, les enseignantes et enseignants doivent s'assurer que les TIC sont abordées non seulement du point de vue de l'utilisatrice ou de l'utilisateur, mais aussi du point de vue de la conceptrice ou du concepteur. Par exemple, les élèves peuvent être amenés à développer une interface ou une applique pour le site Web de l'école. Des ressources et des outils tels que des tutoriels ou des extraits de code source, des bases de données concernant le matériel ou la recherche informatique, des environnements de développement intégrés et des logiciels servant à gérer des projets informatiques sont utilisés. Les enseignantes et enseignants peuvent aussi choisir d'intégrer des plateformes technologiques permettant le développement rapide de logiciels.

Il faut encourager les élèves à utiliser les TIC chaque fois que cela est approprié. Cependant, l'utilisation des TIC présente aussi des dangers et des risques dont les élèves doivent prendre conscience. Il faut donc les sensibiliser aux questions concernant leur sécurité, leur droit à la vie privée ainsi qu'aux enjeux éthiques posés par ces technologies.

LA PLANIFICATION DES COURS D'ÉTUDES INFORMATIQUES DESTINÉS AUX ÉLÈVES EN DIFFICULTÉ

Comme il incombe aux enseignantes et enseignants d'aider tous les élèves à apprendre, leur rôle dans l'éducation des élèves en difficulté est primordial. Afin de leur permettre d'assumer pleinement ce rôle, un personnel enseignant spécialisé en éducation de l'enfance en difficulté est mis à leur disposition. À cet égard, le rapport intitulé

Transformation de l'éducation de l'enfance en difficulté : Rapport des coprésidentes avec les recommandations de la Table de concertation sur l'éducation de l'enfance en difficulté, 2006, recommandait une série de principes sur lesquels doit reposer la planification des programmes destinés aux élèves en difficulté. Il importe donc que celles et ceux qui planifient les cours d'études informatiques y accordent une attention toute particulière.

Ce rapport réitère sept grands principes directeurs :

- Tous les élèves peuvent réussir.
- La conception universelle de l'apprentissage³ et la différenciation pédagogique sont des moyens pour répondre aux besoins d'apprentissage et de réussite de tout groupe d'élèves.
- Des pratiques réussies d'enseignement s'appuient sur les recherches et les expériences vécues.
- Les enseignantes et enseignants sont les acteurs clés pour l'acquisition de la littératie et de la numératie par les élèves.
- Chaque enfant possède son propre style d'apprentissage.
- Le personnel enseignant a besoin de l'appui de la communauté pour créer un milieu d'apprentissage favorable aux élèves ayant des besoins particuliers.
- Chaque élève est unique.

Les élèves de toute salle de classe présentent collectivement un ensemble de styles d'apprentissage et de besoins d'apprentissage. Il appartient au personnel enseignant de planifier des programmes en fonction de cette diversité et de confier aux élèves des tâches correspondant à leurs habiletés pour que chaque élève puisse bénéficier au maximum du processus d'enseignement et d'apprentissage. Le recours à des groupes souples dans le cadre de l'enseignement et l'évaluation continue constituent des composantes importantes des programmes qui tiennent compte de la diversité des besoins en apprentissage.

Au moment de la planification du programme d'études informatiques à l'intention de l'élève en difficulté, l'enseignante ou l'enseignant devrait examiner le niveau de rendement actuel de l'élève, les points forts et les besoins en apprentissage de l'élève, de même que les connaissances et les habiletés qui sont attendues de la part des élèves à la fin du cours, afin de déterminer laquelle des options suivantes est la plus appropriée :

- aucune adaptation⁴ ou modification;
- adaptations seulement;
- attentes modifiées et adaptations au besoin;
- attentes différentes – qui ne découlent pas des attentes prescrites des cours du présent programme-cadre.

Si l'élève requiert des adaptations, des attentes modifiées ou une combinaison des deux, les renseignements pertinents figurant aux paragraphes ci-dessous doivent être consignés dans son plan d'enseignement individualisé (PEI). Pour en savoir davantage

3. La conception universelle de l'apprentissage a pour but de créer un milieu d'apprentissage ouvert et accessible à tous les élèves, sans égard à l'âge, aux habiletés ou à la situation. L'enseignement basé sur les principes de la conception universelle de l'apprentissage se caractérise par sa flexibilité et sa relation d'aide, se prête à diverses adaptations pour combler les besoins particuliers des élèves et permet à tous les élèves d'avoir accès au curriculum dans toute la mesure du possible.

4. Les adaptations désignent des stratégies d'enseignement et d'évaluation individualisées, un soutien fourni par du personnel ou par un équipement personnalisé.

sur les exigences du ministère de l'Éducation au sujet des PEI, veuillez consulter le document intitulé *Plan d'enseignement individualisé – Normes pour l'élaboration, la planification des programmes et la mise en œuvre, 2000* (appelé ci-après *Normes du PEI, 2000*). On trouvera des renseignements plus détaillés sur la planification des programmes pour l'enfance en difficulté dans le document intitulé *Plan d'enseignement individualisé – Guide, 2004* (appelé ci-après *Guide du PEI, 2004*). Ces deux documents sont affichés sur le site Web du ministère de l'Éducation au www.edu.gov.on.ca.

L'élève en difficulté qui ne requiert que des adaptations. Certains élèves en difficulté peuvent suivre le curriculum prévu pour le cours et démontrer un apprentissage autonome si on leur fournit des adaptations. Les adaptations facilitent l'accès au cours sans avoir à modifier les connaissances et les habiletés que l'élève doit manifester. Les adaptations requises pour faciliter l'apprentissage de l'élève doivent être inscrites dans le PEI (voir page 11 des *Normes du PEI, 2000*). Les mêmes adaptations seront probablement inscrites dans le PEI pour plusieurs cours, voire tous les cours.

Offrir des adaptations aux élèves en difficulté devrait être la première option envisagée dans le cadre de la planification des programmes. Les élèves en difficulté peuvent réussir lorsqu'on leur offre des adaptations appropriées. La prestation de l'enseignement axé sur la conception universelle et la différenciation pédagogique met l'accent sur la disponibilité des adaptations permettant de satisfaire les besoins divers des apprenantes et apprenants.

Il existe trois types d'adaptations :

- *Les adaptations pédagogiques* désignent les changements apportés aux stratégies d'enseignement, tels que les styles de présentation, les méthodes d'organisation et l'utilisation d'outils technologiques et du multimédia.
- *Les adaptations environnementales* désignent les changements apportés à la salle de classe ou au milieu scolaire, tels que la désignation préférentielle d'une place ou le recours à un éclairage particulier.
- *Les adaptations en matière d'évaluation* désignent les changements apportés aux stratégies d'évaluation pour permettre à l'élève de démontrer son apprentissage. Par exemple, on pourrait lui donner plus de temps pour terminer les examens ou ses travaux scolaires, ou lui permettre de répondre oralement à des questions d'examen (pour d'autres exemples, voir la page 33 du *Guide du PEI, 2004*).

Si seules des adaptations sont nécessaires dans les cours d'études informatiques, le rendement de l'élève sera évalué par rapport aux attentes du cours et par rapport aux niveaux de rendement décrits dans le présent document. Sur le bulletin scolaire de l'Ontario, la case du PEI ne sera pas cochée et on n'inclura pas d'information sur l'offre d'adaptations.

L'élève en difficulté qui requiert des attentes modifiées. Certains élèves en difficulté auront besoin d'attentes et de tâches modifiées qui ne correspondent pas aux attentes et aux contenus d'apprentissage dans le cours. Dans la plupart des cas, ces attentes modifiées seront fondées sur la matière du cours, mais refléteront des changements en ce qui a trait à leur nombre et à leur complexité. Les attentes modifiées représentent des réalisations précises, réalistes, observables et mesurables, et décrivent les connaissances ou les habiletés précises que l'élève peut démontrer de façon autonome en utilisant, au besoin, des adaptations en matière d'évaluation.

Il est important de vérifier l'étendue des modifications apportées aux attentes et de les noter clairement dans le PEI. Tel qu'indiqué dans la section 7.12 du document de politique ministériel intitulé *Les écoles secondaires de l'Ontario de la 9^e à la 12^e année – Préparation au diplôme d'études secondaires de l'Ontario, 1999*, il reviendra à la directrice ou au directeur d'école de déterminer si la réalisation des attentes modifiées fondées sur le niveau de rendement actuel de l'élève signifie que l'élève a réussi le cours et si l'élève peut recevoir un crédit pour le cours. La directrice ou le directeur d'école informera les parents et l'élève de sa décision.

Lorsqu'on s'attend à ce qu'un élève satisfasse à la plupart des attentes du curriculum d'un cours, les attentes modifiées devraient indiquer comment les connaissances, les habiletés et les tâches de l'élève différeront de celles des autres élèves suivant ce cours. Lorsque les modifications sont si étendues que la réalisation des attentes d'apprentissage (connaissances, habiletés) ne donnerait probablement pas droit à un (1) crédit, les attentes devraient préciser les exigences précises ou les tâches d'après lesquelles le rendement de l'élève sera évalué et en fonction desquelles une note pour le cours sera inscrite dans le bulletin scolaire de l'Ontario.

Les attentes modifiées indiquent les connaissances ou les habiletés que l'élève devrait pouvoir démontrer et qui seront évaluées lors de chaque période visée par le bulletin scolaire (voir pages 10 et 11 des *Normes du PEI, 2000*). Les attentes de l'élève doivent être revues au moins une fois pour chaque période visée par le bulletin scolaire et, au besoin, être mises à jour à la lumière des progrès accomplis par l'élève (voir page 11 des *Normes du PEI, 2000*).

Si l'élève requiert des attentes modifiées en études informatiques, l'évaluation de son rendement sera fondée sur les attentes inscrites dans son PEI et sur les niveaux de rendement décrits dans le présent document. Si certaines des attentes d'un élève pour un cours sont modifiées, mais que l'élève essaie d'obtenir un (1) crédit pour ce cours, il faut cocher la case PEI sur le bulletin scolaire de l'Ontario. Cependant, si la directrice ou le directeur d'école estime que les attentes modifiées ne permettent pas d'accorder un (1) crédit à l'élève pour le cours, la case PEI doit être cochée et on doit inscrire l'énoncé approprié du *Guide du bulletin scolaire de l'Ontario de la 9^e à la 12^e année, 1999* (voir page 7). Les commentaires de l'enseignante ou l'enseignant devraient comprendre des renseignements pertinents sur la capacité de l'élève à démontrer qu'elle ou il a satisfait aux attentes modifiées. L'enseignante ou l'enseignant doit aussi indiquer les prochaines étapes de l'apprentissage de l'élève dans le cadre du cours.

L'ÉLÈVE BÉNÉFICIAIRE DES PROGRAMMES D'ACTUALISATION LINGUISTIQUE EN FRANÇAIS OU DE PERFECTIONNEMENT DU FRANÇAIS

L'école de langue française tient compte de la diversité linguistique, scolaire ou socio-culturelle des élèves qu'elle accueille et répond à leurs besoins particuliers en leur offrant des programmes de soutien appropriés.

Actualisation linguistique en français (ALF). Le programme d'ALF est axé sur l'acquisition de compétences linguistiques en français qui sont indispensables à la poursuite des études et à l'enrichissement du répertoire linguistique de l'élève. Il favorise aussi le développement d'une attitude positive envers l'utilisation du français. Ce programme

s'adresse à l'élève qui parle peu ou ne parle pas le français et qui doit se familiariser avec la langue française, les expressions et le vocabulaire couramment utilisés dans les écoles de langue française et dans l'ensemble du curriculum.

Perfectionnement du français (PDF). Le programme de PDF est axé sur le perfectionnement des compétences en littératie et sur l'initiation à la société canadienne. Ce programme s'adresse à l'élève qui parle français, mais qui a connu une scolarisation très différente de celle que reçoivent les élèves des écoles de langue française de l'Ontario ou qui a subi des interruptions dans sa scolarité. Il favorise l'enrichissement et l'élargissement du répertoire linguistique de l'élève pour lui permettre d'intégrer et de suivre avec plus d'aisance le programme régulier des écoles de langue française de l'Ontario. Le programme permet aussi à l'élève de se familiariser avec les particularités du système d'enseignement de langue française et avec son nouveau milieu socioculturel.

Portée des programmes d'ALF et de PDF. Ces deux programmes assurent une meilleure intégration des élèves à leur nouvel environnement scolaire, culturel et linguistique, tout en les appuyant dans leur cheminement identitaire et leur réussite scolaire. Ces programmes d'appui visent l'intégration la plus rapide possible au programme d'études ordinaire.

Responsabilité de l'enseignante ou l'enseignant. Tout le personnel enseignant doit porter une attention particulière à l'élève inscrit au programme d'ALF ou de PDF. Il faut veiller en particulier à ce que l'élève comprenne et assimile dans chaque matière scolaire la terminologie propre au français, acquière les compétences fondamentales requises et se familiarise avec les référents propres à la francophonie. En consultant le profil de l'élève, en suivant le programme d'ALF ou de PDF et en recourant à la différenciation pédagogique, l'enseignante ou l'enseignant pourra assurer une continuité dans le mode de prestation du programme de l'élève.

On peut consulter *Le curriculum de l'Ontario de la 9^e à la 12^e année – Actualisation linguistique en français et Perfectionnement du français, 1999* sur le site Web du ministère de l'Éducation au www.edu.gov.on.ca.

L'ÉDUCATION ANTIDISCRIMINATOIRE ET LES ÉTUDES INFORMATIQUES

Comme tous les programmes-cadres qui composent le curriculum de l'Ontario, le programme d'études informatiques prépare l'élève à devenir une citoyenne ou un citoyen responsable, comprenant la société complexe dans laquelle elle ou il vit et y participant pleinement. On s'attend donc à ce que l'élève comprenne bien les droits, les privilèges et les responsabilités inhérents à la citoyenneté. On s'attend aussi à ce que, dans ses paroles et dans ses actes, elle ou il fasse preuve de respect, d'ouverture et de compréhension envers les individus, les groupes et les autres cultures. Pour ce faire, l'élève doit comprendre toute l'importance de respecter et de protéger les droits de la personne et de s'opposer à toute forme de discrimination et d'expression de haine. De plus, on amènera l'élève à apprécier l'apport des peuples autochtones dans toute leur diversité à l'identité canadienne ainsi qu'à reconnaître la contribution de personnalités francophones ou francophiles de différentes cultures à l'avancement et à la diffusion de la langue et de la culture d'expression française au Canada et dans le monde.

Les activités d'apprentissage proposées dans le cadre du programme devraient être de nature inclusive, refléter et sensibiliser l'élève aux divers points de vue et expériences. Les habiletés de réflexion et de recherche acquises selon ce programme apprendront à l'élève à reconnaître les partis pris, les stéréotypes et les représentations fondées sur des préjugés et à comprendre comment les relations interpersonnelles sont réellement gérées dans un contexte de mondialisation.

L'éducation inclusive vise à fournir à tous les élèves de la province une chance égale d'atteindre leur plein potentiel en leur permettant d'évoluer dans un environnement sain et sécuritaire. En effet, la classe devrait offrir aux élèves un climat d'apprentissage sécurisant et propice à l'épanouissement et au développement de leurs connaissances et leurs habiletés, y compris leurs habiletés intellectuelles de niveau supérieur. À cet égard, l'enseignante ou l'enseignant joue un rôle primordial en fixant, entre autres, des attentes élevées pour tous ses élèves et en accordant à chacune et chacun une attention particulière.

Cette approche inclusive permet, par le choix d'activités, de sensibiliser les élèves à divers phénomènes et enjeux sociaux, notamment en mettant en lumière des événements survenus dans la communauté et en établissant des liens avec les technologies de l'information et des communications abordées dans les cours d'études informatiques. C'est aussi en proposant aux élèves des activités qui mettent en valeur l'utilité du français et du bilinguisme dans la vie socioéconomique et culturelle que l'enseignante ou l'enseignant contribue à accroître l'intérêt et la motivation des élèves, tout en les préparant à devenir des citoyennes et citoyens responsables.

Le choix des activités proposées aux élèves revêt une importance primordiale dans les cours d'études informatiques. Celles-ci doivent non seulement refléter la diversité ethnoculturelle de la société canadienne, mais aussi présenter un intérêt pour les filles comme pour les garçons. Les garçons s'intéressent davantage aux défis de type compétitif et aux éléments visuels dynamiques, alors que les filles préfèrent les activités coopératives et les projets comportant une dimension sociale faisant intervenir des outils de communication. Plusieurs organisations fondées sur le volontariat proposent des programmes visant à susciter l'intérêt des filles pour l'informatique et le génie, et à développer leur confiance en soi ainsi qu'à favoriser la création d'un milieu d'apprentissage stimulant pour tous.

Les élèves devraient aussi être sensibilisés aux réalités et aux enjeux du monde du travail en informatique. L'enseignante ou l'enseignant pourrait profiter d'un salon de l'informatique pour explorer les nombreuses applications de l'informatique ou inviter une personnalité féminine ou issue d'une minorité visible (p. ex., chef de l'industrie, chercheuse, ancienne élève) pouvant servir de modèle et de source d'inspiration.

L'ÉDUCATION ENVIRONNEMENTALE ET LES ÉTUDES INFORMATIQUES

« L'éducation environnementale est l'éducation concernant l'environnement, pour l'environnement et dans l'environnement qui favorise une compréhension, une expérience riche et pratique et une appréciation des interactions dynamiques entre :

- *les systèmes physiques et biologiques de la Terre;*
- *la dépendance de nos systèmes sociaux et économiques à l'égard de ces systèmes naturels;*
- *les dimensions scientifiques et humaines des enjeux environnementaux;*
- *les conséquences positives et négatives, voulues et involontaires, des interactions entre les systèmes créés par l'homme et les systèmes naturels. »*

« L'ensemble du milieu scolaire a la responsabilité de promouvoir l'éducation environnementale. C'est un champ d'études; on peut donc l'enseigner. C'est une approche à la réflexion critique, au civisme et à la responsabilité personnelle qui peut servir de modèle. C'est un contexte qui peut enrichir et dynamiser l'enseignement dans toutes les matières et qui offre aux élèves la possibilité de mieux se comprendre eux-mêmes et de comprendre leur rôle en société, leur interdépendance mutuelle et les systèmes naturels de la Terre. Les recommandations décrites dans le présent rapport sont jugées comme des composantes essentielles – et interdépendantes – d'une approche intégrée à l'éducation environnementale en Ontario qui permettra aux élèves de devenir des citoyennes et citoyens informés et engagés. »

Préparons nos élèves, Préparons notre avenir :
L'éducation environnementale dans les écoles de l'Ontario, juin 2007 (p. 6 et 10)

L'éducation environnementale se prête bien à l'intégration transdisciplinaire et les études informatiques ne font pas exception. Dans chacun des cours d'études informatiques, des attentes et des contenus d'apprentissage ayant trait à la gestion durable de l'environnement permettent aux élèves de mettre l'accent sur des connaissances liées à la pensée objective, à la citoyenneté et à la responsabilité personnelle. En abordant des questions telles que le recyclage des matériaux et des substances utilisées dans la construction des ordinateurs et de leurs périphériques, les élèves ont plusieurs occasions d'approfondir leurs connaissances au sujet de l'impact et des conséquences que les activités humaines, même les plus simples, peuvent avoir sur l'environnement. On s'attend à ce que les élèves s'impliquent activement dans le développement et la mise en œuvre de stratégies visant à réduire, à réutiliser et à recycler les ordinateurs, les périphériques et les autres appareils. Les élèves tenteront également d'identifier les agences gouvernementales et les partenaires communautaires qui ont élaboré des programmes et des initiatives pour soutenir de telles activités. En identifiant et en mettant en place des mesures visant à minimiser les retombées négatives de l'ordinateur sur l'environnement, les élèves participeront de manière active et responsable à la gestion durable de l'environnement.

Des projets de programmation informatiques peuvent être utilisés pour sensibiliser les élèves au respect de l'environnement. Par exemple, les élèves pourraient concevoir un questionnaire de sondage afin de permettre d'évaluer le taux de connaissance des gens par rapport aux stratégies environnementales pour une utilisation responsable des ordinateurs. Le programme informatique pourrait assigner une note de « conscience environnementale » et, en fonction de celle-ci, suggérer des stratégies responsables additionnelles ou encore fournir un contenu éducatif après chaque réponse. Les élèves

pourraient aussi concevoir un questionnaire de sondage permettant d'évaluer le recours à des stratégies environnementales visant une utilisation responsable des ordinateurs en salle de classe.

Le développement de logiciels peut porter sur des applications telles que la simulation d'écosystèmes sains (p. ex., le fragile équilibre entre la faune et la flore au sein d'un système clos), les conséquences d'une catastrophe environnementale (p. ex., l'impact d'un déversement de mazout sur l'eau, le littoral, le fond marin et les espèces animales et végétales ainsi que sa vitesse d'expansion), ou le coût social et les bénéfices de la conception ou de la rénovation d'édifices en fonction de normes d'efficacité énergétique. La nature dynamique des liens qui unissent l'homme à son environnement fournit un contexte riche et propice au développement d'activités d'apprentissage originales au sein des cours d'études informatiques.

LES HABILITÉS DE LA PENSÉE ET DE LA RECHERCHE

Dans les cours du programme-cadre d'études informatiques, l'élève développe sa capacité à formuler des questions et à planifier les recherches nécessaires pour y répondre. On lui apprend diverses méthodes utiles en recherche et comment choisir celles qui sont adaptées à une recherche particulière. L'élève saura comment tirer des renseignements pertinents de sources imprimées (p. ex., livres, journaux, entrevues, diagrammes, illustrations) et médiatiques (p. ex., Internet, radio, télévision), et dégager des perspectives d'avenir. Avec le temps et l'expérience, l'élève utilisera ces sources d'une manière de plus en plus précise et approfondie, et fera la distinction entre sources primaires et sources secondaires pour déterminer leur validité et leur pertinence et pour en tirer profit de manière adéquate. Ceci s'applique tout particulièrement aux sources électroniques.

LA LITTÉRATIE ET LA NUMÉRATIE

Les compétences liées à la littératie et à la numératie sont essentielles à tous les apprentissages, dans toutes les disciplines. On définit la littératie comme la maîtrise des savoirs qui permettent à l'élève de s'exprimer, d'écrire, de lire, de chercher des informations, d'utiliser les technologies de l'information et des communications et d'exercer une pensée critique à un niveau fonctionnel dans ses apprentissages actuels et futurs. Quant à la numératie, elle comprend l'ensemble des compétences essentielles basées sur des concepts mathématiques et des compétences connexes, qui permettent à l'élève d'utiliser la mesure et les propriétés des nombres et des objets géométriques, de résoudre des problèmes, de développer sa pensée critique, de lire et d'interpréter l'information, y compris dans des graphiques, des tableaux et des diagrammes faisant appel aux concepts mathématiques et de communiquer des données mathématiques.

La littératie et la numératie permettront à l'élève d'apprendre, sa vie durant, dans toutes les disciplines et d'accéder aux niveaux supérieurs de pensée. Il incombe au personnel enseignant de toutes les disciplines de veiller à ce que l'élève progresse dans l'acquisition des compétences liées à la littératie et à la numératie. L'enseignante ou l'enseignant qui remarque que l'élève accuse un retard dans l'acquisition des compétences liées à la littératie et à la numératie devra prendre des dispositions particulières pour l'aider en s'inspirant des initiatives de littératie et de numératie élaborées par son conseil scolaire et son école.

Le ministère de l'Éducation facilite l'élaboration de ressources pour appuyer le développement de compétences liées à la littératie et la numératie dans tout le curriculum. Des stratégies pratiques applicables à tous les cours sont fournies dans les documents suivants :

- *La littératie en tête de la 7^e à la 12^e année : Rapport du Groupe d'experts sur les élèves à risque, 2003*
- *La numératie en tête de la 7^e à la 12^e année : Rapport du Groupe d'experts pour la réussite des élèves, 2004*
- *La littératie en tête : Stratégies pour toutes les matières de la 7^e à la 12^e année, 2005*
- *Moi, lire? Tu blagues! Guide pratique pour aider les garçons en matière de littératie, 2005*

Ces ressources sont affichées sur le site Web du ministère de l'Éducation au www.edu.gov.on.ca.

LA MAJEURE HAUTE SPÉCIALISATION

La Majeure Haute Spécialisation est un programme spécialisé approuvé par le ministère de l'Éducation qui permet aux élèves de personnaliser leur expérience au palier secondaire tout en répondant aux conditions d'obtention du diplôme d'études secondaires de l'Ontario (DESO). Elle prépare également les élèves à faire la transition entre l'école secondaire et la formation en apprentissage, le collège, l'université ou le marché du travail. Le programme de la majeure permet aux élèves d'acquérir des connaissances et des habiletés relevant d'un secteur d'activités spécifique dans des milieux d'apprentissage engageants et liés à leurs objectifs de carrière.

Chaque Majeure Haute Spécialisation doit inclure les cinq (5) composantes suivantes. Celles-ci sont présentées de façon plus détaillée dans la série de guides portant sur chaque secteur :

- un ensemble de 8 à 10 crédits de 11^e et 12^e année qui comprend :
 - quatre crédits de spécialisation axés sur des connaissances et des compétences propres au secteur,
 - de deux à quatre crédits d'appui intégrant des activités d'apprentissage contextualisées reliées au secteur,
 - deux crédits en éducation coopérative;
- des certifications et des formations reconnues par le secteur qui sont énumérées dans chaque guide de secteur;
- des occasions d'apprentissage par l'expérience et d'exploration de carrière;
- des expériences d'anticipation qui reflètent le choix de destination postsecondaire de l'élève;
- le développement de compétences essentielles et d'habitudes de travail propres au secteur et leur documentation à l'aide d'outils du Passeport-compétences de l'Ontario (PCO).

Les cours d'études informatiques s'inscrivent dans l'ensemble des crédits requis en tant que crédits de spécialisation dans les programmes de la Majeure Haute Spécialisation ou dans les programmes conçus pour offrir aux élèves des itinéraires

d'études spécialisés. Ils permettent à l'élève d'acquérir des connaissances et des habiletés qui sont importantes dans des secteurs économiques et qui sont nécessaires pour réussir sur le marché du travail ou pour poursuivre des études postsecondaires. Les cours d'études informatiques peuvent être combinés aux crédits d'éducation coopérative pour fournir à l'élève l'expérience en milieu de travail exigée par des programmes de la majeure et par différents itinéraires d'études spécialisés. Les programmes de la Majeure Haute Spécialisation pourraient fournir des possibilités d'apprentissage dans des secteurs spécifiques, qu'elles soient offertes par des employeuses et employeurs, des centres de formation professionnelle, des collèges ou des organismes communautaires.

LA PLANIFICATION DE CARRIÈRE

Les cours d'études informatiques portent sur des projets authentiques et reflètent les pratiques de l'industrie de l'informatique en constante évolution. Lors de projets réalisés dans ces cours, l'élève développe des connaissances et des habiletés telles que des compétences techniques, la capacité de raisonnement, la capacité à résoudre des problèmes et des habitudes de travail telles que le sens de l'organisation et le travail en équipe qui sont très appréciées dans le monde du travail. Enfin, l'élève explore des itinéraires d'études postsecondaires et de formation professionnelle et examine les débouchés sur le marché de l'emploi.

LE PASSEPORT-COMPÉTENCES DE L'ONTARIO ET LES COMPÉTENCES ESSENTIELLES

Le personnel enseignant qui planifie les cours d'études informatiques doit encourager la connaissance, la compréhension et le développement des compétences essentielles et des habitudes de travail nécessaires pour réussir au travail. Le Passeport-compétences de l'Ontario (PCO) est une ressource Web bilingue qui offre une description claire des compétences essentielles telles que la lecture de textes, la rédaction, l'utilisation de documents, l'informatique, le calcul et la capacité de raisonnement. On se sert des compétences essentielles dans la vie de tous les jours et elles sont transférables de l'école au travail, d'un emploi à l'autre et d'un secteur à l'autre. Le PCO inclut une base de données de tâches en milieu de travail et de descriptions d'habitudes de travail importantes tels la fiabilité, la sécurité au travail et le service à la clientèle. Il offre aussi aux employeuses et employeurs une méthode cohérente pour consigner les stages d'éducation coopérative auxquels a participé l'élève et pour évaluer les compétences et les habitudes de travail qu'elle ou il a démontrées en cours de stage. L'élève peut se servir du PCO pour préciser les compétences et les habitudes de travail déjà acquises, planifier le développement de nouvelles compétences ou montrer aux employeuses et employeurs ce qu'elle ou il a appris et aimerait faire.

Les compétences décrites dans le PCO sont basées sur des compétences essentielles que des agences nationales et internationales et le gouvernement du Canada ont cernées, à la suite de recherches exhaustives, comme étant les compétences requises pour travailler et apprendre tout au long de sa vie. Les compétences essentielles constituent la base de l'apprentissage de toute autre habileté et permettent aux personnes de progresser dans leur emploi et de s'adapter au changement en milieu de travail. Pour des précisions sur le PCO et les compétences essentielles, consulter le site Web au <http://skills.edu.gov.on.ca>.

L'ÉDUCATION COOPÉRATIVE ET LES AUTRES EXPÉRIENCES EN MILIEU DE TRAVAIL

L'éducation coopérative et les autres formes d'apprentissage par l'expérience permettent à l'élève d'appliquer les habiletés acquises en salle de classe dans des contextes authentiques au sein de la communauté du monde des affaires et des services publics. L'éducation coopérative et autres expériences en milieu de travail aident l'élève à approfondir sa connaissance des possibilités d'emploi dans de nombreux domaines, notamment en développement d'applications Web, en administration de bases de données et en génie logiciel. De plus, l'élève élargit sa compréhension des pratiques du monde du travail, des certifications et de la nature des relations employeurs-employés. En outre, en se basant sur ses expériences, elle ou il reconnaît l'apport important de la connaissance des deux langues officielles du Canada. Il s'avère important que les enseignantes et enseignants des cours d'études informatiques entretiennent des liens avec les entreprises locales, notamment celles de la communauté francophone, afin d'assurer à l'élève des expériences pratiques qui viendront renforcer les connaissances et les habiletés acquises à l'école.

La préparation aux expériences en milieu de travail doit comprendre un enseignement sur les mesures liées à la santé et la sécurité en milieu de travail. L'enseignante ou l'enseignant appuyant l'élève en situation d'apprentissage en milieu de travail doit évaluer les conditions relatives à la santé et la sécurité dans le milieu de travail. Avant de participer à une expérience en milieu de travail, l'élève doit acquérir les connaissances et les compétences nécessaires pour assurer sa sécurité physique et son bien-être personnel ainsi que celui de ses collègues de travail. L'élève doit comprendre l'importance de la confidentialité et du respect de la vie privée, tel qu'il est énoncé dans la *Loi sur l'accès à l'information et la protection de la vie privée* (1990), ainsi que les principaux points des lois qui régissent les logiciels et les systèmes informatiques telle la *Loi sur le droit d'auteur* (1985). Elle ou il a le droit de travailler dans un milieu exempt de mauvais traitements et de harcèlement. L'élève doit être renseigné quant aux ressources scolaires et communautaires, aux politiques de l'école et à la marche à suivre pour signaler toutes formes d'abus et de harcèlement.

La note Politique/Programmes n° 76A intitulée *Assurance contre les accidents du travail pour les élèves des programmes de formation pratique* (septembre 2000) trace les grandes lignes des procédures à suivre pour assurer le respect des dispositions de la *Loi sur la sécurité professionnelle et l'assurance contre les accidents du travail* (1997) aux élèves âgés d'au moins 14 ans inscrits à un stage de plus d'une journée en milieu de travail. L'observation au poste de travail et le jumelage sont considérés comme une sortie éducative. Le personnel enseignant doit connaître l'âge minimum requis selon la *Loi sur la santé et la sécurité au travail* (1990) pour trouver un milieu de travail où l'élève peut travailler. Tous les stages d'éducation coopérative et les autres expériences en milieu de travail sont dispensés selon les prescriptions énoncées dans *Éducation coopérative et autres formes d'apprentissage par l'expérience : Lignes directrices pour les écoles secondaires de l'Ontario, 2000*.

LA SANTÉ ET LA SÉCURITÉ

La santé et la sécurité sont d'une importance vitale dans tout le programme d'études informatiques. Dans chacun des cours, l'élève sera amené à développer des attitudes et des habitudes responsables et sécuritaires à l'école, à la maison et au travail. Avant de permettre à l'élève d'utiliser l'équipement, en atelier ou au travail, on s'assurera qu'elle ou il possède les compétences et les attitudes indispensables au maintien de sa santé et de sa sécurité.

L'utilisation prolongée d'un ordinateur peut occasionner, entre autres, de la fatigue oculaire et des blessures musculosquelettiques. Le personnel enseignant encouragera les élèves à comprendre ces risques et à prendre toutes les dispositions raisonnables pour les contrôler.

Les questions relatives à la santé et à la sécurité peuvent s'avérer importantes lorsque l'apprentissage fait appel à des activités pratiques, en particulier celles qui se déroulent à l'extérieur de l'école. Ces activités offrent une dimension authentique et motivante en ce qui a trait aux expériences d'apprentissage de l'élève. L'enseignante ou l'enseignant planifiera avec soin ces activités afin de prévoir les problèmes et de prévenir les risques pour la santé et la sécurité de l'élève.

COURS



Introduction à l'informatique, 10^e année

cours ouvert

ICS20

Ce cours présente à l'élève les concepts fondamentaux de l'informatique et les techniques élémentaires de programmation qui lui permettent de développer des programmes répondant aux besoins des utilisatrices et utilisateurs. L'élève étudie le fonctionnement du matériel informatique, des logiciels, des systèmes d'exploitation, des réseaux locaux et de la maintenance informatique. De plus, ce cours amène l'élève à explorer les enjeux sociétaux de l'industrie de l'informatique ainsi que des possibilités de carrière et de formation dans ce secteur.

Préalable : Aucun

A. ENVIRONNEMENT INFORMATIQUE DE TRAVAIL

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- A1.** décrire le fonctionnement du matériel informatique constituant un ordinateur personnel en utilisant les termes justes en français.
- A2.** comparer différents types de logiciels et leurs fonctions respectives.
- A3.** utiliser les fonctions de base d'un système d'exploitation.
- A4.** expliquer divers moyens d'utiliser des réseaux locaux pour développer des programmes informatiques.
- A5.** expliquer le rôle de la maintenance informatique.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Matériel informatique

- A1.1** décrire les composants matériels internes d'un ordinateur personnel (*p. ex., microprocesseur, mémoire vive, disque dur, bloc d'alimentation électrique*), leurs fonctions (*p. ex., exécuter des instructions, stocker des informations de façon permanente, alimenter les composants matériels en électricité*) et leurs caractéristiques (*p. ex., fréquence du microprocesseur [en gigahertz], taille de la mémoire vive [en gigaoctets], consommation électrique [en watts]*).
- A1.2** définir les fonctions des périphériques externes qui sont couramment utilisés (*p. ex., imprimante, moniteur, clavier, souris, clé de mémoire, lecteur de cédéroms*).
- A1.3** comparer divers composants matériels selon leurs performances respectives (*p. ex., capacité de la mémoire interne, définition graphique, consommation énergétique*) et en fonction des besoins des utilisatrices et utilisateurs (*p. ex., accessibilité améliorée, efficacité au bureau, divertissement*).
- A1.4** décrire les principales interfaces matérielles d'un ordinateur (*p. ex., bus interne, bus série universel, interface réseau*), leurs fonctions (*p. ex., communication entre le microprocesseur et la mémoire vive, conversion des signaux numériques en signaux analogiques, accès à Internet*) et leurs caractéristiques (*p. ex., nombre de bits, fréquence, débit de transmission [en gigabit par seconde]*).

Logiciels

- A2.1** distinguer les logiciels liés au système d'exploitation (*p. ex., logiciel utilitaire, pilote de périphérique, planificateur de tâches, éditeur de registre*) des logiciels d'application (*p. ex., environnement de développement intégré [EDI], logiciel de traitement de texte, logiciel de gestion de courriel, logiciel de dessin, lecteur multimédia*).
- A2.2** comparer différents systèmes d'exploitation pour ordinateurs personnels selon différents critères (*p. ex., fonction, performance, utilisation*).
- A2.3** comparer divers logiciels selon leur type (*p. ex., suite bureautique, jeu, compilateur, synthétiseur de la parole à partir du texte*) et en fonction des besoins des utilisatrices et utilisateurs (*p. ex., outil de bureautique, divertissement, accessibilité, développement de service en ligne*).

Système d'exploitation

- A3.1** décrire les fonctions d'un système d'exploitation (*p. ex., exécution de logiciels, archivage de fichiers, contrôle d'accès*).
- A3.2** expliquer l'utilité des raccourcis-clavier (*p. ex., copier un fragment de texte, ouvrir un fichier, terminer l'exécution d'un logiciel*) lors du développement d'un programme.

A3.3 organiser l'information en utilisant des techniques de gestion de fichiers (*p. ex., créer un répertoire, copier un raccourci, déplacer un fichier*).

Réseaux locaux

A4.1 reconnaître diverses applications de la réseautique (*p. ex., téléphonie sur IP, Web, courriel, messagerie instantanée*).

A4.2 décrire le matériel essentiel (*p. ex., carte réseau, routeur, modem*) et ses caractéristiques (*p. ex., connexion par câble ou sans fil, connexion entre machines et entre réseaux*) pour bâtir des réseaux locaux.

A4.3 distinguer les services d'accès (*p. ex., Internet, réseaux de téléphonie mobile*) offerts par différents fournisseurs.

A4.4 expliquer les éléments de configuration (*p. ex., adresse URL, permission*) permettant de partager des ressources informatiques en réseau (*p. ex., dossier, imprimante, connexion Internet*) et de développer des programmes informatiques.

Maintenance informatique

A5.1 expliquer les enjeux (*p. ex., sécurité, compatibilité avec de nouveaux périphériques, maintien d'un bon niveau de performance*) de la mise à jour des logiciels (*p. ex., antivirus, système d'exploitation, logiciel d'application*) et du matériel informatique (*p. ex., routeur, mémoire vive, carte vidéo*).

A5.2 décrire les effets des programmes malveillants (*p. ex., virus, cheval de Troie, logiciel espion*) et des contre-mesures de sécurité informatique connues (*p. ex., antivirus, coupe-feu, blocage de fenêtres, filtre antipourriel*).

A5.3 décrire des méthodes de protection contre l'utilisation illicite du matériel informatique (*p. ex., choix d'un mot de passe efficace, lecteur biométrique, logiciel de chiffrement PGP*).

A5.4 expliquer diverses pratiques de maintenance préventive (*p. ex., classification et archivage de fichiers, vérification du système de fichiers, optimisation du système*).

B. INTRODUCTION À LA PROGRAMMATION

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- B1.** appliquer des techniques élémentaires de programmation.
- B2.** développer des programmes.
- B3.** appliquer des techniques élémentaires d'assurance-qualité logicielle.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Techniques de programmation

- B1.1** utiliser un modèle de résolution de problèmes (*p. ex., entrée-traitement-sortie, diagramme, scénario*) afin de produire un code source (*p. ex., appliquelette, script, macro*).
- B1.2** décrire les types de données primitifs définis par un langage de programmation donné (*p. ex., nombre entier, caractère alphanumérique*) ainsi que certains types de données complexes (*p. ex., chaîne de caractères*).
- B1.3** définir les constantes et les variables appropriées pour résoudre un problème de programmation (*p. ex., programme qui calcule la circonférence d'un cercle : la constante π et la variable r décrivant le rayon*).
- B1.4** définir les expressions et les instructions d'un programme ainsi que leur ordre d'exécution en tenant compte des priorités des opérateurs (*p. ex., opérateur arithmétique, opérateur d'affectation, opérateur de comparaison*).
- B1.5** identifier des situations mettant en jeu des structures logiques de décision et de répétition (*p. ex., choix de fonction dans un menu, différence d'affichage d'une cellule de tableur selon sa valeur, tâche répétitive, manipulation d'une liste d'objets*).
- B1.6** rédiger des expressions variées utilisant des opérateurs (*p. ex., opérateur booléen, opérateur de comparaison, opérateur arithmétique*).
- B1.7** comparer des manières de résoudre un même problème de programmation en utilisant des environnements différents (*p. ex., tableur, application Web, programme créé sur demande*).

Développement de programmes

- B2.1** rédiger des énoncés d'entrée et de sortie se conformant au cahier des charges d'un programme.
- B2.2** concevoir des programmes utilisant des variables et des expressions pour manipuler des nombres et des chaînes de caractères (*p. ex., calcul de moyenne, quiz, jeu de conversation*).
- B2.3** développer des programmes utilisant une structure de décision incluant au moins deux choix possibles pour répondre à un problème donné (*p. ex., jeu de devinettes, roche-papier-ciseaux, question à choix multiples*).
- B2.4** développer des programmes (*p. ex., animation pour un site Web, jeu de dés, application scientifique*) utilisant une ou plusieurs boucles avec ou sans compteur.

Assurance-qualité logicielle

- B3.1** utiliser une nomenclature (*p. ex., préfixes dépendant du type de variable, majuscules et minuscules, nom significatif*) pour définir les variables, les fonctions, les fichiers et les répertoires utilisés.
- B3.2** rédiger une documentation interne (*p. ex., ligne de commentaire, infobulle*) pour décrire un programme en respectant des normes visant à en faciliter la maintenance informatique.
- B3.3** utiliser des stratégies de débogage adaptées aux différents types d'erreurs – erreur de syntaxe, erreur de logique et erreur d'exécution.

B3.4 utiliser la technique du traçage pour repérer et corriger les erreurs de logique et les erreurs d'exécution d'un programme.

B3.5 tester un programme donné à l'aide de données pertinentes (*p. ex., données de l'énoncé du problème, cas extrêmes*).

C. ENJEUX SOCIÉTAUX ET PERSPECTIVES PROFESSIONNELLES

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- C1.** décrire des procédures informatiques visant à protéger l'environnement et la santé publique.
- C2.** expliquer les enjeux sociétaux de l'informatique.
- C3.** déterminer la valeur d'une formation technologique sur le plan personnel ainsi que des possibilités de carrière et de formation, notamment en informatique.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Environnement et santé

- C1.1** expliquer les répercussions positives et négatives de l'usage du matériel informatique sur l'environnement et la santé publique (p. ex., *régulation de la consommation énergétique, gaspillage du papier, séquençage du génome humain, accumulation de déchets toxiques dans les sites d'enfouissement, risque de blessure musculosquelettique*).
- C1.2** identifier des mesures visant à réduire les risques que l'utilisation de matériel informatique pose pour l'environnement et la santé (p. ex., *programme du bureau sans papier, usage de logiciels plus performants pour allonger la durée de vie du matériel informatique, adaptation ergonomique*).
- C1.3** décrire des procédures environnementales de gestion des déchets informatiques (p. ex., *recyclage du papier et de l'encre en poudre, réutilisation des manuels d'installation et d'entretien, transformation du matériel informatique*).

Enjeux sociétaux

- C2.1** décrire l'impact du matériel informatique (p. ex., *ordinateur de poche, calculatrice graphique programmable, lecteur multimédia, système mondial de positionnement [GPS]*) et des technologies des communications (p. ex., *blogue, balado, informatique mobile*) sur le mode de vie, de travail et de communication des gens.

- C2.2** donner des exemples de logiciel ou de matériel informatique (p. ex., *loupe, reconnaissance vocale, clavier ergonomique*) qui rendent l'ordinateur et les services informatisés plus accessibles et plus faciles à utiliser.

- C2.3** analyser les enjeux sécuritaires de l'accès en ligne (p. ex., *cyberintimidation, vol d'identité, pérennité de l'information diffusée sur Internet*).

- C2.4** expliquer les enjeux éthiques et légaux liés à l'informatique (p. ex., *téléchargement de fichiers multimédias, utilisation de ressources en ligne, adoption de licences libres*).

Carrière et formation

- C3.1** préciser l'utilité d'avoir une connaissance pratique de l'informatique dans la vie quotidienne (p. ex., *installation de logiciel, réparation d'ordinateur, montage d'album de photos*).
- C3.2** explorer diverses possibilités de carrière et de formation en informatique (p. ex., *programmation, génie électrique, bio-informatique*), en consultant diverses sources de renseignements (p. ex., *journaux, Classification nationale des professions [CNP], site des associations professionnelles, annuaire des établissements de formation*).
- C3.3** identifier des occasions de bénévolat, de travail à temps partiel et d'activités parascolaires en technologie, notamment en informatique, et les conditions à remplir pour s'en prévaloir.

- C3.4** examiner les stéréotypes (*p. ex., l'informatique est un milieu pour les hommes seulement*) et les préjugés (*p. ex., à compétence égale, salaires inégaux entre les femmes et les hommes*) qui sont véhiculés au sujet de certains métiers, notamment en informatique.
- C3.5** identifier les compétences essentielles (*p. ex., communication verbale, prise de décisions, résolution de problèmes*) ainsi que les habitudes de travail (*p. ex., habitudes associées à la sécurité au travail, à la capacité de travailler en équipe et de façon autonome, à la fiabilité, au service à la clientèle*) répertoriées dans le Passeport-compétences de l'Ontario (PCO) qui sont indispensables pour réussir dans l'industrie de l'informatique.
- C3.6** expliquer les avantages de se constituer un portfolio et de le garder à jour (*p. ex., gérer son apprentissage, planifier son cheminement de carrière*).

Introduction au génie informatique, 11^e année

cours préuniversitaire

ICS3U

Ce cours initie l'élève aux concepts fondamentaux de l'informatique et aux techniques de développement de logiciels. Dans le cadre de divers projets illustrant le cycle de vie d'un logiciel, l'élève développe des habiletés et une compréhension solide d'un langage de programmation en se familiarisant avec les outils et les techniques de développement de logiciels, notamment la résolution de problèmes, la conception d'algorithmes et l'assurance-qualité. Le cours amène également l'élève à explorer l'impact de l'industrie de l'informatique sur l'environnement et la santé publique ainsi que les possibilités de carrière et de formation professionnelle dans ce secteur.

Préalable : Aucun

A. ENVIRONNEMENT INFORMATIQUE DE TRAVAIL

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- A1.** expliquer le fonctionnement d'un ordinateur personnel en utilisant les termes justes en français.
- A2.** appliquer des techniques de gestion de fichiers.
- A3.** utiliser des outils appropriés pour développer des programmes.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Fonctionnement d'un ordinateur personnel

- A1.1** expliquer les fonctions des composants matériels internes d'un ordinateur personnel (p. ex., carte mère, microprocesseur, mémoire vive, carte vidéo, carte son).
- A1.2** expliquer les fonctions des périphériques externes qui sont couramment utilisés (p. ex., souris, clavier, moniteur, imprimante, appareil photo numérique, caméra, clé de mémoire).
- A1.3** comparer les performances du matériel informatique de divers ordinateurs personnels en utilisant des mesures objectives (p. ex., fréquence du microprocesseur [en gigahertz], capacité de la mémoire vive [en gigaoctets], consommation électrique [en watts]).
- A1.4** expliquer la relation fonctionnelle entre un langage de programmation et les composants essentiels d'un ordinateur (p. ex., enregistrement et transfert de données, opération arithmétique, calcul géométrique).

Gestion de fichiers

- A2.1** répertorier les fichiers d'un programme en utilisant les fonctions du système d'exploitation.
- A2.2** appliquer une procédure systématique de sauvegarde des fichiers (p. ex., archivage programmé de fichiers, copie manuelle sur une clé de mémoire ou en ligne, utilisation d'un système de contrôle de versions).

- A2.3** décrire plusieurs types de virus (p. ex., attaque du système d'amorçage, macro dans un document de bureautique) et autres programmes malveillants (p. ex., cheval de Troie, logiciel espion), leur cycle de vie ainsi que les contre-mesures de sécurité informatique connues (p. ex., antivirus, coupe-feu, blocage des fenêtres, filtre antipourriel).
- A2.4** utiliser les services du réseau local (p. ex., partage de fichier et d'imprimante, centralisation du code et de l'échéancier) pour faciliter la gestion et la sauvegarde des fichiers lors du développement d'un programme.

Outils de développement

- A3.1** comparer les principales fonctions des systèmes d'exploitation (p. ex., exécution de programmes, archivage de fichiers, contrôle d'accès) et des logiciels d'application (p. ex., traitement de texte, lecture de courriels), notamment des outils de développement (p. ex., compilateur, interpréteur).
- A3.2** expliquer les caractéristiques et les avantages de divers environnements de développement (p. ex., environnement de développement intégré [EDI]; éditeur de texte spécialisé et interpréteur) pour développer des programmes.
- A3.3** utiliser les fonctions respectives d'un compilateur ou d'un interpréteur.
- A3.4** utiliser l'aide disponible (p. ex., fonction d'aide de l'EDI, tutoriel, aide en ligne) pour développer des programmes.

B. TECHNIQUES DE PROGRAMMATION

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- B1.** appliquer les principales règles de syntaxe et de sémantique d'un langage de programmation.
- B2.** expliquer des algorithmes et des structures de données élémentaires.
- B3.** appliquer des techniques d'assurance-qualité logicielle.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Syntaxe et sémantique

- B1.1** décrire les types de données primitifs définis par un langage de programmation donné (*p. ex., nombre entier, nombre décimal, caractère alphanumérique*).
- B1.2** décrire la représentation interne (*p. ex., code binaire, code ASCII, Unicode*) de divers types de données (*p. ex., nombre entier, caractère, chaîne de caractères*).
- B1.3** définir les termes suivants : valeur littérale, constante et variable.
- B1.4** définir les concepts de portée et de durée de vie des variables.
- B1.5** décrire différents types de fonctions (*p. ex., fonction avec ou sans paramètres, avec ou sans valeur de retour [procédure]; fonction récursive*).
- B1.6** appliquer les règles de syntaxe fondamentales (*p. ex., déclaration de variables, usage des opérateurs, appel de fonction*) d'un langage de programmation.
- B1.7** rédiger des expressions utilisant des opérateurs (*p. ex., opérateur booléen, opérateur de comparaison, opérateur arithmétique*).

Algorithmes et structures de données

- B2.1** définir les caractéristiques d'un tableau à une dimension telles que les éléments, l'indice et la taille.
- B2.2** expliquer des algorithmes qui permettent de lire ou de modifier les éléments d'un tableau à une dimension.

B2.3 expliquer le fonctionnement des structures de contrôle – séquentielle, décisionnelle et itérative – d'un programme.

B2.4 expliquer des algorithmes qui permettent de traiter l'information entrée par une utilisatrice ou un utilisateur et d'afficher le résultat à l'écran.

Assurance-qualité logicielle

B3.1 expliquer la différence entre les erreurs de logique, les erreurs de syntaxe et les erreurs d'exécution d'un programme.

B3.2 interpréter les messages d'erreur émis par les environnements de développement et d'exécution (*p. ex., compilateur, outil de débogage, machine virtuelle*).

B3.3 corriger des erreurs de logique, des erreurs de syntaxe et des erreurs d'exécution d'un programme.

B3.4 déterminer un ensemble de valeurs appropriées pour tester l'exactitude d'un programme.

B3.5 déboguer des programmes à l'aide de différentes méthodes (*p. ex., traçage, exécution pas à pas, utilisation de points d'interruption, observation des variables, insertion d'instructions dans le programme*).

B3.6 appliquer des règles de mise en page (*p. ex., tabulation*), de rédaction (*p. ex., commentaire*) et de nomenclature (*p. ex., nom de variable*) au programme afin d'en assurer sa maintenance et sa documentation.

C. DÉVELOPPEMENT DE LOGICIELS

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- C1.** appliquer des techniques de développement de logiciels.
- C2.** concevoir des algorithmes répondant aux problèmes donnés.
- C3.** développer des programmes répondant aux problèmes donnés.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Techniques de développement de logiciels

- C1.1** décrire les problèmes en termes de données d'entrée, de traitement de données et de données de sortie.
- C1.2** utiliser différentes approches pour résoudre des problèmes (*p. ex., méthode de la dichotomie, méthode empirique, représentation graphique*).
- C1.3** décrire les activités principales du développement de logiciels (*p. ex., définition des besoins, conception, codage, vérification et validation*) et leurs produits livrables (*p. ex., cahier des charges, code source, résultats des tests*).
- C1.4** interpréter des échéanciers (*p. ex., calendrier des tâches, diagramme de Gantt, graphique PERT*).
- C1.5** appliquer un plan de test (*p. ex., identification des scénarios possibles, entrée des données, prédiction des résultats, comparaison avec les données de sortie*).
- C1.6** présenter à l'oral comme à l'écrit les produits livrables (*p. ex., cahier des charges, algorithme, résultat des tests*) en utilisant les termes justes en français.

Conception d'algorithmes

- C2.1** concevoir des algorithmes qui répondent à des problèmes mathématiques donnés (*p. ex., calcul de la longueur du troisième côté d'un triangle rectangle lorsque les deux premiers côtés sont connus, affichage des nombres premiers inférieurs à 100*).

C2.2 concevoir des algorithmes de traitement des données (*p. ex., validation des données entrées par l'utilisatrice ou l'utilisateur, manipulation des éléments d'un tableau*).

C2.3 concevoir des imbrications de structures de contrôle (*p. ex., structures décisionnelles imbriquées, utilisation d'une valeur sentinelle dans une structure itérative pour effectuer une sortie exceptionnelle*).

C2.4 concevoir des algorithmes traitant des exceptions (*p. ex., division par zéro, racine carrée d'un nombre négatif, 0⁰ [zéro exposant zéro]*).

C2.5 comparer qualitativement et quantitativement les performances de deux algorithmes qui répondent au même problème (*p. ex., type de données, nombre de lignes, usage de la mémoire*).

Développement de programmes

C3.1 concevoir des fonctions répondant à des besoins donnés (*p. ex., calcul de formule, encapsulation d'un bloc de code, répétition d'un bloc d'instructions*).

C3.2 modéliser un problème de programmation en utilisant diverses techniques (*p. ex., organigramme, langage de modélisation UML, diagramme de flux, pseudocode*).

C3.3 appliquer le principe de modularisation au développement de programmes (*p. ex., en créant des fonctions pour éviter la duplication de blocs de code, en créant des classes, en séparant au besoin le code en plusieurs fichiers*).

C3.4 concevoir des interfaces utilisateurs conviviales (*p. ex., interface d'une console, interface d'une page Web, interface graphique*) qui répondent aux besoins des utilisatrices et utilisateurs.

C3.5 développer des programmes répondant à des problèmes donnés en utilisant des modules préalablement conçus à cet effet (*p. ex., fonction, interface utilisateur, documentation*) et des fonctions prédéfinies (*p. ex., générateur de nombres aléatoires, gestionnaire d'événements, convertisseur de chaînes de caractères*).

D. ENJEUX SOCIÉTAUX ET PERSPECTIVES PROFESSIONNELLES

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- D1.** analyser des mesures favorables pour l'environnement et la santé publique concernant l'utilisation du matériel informatique.
- D2.** analyser diverses possibilités de carrière et de formation professionnelle en informatique.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Environnement et santé

- D1.1** analyser l'impact de l'industrie de l'informatique sur l'environnement et la santé publique en identifiant les facteurs bénéfiques (*p. ex., modélisation des forces environnementales et de leur impact au niveau planétaire, séquençage du génome humain, repliement des protéines*) et les facteurs nuisibles (*p. ex., volume et toxicité des déchets, consommation électrique, utilisation du papier*).
- D1.2** évaluer l'impact des mesures prises par les organismes publics (*p. ex., création de normes environnementales, mesure fiscale, subvention*) et privés (*p. ex., initiative des fabricants de matériel informatique, action d'une association proenvironnementale, initiative individuelle*) sur l'empreinte environnementale de l'informatique.
- D1.3** déterminer des stratégies pour réduire la consommation des systèmes informatiques, et pour réutiliser et recycler du matériel informatique (*p. ex., éteindre le moniteur en fin de journée, réaffecter des ordinateurs obsolètes à des tâches plus appropriées, recycler les cartouches d'encre*).
- D1.4** décrire des moyens pour prévenir (*p. ex., poste de travail ergonomique, exercice musculaire*) des problèmes de santé liés à l'usage des ordinateurs (*p. ex., trouble musculosquelettique, trouble de la vue*).

Carrière et formation

- D2.1** comparer des choix de carrière possibles en informatique (*p. ex., programmation, architecture d'entreprise, recherche*), notamment en ce qui a trait aux tâches à accomplir, aux conditions salariales et aux possibilités d'avancement.
- D2.2** répertorier les programmes de formation offerts en informatique en identifiant les préalables et les options éducatives qui existent en français.
- D2.3** identifier des occasions et des moyens d'obtenir de l'expérience en informatique (*p. ex., bénévolat, éducation coopérative, salon de l'emploi, mentorat*).
- D2.4** décrire des services de soutien qui favorisent l'orientation vers des métiers non traditionnels en informatique (*p. ex., programme de mentorat, groupe de soutien virtuel, programme postsecondaire spécialisé, association syndicale*).
- D2.5** décrire les compétences essentielles (*p. ex., communication verbale, prise de décisions, résolution de problèmes*) ainsi que les habitudes de travail (*p. ex., habitudes associées à la sécurité au travail, à la capacité de travailler en équipe et de façon autonome, à la fiabilité, au service à la clientèle*) du Passeport-compétences de l'Ontario (PCO) qui sont indispensables pour réussir dans l'industrie de l'informatique.
- D2.6** créer un portfolio en sélectionnant les travaux ou les réalisations (*p. ex., logiciel, code source, diagramme*) qui témoignent de ses habiletés, de ses compétences, de ses expériences et des certifications acquises en informatique.

Introduction à la programmation informatique, 1¹^e année

cours précollégial

ICS3C

Ce cours initie l'élève aux techniques de programmation qui lui permettent de développer des programmes répondant aux besoins des utilisatrices et utilisateurs, notamment en ce qui a trait à l'interface utilisateur et à la présentation des données. Dans le cadre de divers projets illustrant le cycle de vie d'un logiciel, l'élève se familiarise avec l'environnement informatique de travail, et développe et débogue des programmes. Le cours amène aussi l'élève à explorer l'impact de l'industrie de l'informatique sur l'environnement et la santé publique ainsi que les diverses possibilités de carrière et de formation dans ce secteur.

Préalable : Aucun

A. ENVIRONNEMENT INFORMATIQUE DE TRAVAIL

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- A1.** expliquer le fonctionnement d'un ordinateur personnel en utilisant les termes justes en français.
- A2.** appliquer des techniques de gestion de fichiers.
- A3.** utiliser des outils appropriés pour développer des programmes.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Fonctionnement d'un ordinateur personnel

- A1.1** expliquer les fonctions des composants matériels internes d'un ordinateur personnel (p. ex., carte mère, microprocesseur, mémoire vive, carte vidéo, carte son).
- A1.2** expliquer les fonctions des périphériques externes qui sont couramment utilisés (p. ex., souris, clavier, moniteur, imprimante, appareil photo numérique, caméra, clé de mémoire).
- A1.3** comparer les performances du matériel informatique de divers ordinateurs personnels en utilisant des mesures objectives (p. ex., fréquence du microprocesseur [en gigahertz], capacité de la mémoire vive [en gigaoctets], consommation électrique [en watts]).
- A1.4** comparer les caractéristiques majeures des différents types d'accès à Internet (p. ex., câble de radiotélévision, ligne d'abonné numérique, ligne commutée, accès sans fil).

Gestion de fichiers

- A2.1** répertorier les fichiers d'un programme en utilisant les fonctions du système d'exploitation.
- A2.2** appliquer une procédure systématique de sauvegarde des fichiers (p. ex., archivage programmé de fichiers, copie manuelle sur une clé de mémoire ou en ligne, utilisation d'un système de contrôle de versions).

- A2.3** décrire plusieurs types de virus (p. ex., attaque du système d'amorçage, macro dans un document de bureautique) et autres programmes malveillants (p. ex., cheval de Troie, logiciel espion), leur cycle de vie ainsi que les contre-mesures de sécurité informatique connues (p. ex., antivirus, coupe-feu, blocage de fenêtres, filtre antipourriel).
- A2.4** expliquer des raisons motivant l'adoption d'une politique d'utilisation des ordinateurs et du réseau local (p. ex., respect de la vie privée, équité, maintenance informatique).
- A2.5** utiliser les services du réseau local (p. ex., partage de fichier et d'imprimante, centralisation du code et de l'échéancier) pour faciliter la gestion et la sauvegarde des fichiers lors du développement d'un programme.

Outils de développement

- A3.1** expliquer les fonctions respectives des systèmes d'exploitation (p. ex., exécution de programmes, archivage de fichiers, contrôle d'accès) et des logiciels d'application (p. ex., traitement de texte, lecture de courriels), notamment des outils de développement (p. ex., compilateur, interpréteur).
- A3.2** expliquer les caractéristiques et les avantages de divers environnements de développement intégrés (EDI) pour développer des programmes.
- A3.3** utiliser les commandes d'un EDI requises pour éditer, exécuter et déboguer des programmes.
- A3.4** utiliser l'aide disponible (p. ex., fonction d'aide de l'EDI, tutoriel, aide en ligne) pour développer des programmes.

B. TECHNIQUES DE PROGRAMMATION

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- B1.** appliquer des techniques de traitement des données utilisant des variables et des expressions.
- B2.** développer des programmes qui utilisent des structures de contrôle.
- B3.** appliquer des techniques d'assurance-qualité logicielle.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Traitement des données

- B1.1** décrire les types de données primitifs définis par un langage de programmation donné (*p. ex., nombre entier, nombre décimal, caractère alphanumérique*).
- B1.2** expliquer les règles (*p. ex., portée, durée de vie*) qui s'appliquent à la gestion de la mémoire (*p. ex., variable locale ou globale, paramètre d'une fonction*).
- B1.3** utiliser différents types de données (*p. ex., nombre entier, valeur booléenne, chaîne de caractères*) dans un programme.
- B1.4** manipuler des données dans un programme en utilisant des constantes et des variables.
- B1.5** appliquer des techniques de manipulation des chaînes de caractères (*p. ex., échange de caractères, dénombrement*).
- B1.6** rédiger des expressions utilisant des opérateurs (*p. ex., opérateur booléen, opérateur de comparaison, opérateur arithmétique*).

Structures de contrôle

- B2.1** développer un programme qui traite l'information entrée par l'utilisatrice ou l'utilisateur pour répondre à un problème donné (*p. ex., déterminer si une année donnée est bissextile*) en utilisant une ou plusieurs structures de contrôle.
- B2.2** utiliser les structures de contrôle – séquentielle, alternative et itérative – dans un programme pour résoudre des problèmes (*p. ex., minuterie, détermination de la valeur minimale*).

- B2.3** intégrer une imbrication de structures de contrôle (*p. ex., une structure itérative contenant une structure décisionnelle*) dans un programme.

Assurance-qualité logicielle

- B3.1** expliquer la différence entre les erreurs de logique, les erreurs de syntaxe et les erreurs d'exécution d'un programme.
- B3.2** interpréter les messages d'erreur émis par les environnements de développement et d'exécution (*p. ex., compilateur, outil de débogage, machine virtuelle*).
- B3.3** corriger des erreurs de logique, des erreurs de syntaxe et des erreurs d'exécution d'un programme.
- B3.4** déterminer un ensemble de valeurs appropriées pour tester l'exactitude d'un programme.
- B3.5** déboguer des programmes à l'aide de différentes méthodes (*p. ex., traçage, exécution pas à pas, utilisation de points d'interruption, observation des variables, insertion d'instructions dans le programme*).
- B3.6** appliquer des règles de mise en page (*p. ex., tabulation*), de rédaction (*p. ex., commentaire*) et de nomenclature (*p. ex., nom de variable*) au programme afin d'en assurer sa maintenance et sa documentation.

C. DÉVELOPPEMENT DE LOGICIELS

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- C1.** appliquer des techniques de développement de logiciels.
- C2.** concevoir des algorithmes répondant aux problèmes donnés.
- C3.** développer des programmes répondant aux problèmes donnés.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Techniques de développement de logiciels

- C1.1** décrire des problèmes en termes de données d'entrée, de traitement de données et de données de sortie.
- C1.2** utiliser différentes approches pour résoudre des problèmes (*p. ex., méthode de la dichotomie, méthode empirique, représentation graphique*).
- C1.3** décrire les activités principales du développement de logiciels (*p. ex., définition des besoins, conception, codage, vérification et validation*) et leurs produits livrables (*p. ex., cahier des charges, code source, résultats des tests*).
- C1.4** interpréter des échéanciers (*p. ex., calendrier des tâches, diagramme de Gantt, graphique PERT*).
- C1.5** appliquer un plan de test (*p. ex., identification des scénarios possibles, entrée des données, prédiction des résultats, comparaison avec les données de sortie*).
- C1.6** présenter à l'oral comme à l'écrit les produits livrables (*p. ex., cahier des charges, algorithme, résultat des tests*) en utilisant les termes justes en français.

Conception d'algorithmes

- C2.1** concevoir des algorithmes employant des structures de contrôle (*p. ex., utilisation d'un compteur ou d'une valeur sentinelle dans une structure itérative*) pour répondre à des problèmes donnés (*p. ex., entrée de données, contrainte de temps dans un jeu, interaction avec l'utilisatrice ou l'utilisateur*).

- C2.2** concevoir des algorithmes répondant à des problèmes de mathématiques tirés de la vie courante ou à des applications scientifiques (*p. ex., calcul de salaire brut, de la trajectoire d'un missile, de la consommation d'essence*).

- C2.3** concevoir des algorithmes traitant des exceptions (*p. ex., division par zéro, racine carrée d'un nombre négatif, 0⁰ [zéro exposant zéro]*).

- C2.4** comparer différents algorithmes qui répondent au même problème.

Développement de programmes

- C3.1** modéliser un problème de programmation en utilisant diverses techniques (*p. ex., organigramme, langage de modélisation UML, diagramme de flux, pseudocode*).

- C3.2** développer des sous-programmes (*p. ex., procédure, fonction, macro*) qui répondent à un problème donné (*p. ex., calcul de puissance, affichage d'un élément visuel, manipulation d'une donnée*).

- C3.3** appliquer le principe de modularisation au développement de programmes (*p. ex., en créant des sous-programmes pour éviter la duplication de blocs de code, en séparant au besoin le code en plusieurs fichiers*).

- C3.4** concevoir des interfaces utilisateurs conviviales (*p. ex., interface d'une console, interface d'une page Web, interface graphique*) qui répondent aux besoins des utilisatrices et utilisateurs.

- C3.5** développer des programmes répondant à des problèmes donnés en utilisant des modules préalablement conçus à cet effet (*p. ex., fonction, interface utilisateur, documentation*) et des fonctions prédéfinies (*p. ex., générateur de nombres aléatoires, gestionnaire d'événements, convertisseur de chaînes de caractères*).

D. ENJEUX SOCIÉTAUX ET PERSPECTIVES PROFESSIONNELLES

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- D1.** déterminer des mesures favorables pour l'environnement et la santé publique concernant l'utilisation du matériel informatique.
- D2.** décrire diverses possibilités de carrière en informatique, en y associant les exigences sur le plan de la formation et de la reconnaissance professionnelle.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Environnement et santé

- D1.1** expliquer l'impact de l'industrie de l'informatique sur l'environnement et la santé publique en identifiant les facteurs bénéfiques (*p. ex., modélisation des forces environnementales et de leur impact au niveau planétaire, séquençage du génome humain, repliement des protéines*) et les facteurs nuisibles (*p. ex., volume et toxicité des déchets, consommation électrique, utilisation du papier*).
- D1.2** décrire des initiatives dans le domaine technologique (*p. ex., projets et programmes gouvernementaux, privés ou communautaires*) qui font la promotion d'une gestion durable et de la protection de l'environnement.
- D1.3** déterminer des stratégies pour réduire la consommation des systèmes informatiques, et pour réutiliser et recycler du matériel informatique (*p. ex., éteindre le moniteur en fin de journée, réaffecter des ordinateurs obsolètes à des tâches mieux appropriées, recycler les cartouches d'encre*).
- D1.4** décrire des moyens pour prévenir (*p. ex., poste de travail ergonomique, exercice musculaire*) des problèmes de santé liés à l'usage des ordinateurs (*p. ex., trouble musculosquelettique, trouble de la vue*).

Carrière et formation

- D2.1** comparer des choix de carrière possibles en informatique (*p. ex., programmation, développement de sites Web, administration de systèmes*),

notamment en ce qui a trait aux tâches à accomplir, aux conditions salariales et aux possibilités d'avancement.

- D2.2** répertorier les programmes de formation offerts en informatique en identifiant les préalables et les options éducatives qui existent en français.
- D2.3** identifier des occasions et des moyens d'obtenir de l'expérience en informatique (*p. ex., bénévolat, éducation coopérative, salon de l'emploi, mentorat*).
- D2.4** décrire des services de soutien qui favorisent l'orientation vers des métiers non traditionnels en informatique (*p. ex., programme de mentorat, groupe de soutien virtuel, programme postsecondaire spécialisé, association syndicale*).
- D2.5** décrire les compétences essentielles (*p. ex., communication verbale, prise de décisions, résolution de problèmes*) ainsi que les habitudes de travail (*p. ex., habitudes associées à la sécurité au travail, à la capacité de travailler en équipe et de façon autonome, à la fiabilité, au service à la clientèle*) du Passeport-compétences de l'Ontario (PCO) qui sont indispensables pour réussir dans l'industrie de l'informatique.
- D2.6** créer un portfolio en sélectionnant les travaux ou les réalisations (*p. ex., logiciel, code source, diagramme*) qui témoignent de ses habiletés, de ses compétences, de ses expériences et des certifications acquises en informatique.

Génie informatique, 12^e année

cours préuniversitaire

ICS4U

Dans ce cours, l'élève approfondit ses connaissances théoriques liées au développement de logiciels, aux algorithmes et aux structures de données élémentaires, ainsi que ses habiletés pratiques en concevant des fonctions récursives et en développant des programmes de recherche et de tri. Dans le cadre d'un projet d'envergure en équipe, l'élève applique des techniques de développement et de gestion du cycle de vie du logiciel. Le cours amène aussi l'élève à explorer les enjeux sociétaux, les progrès en informatique ainsi que les possibilités de carrière et de formation professionnelle dans ce secteur.

Préalable : Introduction à l'informatique, 11^e année, cours préuniversitaire

A. TECHNIQUES DE PROGRAMMATION

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- A1.** appliquer les règles de syntaxe et de sémantique d'un langage de programmation.
- A2.** analyser des algorithmes et des structures de données.
- A3.** documenter un logiciel afin d'en faciliter sa maintenabilité.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Syntaxe et sémantique

- A1.1** dégager les principes de la conception modulaire (*p. ex., découpage en sous-problèmes simples, utilisation du code existant, travail à partir d'interfaces définies*).
- A1.2** appliquer les règles de syntaxe d'un langage de programmation.
- A1.3** appliquer des stratégies de décomposition du code d'un logiciel en plusieurs fichiers (*p. ex., classe, constante, fonction*).
- A1.4** appliquer les règles de syntaxe et de sémantique d'un langage de programmation telles que celles gouvernant les opérateurs arithmétiques et logiques (*p. ex., division euclidienne, arithmétique des nombres réels, expression logique composée*).

Algorithmes et structures de données

- A2.1** décrire des structures de données avancées et les règles de transtypage qui y sont associées (*p. ex., structure définie par la programmeuse ou le programmeur, conversion d'une chaîne de caractères en nombre*).
- A2.2** définir le concept de type de données abstrait et ses avantages en programmation.
- A2.3** définir le concept de récursivité et ses applications (*p. ex., calcul des éléments de la suite de Fibonacci*).
- A2.4** analyser des algorithmes qui permettent de manipuler des tableaux à une ou à deux dimensions (*p. ex., tableau de coordonnées de géopositionnement, tableau de structures*).

A2.5 manipuler des fichiers en lecture et en écriture (*p. ex., base de données, fichier binaire, fichier XML*) à l'aide d'un programme.

A2.6 analyser des algorithmes qui permettent de rechercher et de trier des données (*p. ex., recherche linéaire, tri à bulles, tri par insertion*).

A2.7 comparer l'efficacité et la complexité d'algorithmes donnés (*p. ex., recherche linéaire, recherche dichotomique*).

Documentation d'un logiciel

A3.1 comparer des ressources documentaires (*p. ex., bloc de code partagé sur Internet, interface de programmation [API], tutoriel*) servant à développer et à déboguer des programmes.

A3.2 documenter le code source d'un programme en suivant les normes de l'industrie (*p. ex., nom et signification des variables, brève explication de l'algorithme principal*) et en utilisant des outils du langage de programmation (*p. ex., générateur automatique de documentation*).

A3.3 documenter un protocole de vérification et de validation d'un programme (*p. ex., essai unitaire, test d'intégration, test de non-régression*).

A3.4 rédiger, en utilisant les termes justes en français, une documentation externe appropriée (*p. ex., fichier d'aide, guide d'installation, manuel de l'utilisateur, foire aux questions [FAQ]*) qui est destinée à l'utilisatrice ou l'utilisateur d'un logiciel donné.

B. DÉVELOPPEMENT DE LOGICIELS

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- B1.** appliquer une méthodologie de développement de logiciels.
- B2.** concevoir des algorithmes répondant aux problèmes donnés.
- B3.** développer des logiciels répondant aux problèmes donnés.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Méthodologie de développement de logiciels

- B1.1** élaborer un cahier des charges détaillé (*p. ex., analyse fonctionnelle, analyse non fonctionnelle, domaine*) présentant les conclusions de l'analyse des besoins de la cliente ou du client.
- B1.2** élaborer un plan de projet en identifiant clairement les étapes du cycle de développement du logiciel, les produits livrables et les stratégies de travail (*p. ex., rôles et responsabilités, tâches à réaliser, protocole de communication*).
- B1.3** tenir à jour un calendrier des tâches (*p. ex., indiquer ce qui est fait, ce qui reste à faire, comment, par qui*).
- B1.4** utiliser les outils de développement de logiciels et de gestion de projets appropriés (*p. ex., système de contrôle de versions, tableur, calendrier, forum, wiki, outil graphique*).
- B1.5** appliquer des techniques de vérification de logiciels (*p. ex., relecture du code par ses pairs, essai unitaire, test d'acceptation*).
- B1.6** comparer des stratégies de distribution d'un logiciel (*p. ex., logiciel payant ou gratuit, licence propriétaire ou libre, format des fichiers*) et de suivi (*p. ex., mise à jour, soutien technique, développement futur*) en tenant compte des objectifs du cahier des charges.

Conception d'algorithmes

- B2.1** concevoir des algorithmes pour comparer des types de données non primitifs (*p. ex., chaîne de caractères, tableau, type de données défini par la programmeuse ou le programmeur*).

B2.2 concevoir des algorithmes pour manipuler, rechercher et trier les éléments d'un tableau à deux dimensions (*p. ex., multiplication de matrices par un nombre entier, permutation des éléments, recherche linéaire, tri à bulle*).

B2.3 concevoir des algorithmes récursifs (*p. ex., calcul de factorielle, conversion d'un nombre entier en chaîne de caractères, tri par fusion, programmation de fractales, traitement de fichier XML*).

Développement de logiciels

B3.1 modéliser un problème à l'aide d'une hiérarchie de classes ou d'un catalogue de fonctions.

B3.2 concevoir l'architecture d'un logiciel (*p. ex., définition de classes, utilisation du langage de modélisation UML*) en respectant son cahier des charges.

B3.3 appliquer le principe de modularisation lors de la conception d'un logiciel (*p. ex., réutiliser des fonctions ou des objets, utiliser des interfaces, utiliser un modèle de conception adaptable*).

B3.4 appliquer des techniques professionnelles (*p. ex., travail en équipe, réusinage, relecture du code par ses pairs, rédaction de commentaires*) pour développer des logiciels.

B3.5 développer chaque module d'un logiciel en respectant les critères du design (*p. ex., élaborer des interfaces utilisateurs, concevoir des algorithmes, rédiger un code source gérant l'entrée et la sortie*).

C. ENJEUX SOCIÉTAUX ET PERSPECTIVES PROFESSIONNELLES

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- C1.** analyser les enjeux sociétaux de diverses activités informatiques.
- C2.** évaluer l'incidence des progrès en informatique dans divers secteurs d'activité.
- C3.** évaluer ses options de carrière et de formation professionnelle en informatique.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Enjeux sociétaux

- C1.1** expliquer les implications éthiques de diverses activités informatiques (*p. ex., respect de la nétiquette, contribution au développement de gratuits, monopolisation de la bande passante du réseau, consultation de documents privés*).
- C1.2** analyser les enjeux sociaux et économiques posés par les technologies de l'information et des communications (*p. ex., application du droit d'auteur, respect des libertés individuelles, adaptation des modèles d'entreprise, sécurité des systèmes informatiques*).
- C1.3** expliquer certains procédés malhonnêtes (*p. ex., plagiat, porte dérobée, virus*) et la pertinence de se munir d'un code de déontologie en informatique tel que ceux suggérés par des associations professionnelles reconnues.

Progrès en informatique

- C2.1** expliquer le rôle de l'informatique dans le développement de nouveaux produits et dans le développement de la connaissance dans d'autres disciplines (*p. ex., radio cognitive, authentification universelle des personnes, nanotechnologie, économie, exploration spatiale, production artistique*).

C2.2 décrire divers progrès réalisés en informatique (*p. ex., sécurité des réseaux, interface homme-machine, Web, système d'exploitation, informatique mobile, bio-informatique*).

C2.3 évaluer (*p. ex., à partir d'une recherche*) les retombées des progrès en informatique (*p. ex., architecture des microprocesseurs, théorie de la complexité, intelligence artificielle, robotique*).

C2.4 présenter son projet à un auditoire cible, oralement ou par écrit, en utilisant un mode de présentation approprié (*p. ex., présentation multimédia, rapport*) et les termes justes en français.

Carrière et formation

C3.1 évaluer ses champs d'intérêt, ses caractéristiques et ses habiletés en fonction de diverses options de carrière en informatique (*p. ex., programmeuse ou programmeur, analyste en informatique de gestion, gestionnaire de projet*).

C3.2 expliquer pourquoi la maîtrise du français constitue un atout sur le marché du travail.

C3.3 expliquer les raisons pour lesquelles les entreprises valorisent l'apprentissage autonome et continu en informatique (*p. ex., apprentissage électronique, implication dans des projets de développement de logiciels libres, contribution au développement de normes techniques*).

- C3.4** évaluer ses compétences essentielles et ses habitudes de travail en fonction de celles qui sont répertoriées dans le Passeport-compétences de l'Ontario (PCO).
- C3.5** maintenir à jour le contenu de son portfolio (*p. ex., logiciel produit, extrait de code source, rapport de stage*) afin de mettre en évidence le développement de ses apprentissages, de ses compétences et de ses habiletés en informatique ainsi que toutes certifications et formations reconnues.

Programmation informatique, 12^e année

cours précollégial

ICS4C

Dans ce cours, l'élève développe un logiciel muni d'une interface utilisateur graphique en utilisant un langage de programmation orientée objet. Dans le cadre d'un projet d'envergure réalisé en équipe, l'élève utilise les outils mis à sa disposition pour développer le logiciel et gérer des tâches diverses. Le cours amène aussi l'élève à explorer les enjeux sociétaux, les progrès en informatique ainsi que les possibilités de carrière et de formation professionnelle dans ce secteur.

Préalable : Introduction à la programmation informatique, 11^e année, cours précollégial

A. TECHNIQUES DE PROGRAMMATION

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- A1.** expliquer les principes de la programmation orientée objet.
- A2.** appliquer des techniques de traitement des données utilisant des structures de données et des fichiers.
- A3.** documenter un logiciel afin d'en faciliter sa maintenabilité.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Programmation orientée objet

- A1.1** définir les concepts clés de la programmation orientée objet (*p. ex., classe, objet, champ, méthode*).
- A1.2** dégager les avantages de la conception modulaire (*p. ex., débogage, maintenance informatique*).
- A1.3** définir les concepts de portée et de visibilité dans un programme orienté objet.
- A1.4** expliquer des principes de la programmation événementielle (*p. ex., Modèle-Vue-Contrôleur, écouteur d'événement*).

Traitement des données

- A2.1** rédiger des expressions constituées d'expressions imbriquées (*p. ex., liées par des opérateurs logiques ou arithmétiques*).
- A2.2** résoudre des problèmes concrets en utilisant divers formats (*p. ex., de date, de prix*) et divers types de données (*p. ex., nombre décimal, chaîne de caractères*).
- A2.3** appliquer des techniques de manipulation des chaînes de caractères (*p. ex., validation de l'entrée des données, conversion numérique*) dans un programme.
- A2.4** définir les caractéristiques d'un tableau telles que les éléments, l'indice, la taille et la dimension.
- A2.5** concevoir une méthode pour manipuler le contenu d'un tableau à une ou à deux dimensions (*p. ex., insertion, suppression, modification*).

- A2.6** appliquer des techniques de manipulation des données externes (*p. ex., base de données, fichier séquentiel, fichier XML*) dans un programme.

Documentation d'un logiciel

- A3.1** identifier des ressources documentaires (*p. ex., bloc de code partagé sur Internet, interface de programmation [API], tutoriel*) servant à développer et à déboguer des programmes.
- A3.2** documenter le code source d'un programme en suivant les normes de l'industrie (*p. ex., nom et signification des variables, valeur de retour d'une fonction, brève explication de l'algorithme principal*) et en utilisant des outils du langage de programmation.
- A3.3** documenter un protocole de vérification et de validation d'un programme (*p. ex., essai unitaire, test d'intégration, test de non-régression*).
- A3.4** rédiger, en utilisant les termes justes en français, une documentation externe appropriée (*p. ex., fichier d'aide, guide d'installation, manuel de l'utilisateur, foire aux questions [FAQ]*) qui est destinée à l'utilisatrice ou l'utilisateur d'un logiciel donné.

B. DÉVELOPPEMENT DE LOGICIELS

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- B1.** appliquer une méthodologie de développement de logiciels.
- B2.** concevoir des algorithmes répondant aux problèmes donnés.
- B3.** développer un logiciel muni d'une interface utilisateur graphique en utilisant les principes de la programmation orientée objet.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Méthodologie de développement de logiciels

- B1.1** définir les étapes du développement d'un logiciel selon la méthodologie adoptée.
- B1.2** élaborer un cahier des charges présentant les conclusions de l'analyse des besoins des clientes et clients.
- B1.3** élaborer un calendrier de soumission des produits livrables en utilisant des outils de gestion de projet (*p. ex., logiciel de gestion de projet, diagramme de Gantt, graphique PERT*).
- B1.4** tenir à jour un calendrier des tâches (*p. ex., indiquer ce qui est fait, ce qui reste à faire, comment, par qui*) en utilisant des outils de travail collaboratif appropriés (*p. ex., courriel, blogue, babillard électronique, calendrier partagé en ligne*).
- B1.5** appliquer un protocole de vérification et de validation d'un programme.

Conception d'algorithmes

- B2.1** expliquer un algorithme (*p. ex., recherche linéaire*) dans un tableau à une dimension.
- B2.2** concevoir des algorithmes pour résoudre divers problèmes concrets impliquant des calculs mathématiques (*p. ex., calculer la quantité de peinture nécessaire pour décorer un appartement, estimer les coûts de la consommation énergétique d'un foyer, calculer le volume d'eau contenu dans un aquarium*).

B2.3 concevoir des algorithmes pour arrondir des quantités avec une précision donnée (*p. ex., calcul de la monnaie à rendre, amortissement du matériel informatique de l'école, conversion de volumes pour une recette de cuisine*).

B2.4 concevoir des algorithmes pour valider les données entrées par l'utilisatrice ou l'utilisateur (*p. ex., majuscule manquante, texte trop long, nombre trop grand, code postal ou numéro de téléphone incorrect*).

Développement de logiciels

- B3.1** modéliser un problème en utilisant diverses techniques (*p. ex., diagramme UML, pseudocode, diagramme de Gantt*) propres à la programmation orientée objet.
- B3.2** définir des classes qui modélisent les objets conceptuels du problème à résoudre.
- B3.3** définir des méthodes (*p. ex., accesseur, mutateur, méthode d'instance, méthode de classe*) avec ou sans arguments pour répondre à divers problèmes donnés.
- B3.4** concevoir des interfaces graphiques adaptées aux besoins des utilisatrices et utilisateurs en respectant les règles de design d'un environnement graphique fenêtré et en appliquant des normes régissant l'accessibilité.
- B3.5** identifier les éléments d'une interface utilisateur graphique qui ne se conforment pas aux critères prédéfinis (*p. ex., norme régissant l'accessibilité, norme élaborée en classe*).

- B3.6** intégrer dans un programme des méthodes d'un objet (*p. ex., recherche, triage, conversion numérique*) et des structures de données (*p. ex., tableau, vecteur, table de hachage*) prédéfinies dans le langage de programmation utilisé ou faisant partie d'une bibliothèque externe (*p. ex., interface de programmation AJAX*).
- B3.7** développer un logiciel qui répond aux événements générés par l'utilisatrice ou l'utilisateur au moyen de l'interface utilisateur graphique.

C. ENJEUX SOCIÉTAUX ET PERSPECTIVES PROFESSIONNELLES

ATTENTES

À la fin du cours, l'élève doit pouvoir :

- C1.** analyser les enjeux sociétaux de diverses activités informatiques.
- C2.** décrire les progrès en informatique.
- C3.** déterminer ses options de carrière et de formation professionnelle en informatique.

CONTENUS D'APPRENTISSAGE

Pour satisfaire aux attentes, l'élève doit pouvoir :

Enjeux sociétaux

- C1.1** expliquer les implications éthiques de diverses activités informatiques (*p. ex., respect de la netiquette, contribution au développement de logiciels gratuits, monopolisation de la bande passante du réseau, consultation de documents privés*).
- C1.2** analyser les enjeux sociaux et économiques posés par les technologies de l'information et des communications (*p. ex., application du droit d'auteur, respect des libertés individuelles, adaptation des modèles d'entreprise, sécurité des systèmes informatiques*).
- C1.3** expliquer, en décrivant certains procédés malhonnêtes (*p. ex., plagiat, porte dérobée, virus*), la pertinence de se munir d'un code de déontologie en informatique tel que ceux suggérés par des associations professionnelles reconnues.

Progrès en informatique

- C2.1** décrire l'évolution des langages de programmation.
- C2.2** citer des innovations récentes dans le domaine de l'informatique (*p. ex., baladodiffusion, téléphonie sur IP, jeu en ligne massivement multijoueurs, jeu vidéo publicitaire*).
- C2.3** décrire des plateformes technologiques et des environnements de programmation novateurs (*p. ex., Web 2.0, téléphone intelligent, environnement sémantique*).

Carrière et formation

- C3.1** déterminer parmi diverses options de carrière en informatique (*p. ex., programmation, analyse en informatique de gestion, administration de systèmes*), celles qui correspondent à ses champs d'intérêt, ses caractéristiques et ses habiletés.
- C3.2** identifier des emplois et des possibilités de travail autonome en informatique (*p. ex., développeuse ou développeur d'applications pour des réseaux sociaux, développeuse ou développeur de sites Web, expert-conseil spécialisé dans une ligne de produits commerciaux ou de logiciels libres*), où la maîtrise du français constitue un atout.
- C3.3** décrire des modes de prestation du perfectionnement professionnel en cours d'emploi (*p. ex., éducation continue, apprentissage en ligne, mentorat, conférence, atelier*) en informatique.
- C3.4** décrire des services de soutien qui favorisent l'orientation vers des métiers non traditionnels en informatique (*p. ex., programme de mentorat, groupe de soutien virtuel, programme postsecondaire spécialisé, association syndicale*).
- C3.5** évaluer ses compétences essentielles et ses habitudes de travail en fonction de celles qui sont répertoriées dans le Passeport-compétences de l'Ontario (PCO).
- C3.6** maintenir à jour le contenu de son portfolio (*p. ex., logiciel produit, extrait de code source, rapport de stage*) afin de mettre en évidence le développement de ses apprentissages, de ses compétences et de ses habiletés en informatique ainsi que toutes certifications et formations reconnues.

GLOSSAIRE

Appliquette. Petit logiciel, souvent une animation ou une application ludique, destiné à être exécuté à l'intérieur d'un logiciel d'application ou d'une page Web.

Conception modulaire. Activité partant du principe qu'un produit constitué de modules indépendants est plus facile à créer et à maintenir à jour.

Contre-mesure de sécurité informatique. Mesure de défense visant à parer une attaque spécifique telle qu'un virus ou un cheval de Troie.

Développeur. Personne responsable de l'analyse et du développement de logiciels.

Gratuitiel. Logiciel d'application qui est disponible gratuitement, bien que l'auteur en reste le propriétaire selon la loi.

Logiciel. Ensemble de programmes, de fichiers d'aide, de fichiers multimédias et autres ressources nécessaires au bon fonctionnement de l'ordinateur en vue d'accomplir des tâches données. En informatique, tout ce qui n'est pas logiciel est matériel.

Logiciel d'application. Logiciel permettant à l'utilisatrice ou l'utilisateur d'accomplir une tâche spécifique; par exemple, écrire une lettre, naviguer dans Internet.

Logiciel libre. Logiciel généralement gratuit et couvert par une licence qui garantit certaines libertés fondées sur l'accès au code source, dont la possibilité de l'étudier, de le modifier et de le redistribuer.

Macro. Série de commandes permettant de combiner diverses fonctions d'un logiciel d'application; par exemple, sauvegarder un document sous différents formats à l'aide d'un simple clic.

Maintenabilité. Facilité avec laquelle il est possible de modifier ou de mettre à jour un logiciel.

Matériel informatique. Tous les composants matériels (électroniques, électriques et autres) qui constituent un ordinateur. En informatique, tout ce qui n'est pas matériel est logiciel.

Ordinateur. Pour les besoins de ce document, un ordinateur est défini comme étant un appareil programmable tel qu'un ordinateur personnel, un téléphone cellulaire ou une tablette PC.

Périphérique. Composant matériel, à l'exception du microprocesseur, de la carte mère et de la mémoire vive, servant à l'entrée ou la sortie de données.

Programme. Ensemble d'instructions formulées dans un langage de programmation ou en code machine et destinées à être exécutées par un ordinateur.

Programmeur. Personne responsable du codage d'un programme informatique dans un langage donné.

Programmeur-analyste. Voir développeur.

Script. Petit programme destiné à être interprété par une application munie d'un interpréteur, telle que le navigateur Web ou la console.

Sémantique. Sens d'un programme informatique tel que le comprend un compilateur ou un interpréteur avant de le traduire en code machine.

Syntaxe. Règles spécifiant la manière d'écrire les mots-clés, les expressions et les instructions d'un programme présenté dans un langage donné.

Système d'exploitation. Logiciel qui gère les composants matériels et logiciels d'un ordinateur et qui sert d'interface à l'utilisatrice ou l'utilisateur.

Téléphonie sur IP. Téléphonie qui offre les mêmes services que le téléphone, mais en utilisant Internet et sa suite de protocoles TCP/IP pour transmettre la voix.

Transtypage. Action de forcer la modification de type d'une variable.

Wiki. Logiciel d'application se présentant comme un site Web dont les pages peuvent être commentées et entièrement éditées en ligne.

Le ministère de l'Éducation tient à remercier toutes les personnes, les groupes et les organismes qui ont participé à l'élaboration et à la révision de ce document.



Imprimé sur du papier recyclé

08-058

ISBN 978-1-4249-8105-2 (imprimé)

ISBN 978-1-4249-8106-9 (PDF)

ISBN 978-1-4249-8107-6 (TXT)

© Imprimeur de la Reine pour l'Ontario, 2008