



**Informatique au
secondaire 2 (20S)
secondaire 3 (30S)
et secondaire 4 (40S)**

*Cadre manitobain
des résultats d'apprentissage*

***INFORMATIQUE AU SECONDAIRE 2 (20S),
SECONDAIRE 3 (30S) ET SECONDAIRE 4 (40S)***

Cadre manitobain des résultats d'apprentissage

2004

Éducation, Citoyenneté et Jeunesse Manitoba

Données de catalogage avant la publication d'Éducation, Citoyenneté et Jeunesse Manitoba

004.0712 Informatique au secondaire 2 (20S), secondaire 3 (30S) et secondaire 4 (40S) :
cadre manitobain des résultats d'apprentissage.

Comprend des références bibliographiques.

ISBN 0-7711-3297-2

1. Informatique — Programme d'études. 2. Informatique — Études et enseignement (secondaire) — Manitoba. 3. Gestion — Informatique — Études et enseignement (secondaire) — Manitoba. I. Manitoba. Éducation, Citoyenneté et Jeunesse Manitoba.

Droits d'auteur © 2004, la Couronne du chef du Manitoba représentée par le ministre de l'Éducation, de la Citoyenneté et de la Jeunesse. Division du Bureau de l'éducation française, 1181, avenue Portage, salle 509, Winnipeg (Manitoba) R3G 0T3.

Les auteurs se sont efforcés de citer leurs sources et de se conformer à la législation concernant les droits d'auteur. Néanmoins, toute omission à cet égard devrait être signalée à Éducation, Citoyenneté et Jeunesse Manitoba pour qu'elle soit rectifiée dans les prochaines éditions.

Dans le présent document, les mots de genre masculin appliqués aux personnes désignent les hommes et les femmes.

Éducation, Citoyenneté et Jeunesse Manitoba autorise la reproduction de ce document à des fins éducatives et non lucratives.

Ce document est affiché sur le site d'Éducation, Citoyenneté et Jeunesse Manitoba à l'adresse <<http://www.edu.gov.mb.ca/ms4/progetu/informatique/index.html>>.

Remerciements

Éducation, Citoyenneté et Jeunesse Manitoba remercie sincèrement les personnes suivantes pour leur contribution à l'élaboration de *Informatique au secondaire 2 (20S), secondaire 3 (30S) et secondaire 4 (40S) : cadre manitobain des résultats d'apprentissage*.

Comité d'élaboration des résultats d'apprentissage *Informatique au secondaire 2 (20S), secondaire 3 (30S) et secondaire 4 (40S)*

Kathy Athayde	Kildonan-East Collegiate	Division scolaire River East Transcona
Normand Châtel	Collège Béliveau	Division scolaire Louis-Riel
Bob Dagg	Elton Collegiate	Division scolaire Rolling River
Sylvain Delaquis	École/Collège régional Gabrielle-Roy	Division scolaire franco-manitobaine
Dino Difabrizio	Technical Vocational High School	Division scolaire de Winnipeg
Cheryl Klimack	Crocus Plains Regional Secondary School	Division scolaire de Brandon
Kevin Rowan	Fort Richmond Collegiate	Division scolaire Pembina Trails

Représentants de l'industrie

Geoff Bresch	Manitoba Information Technology Industries Inc.
Scott Greenlay	Greenridge Business Systems Corporation
Norm Lee	MindSet, Manitoba Network for Science and Technology
Magahay, Wendy	Applications du travail et de l'apprentissage; Un projet national. Collège Camosun.
Bruce Popham	Manitoba Solution Centre, EDS Canada Inc.
David Tetlock	Groupe Investors//Investors Group

Personnel d'Éducation, Citoyenneté et Jeunesse Manitoba

Jean-Vianney Auclair	Directeur	Division du Bureau de l'éducation française
Monique Barnabé-Saurette	Opératrice de traitement de texte	Division du Bureau de l'éducation française
Normand Châtel	Réviseur	Collège Béliveau
Françoise Chiasson	Traductrice	Moncton
Diane Cooley	Gestionnaire de projet	Division des programmes scolaires
Sylvain Delaquais	Réviseur	École/Collège régional Gabrielle-Roy
Darryl Gervais	Responsable de projet/rédacteur	Division des programmes scolaires
Michelle Larose-Kuzenko	Conseillère pédagogique	Division du Bureau de l'éducation française
Susan Letkemann	Éditrice	Division des programmes scolaires
Nathalie Montambeault	Opératrice de traitement de texte	Division des programmes scolaires
Sam Steindel	Coordonnateur	Division des programmes scolaires

Atelier ATA

Pendant l'élaboration de ce document, les membres du comité d'élaboration ont participé à un atelier de Applications du travail et de l'apprentissage (ATA) animé par Wendy Magahay. Les membres du comité ont visité des sites de travail et interviewé des employés en ce qui a trait aux habiletés, connaissances et attitudes essentielles requises pour des carrières dans le domaine de la programmation informatique et de la gestion de projets. Les organisations suivantes se sont portées volontaires pour participer et ont facilité l'accès à leur personnel pour cet atelier.

EDS Canada
Encore Business
La Great-West, compagnie d'assurance-vie
Greenridge Business Systems Corporation
Manitoba Hydro
Commission des accidents de travail du Manitoba

Table des matières

Remerciements	iii
Introduction	1
Un cadre pour l'informatique	1
Les technologies de l'information et de la communication	1
L'accès à l'ordinateur au moyen de langage de programmation	1
Apprendre, enseigner et évaluer	2
L'élève comme apprenant	2
L'enseignant comme animateur	2
Préalables	3
Communication des résultats	3
Résultats d'apprentissage en Informatique au secondaire 2, secondaire 3 et secondaire 4	
Résultats d'apprentissage en informatique	7
Résultats d'apprentissage généraux	7
Résultats d'apprentissage spécifiques	7
Guide de lecture des résultats d'apprentissage	8
Guide de lecture des codes	9
Résultat d'apprentissage général 1	10
Résultat d'apprentissage général 2	12
Résultat d'apprentissage général 3	14
Résultat d'apprentissage général 4	16
Annexes	
Plans de cours proposés	23
Lexique français-anglais	24
Lexique anglais-français	26
Références	31

Introduction

« Utiliser les technologies de l'information aidera les élèves à acquérir la capacité de résoudre des problèmes, d'améliorer leur rendement personnel et de gagner les compétences en pensée abstraite et en pensée critique nécessaires pour apprendre pendant toute leur vie et contribuer à leur milieu. » (*Éducation et Formation professionnelle, Manitoba, La technologie comme compétence de base*, p. 9).

Un cadre pour l'informatique

Le présent document établit un cadre de résultats d'apprentissage pour les élèves qui complètent les cours d'informatique des dernières années d'études secondaires. Le cadre sert de base pour les cours d'informatique des années de secondaire 2 (20S), secondaire 3 (30S) et secondaire 4 (40S). Il remplace les documents relatifs à l'ancien programme : *Informatique cours 205* (1985) et *Informatique cours 305* (1985).

Les cours d'informatique mettent l'accent sur la résolution de problèmes, l'accomplissement des tâches et l'expression de la créativité des élèves que ce soit individuellement ou en groupe. Les élèves apprendront les techniques de programmation et la syntaxe d'un ou de plusieurs langages de programmation. Mieux encore, les élèves apprendront à s'adapter aux changements dans les langages de programmation et étudieront les nouveaux langages au fur et à mesure que ceux-ci seront développés.

Les technologies de l'information et de la communication

Une bonne intégration des technologies de l'information et de la communication (TIC) à tous les programmes d'études aidera les élèves à acquérir la capacité d'utiliser, de gérer et de comprendre les technologies. Avant de commencer les cours d'informatique, les élèves devraient avoir acquis les compétences décrites dans le **stade d'acquisition des aptitudes** du « Continuum des

résultats d'apprentissage basés sur les technologies de l'information » mises en évidence dans *La technologie comme compétence de base* (ci-après désigné comme TCB - Voir annexe A, pages 27 à 35). Les élèves étudiant l'informatique en sont au **stade d'application et d'extension** du continuum dont il est question dans le TCB.

Les cours d'informatique fournissent aux élèves l'occasion de mettre en application ce qu'ils ont appris et d'étendre leurs habiletés en technologies de l'information et de la communication. Plus spécifiquement, « les élèves devraient avoir la possibilité d'explorer différents langages de programmation, systèmes et dispositifs commandés par ordinateur dans des cours spécialisés du secondaire ». (TCB, p. 34). Les cours d'informatique permettent aux élèves d'explorer et de développer des aptitudes dans la résolution de problèmes et les préparent aux études collégiales ou universitaires.

L'accès à l'ordinateur au moyen de langage de programmation

Plusieurs des résultats d'apprentissage de l'informatique ne peuvent être atteints qu'en programmant un ordinateur. C'est pourquoi tous les élèves doivent avoir accès à un ordinateur muni d'un langage de programmation moderne. Le présent cadre ne prescrit pas un langage spécifique. L'enseignant et les élèves décideront quels langages de programmation modernes ces derniers apprendront.

Apprendre, enseigner et évaluer

L'élève comme apprenant

L'élève est un apprenant curieux et actif avec des intérêts, des habiletés et des besoins propres à chaque individu. Il est essentiel de comprendre comment l'élève apprend et comment l'apprentissage peut être évalué. Apprendre implique le fait d'établir des liens entre les connaissances déjà acquises et la construction de nouveaux savoirs de même que le fait d'ajouter de nouveaux contextes ou expériences aux connaissances déjà acquises. Les résultats d'apprentissage en informatique sont pensés en fonction des connaissances déjà acquises. Chaque niveau d'informatique s'élabore à partir du niveau précédent.

L'élève doit devenir un apprenant autonome pour pouvoir poursuivre des études ou faire carrière dans le domaine de l'informatique. Les techniques et les langages de programmation continuent d'évoluer rapidement. On s'attend à ce que l'élève acquière de nouvelles connaissances et s'adapte continuellement aux changements. Pour faire en sorte que l'élève puisse apprendre pendant toute sa vie, il est essentiel qu'il devienne de plus en plus impliqué dans la planification, le développement et l'évaluation de ses propres situations d'apprentissage. Il doit avoir l'occasion de travailler avec les autres élèves pour susciter le questionnement, pour communiquer ses découvertes et pour compléter des projets qui illustrent ce qu'il a appris.

Au début d'un module, l'enseignant, de concert avec les élèves, doit identifier les résultats d'apprentissage attendus et établir des critères de performance qui

correspondent aux résultats d'apprentissage définis par la province. Cet échange entre l'enseignant et l'élève contribue à la planification de l'enseignement, à la mesure, à l'évaluation et à la communication des résultats. Il aide aussi à identifier clairement ce que l'élève doit accomplir, facilitant ainsi le processus d'apprentissage.

Lorsque l'élève est conscient des résultats d'apprentissage attendus, il est davantage concentré sur le fait d'apprendre et pourra vraisemblablement mieux prendre la mesure de son progrès et de ses réussites. De plus, il peut participer à l'élaboration de critères de mesure et d'évaluation mieux adaptés. Les méthodes d'évaluation doivent être valables, fiables et justes pour l'élève.

L'enseignant comme animateur

Alors que l'enseignant est expert dans l'art d'animer les groupes et d'utiliser des stratégies pédagogiques et d'évaluation propres à stimuler l'apprentissage des élèves, il n'est pas nécessairement expert en programmation d'ordinateur. Si les cours d'informatique doivent demeurer pertinents et actuels, l'enseignant doit cependant être ouvert à l'idée des innovations informatiques et être disposé à apprendre en même temps que les élèves.

Comme animateur, l'enseignant

- assiste l'élève à naviguer dans l'information offerte par les technologies de l'information et de la communication et par d'autres sources;
- dirige l'élève alors qu'il rassemble, organise, analyse et présente l'information;
- aide l'élève à reconnaître, développer, perfectionner, consolider et étendre ses connaissances, aptitudes et compétences.

Préalables

Il n'y a pas de préalables pour les cours d'informatique. Le cadre est conçu de telle sorte que les cours Informatique 30S et 40S sont élaborés à partir des résultats des cours 20S et 30S. Les écoles ont le choix d'offrir un, deux ou trois pleins crédits en informatique. Si l'école n'offre pas le cours au secondaire 2, les résultats d'apprentissage du cours 20S devront alors être intégrés au cours 30S. De même, si l'école n'offre pas le cours au secondaire 3, les résultats d'apprentissage du cours 30S devront alors être intégrés au cours 40S.

Communication des résultats

Tous les cours d'informatique (20S, 30S et 40S) sont des cours plein-crédit. Tel que mentionné dans le *Guide des matières enseignées* du Ministère, le code du cours d'informatique est 0280. Utilisez ce numéro en remplissant les formulaires du personnel professionnel de l'école (PP) et en entrant les résultats des élèves dans le Système des résultats des élèves (SRE).

Notes

Résultats d'apprentissage en Informatique au secondaire 2, secondaire 3 et secondaire 4

Résultats d'apprentissage en informatique

Ce cadre contient les résultats d'apprentissage généraux et spécifiques pour les cours d'informatique. Les résultats d'apprentissage sont des énoncés concis de l'apprentissage attendu chez les élèves des cours d'informatique à la fin du secondaire 2, secondaire 3 et secondaire 4. Cet apprentissage comprend :

- **Les connaissances** : les élèves ont besoin de connaître les faits, les concepts, les principes et les aspects généraux. L'apprentissage de connaissances en informatique comprend le vocabulaire et les fonctions du matériel informatique, le vocabulaire et la syntaxe des langages de programmation informatique, les normes et conventions de programmation, les pratiques de gestion par projet, les questions de santé et sécurité ainsi que l'information sur les carrières.
- **Les habiletés et stratégies** : les élèves ont besoin de connaître et d'appliquer les processus et les stratégies dans le développement des habiletés comme la résolution de problèmes, la pensée critique, la métacognition, la communication et le travail d'équipe.
- **Les attitudes** : les élèves ont besoin de développer des habitudes et des attitudes telles que la détermination d'objectifs, l'approche d'une tâche au moyen de la pensée stratégique, la santé et la sécurité de sa personne, le comportement éthique et moral et le fait de réfléchir à leur propre rendement.

Résultats d'apprentissage généraux

Les résultats d'apprentissage généraux (RAG) sont des énoncés généraux décrivant l'apprentissage des élèves. Les résultats d'apprentissage généraux sont inter-reliés

et interdépendants. Chaque résultat doit être réalisé au moyen d'une variété de stratégies d'apprentissage et d'expériences.

Les quatre résultats d'apprentissage généraux du programme cadre Informatique reflètent les quatre compétences de base :

1. **Relations humaines** : l'élève fait preuve de tolérance, d'esprit d'équipe, de leadership et d'une conduite responsable, éthique et morale.
2. **Littératie et communication** : l'élève possède des qualités de communication efficace dans ses composantes orales et écrites.
3. **Résolution de problèmes** : l'élève démontre des aptitudes à résoudre des problèmes en cherchant des solutions aux défis informatiques.
4. **Technologie** : l'élève développe des aptitudes à utiliser, à gérer et à comprendre les technologies de l'information et de la communication en explorant les langages de programmation et les dispositifs commandés par ordinateur.

Résultats d'apprentissage spécifiques

Chaque résultat d'apprentissage général est élaboré à partir d'une séquence de résultats d'apprentissage spécifiques regroupés en catégorie sous un titre. Les résultats spécifiques (RAS) sont pertinents pour tous les élèves dans une variété d'environnements d'apprentissage et s'accumulent d'année en année. On s'attend à ce que les élèves puissent faire la démonstration des résultats d'apprentissage spécifiques au niveau où ils

sont rendus tout en maintenant et en améliorant les résultats déjà acquis aux niveaux antérieurs.

Guide de lecture des résultats d'apprentissage

Les résultats d'apprentissage pour chaque cours d'informatique (20S, 30S et 40S) sont identifiés dans des colonnes séparées dans les tableaux présentés aux pages suivantes. Une lecture verticale de chaque page permet de voir les résultats d'apprentissage attendus à la fin de chaque cours. La lecture horizontale permet de saisir le continuum des résultats d'apprentissage d'un cours à l'autre. Une flèche (⇒) signale qu'un résultat d'apprentissage d'un cours doit être revu, approfondi et maintenu dans le ou les cours subséquents.

Chaque résultat d'apprentissage est numéroté :

- le premier chiffre indique le RAG
- le second chiffre indique le thème pour un regroupement ou groupe de RAS
- le troisième chiffre indique le RAS.

Les notes à l'intention de l'enseignant ne sont ni exhaustives ni directives; elles visent plutôt à clarifier l'objectif des résultats d'apprentissage et suggèrent des moyens d'atteindre ces résultats en termes généraux.

Guide de lecture des codes

← La lecture horizontale de chaque page permet de saisir le continuum des résultats d'apprentissage d'un cours à l'autre. →

Résultat d'apprentissage général 1				Notes à l'intention de l'enseignant
Relations humaines : l'élève fait preuve de tolérance, d'esprit d'équipe, de leadership et d'une conduite responsable, éthique et morale.				
Désignation du cours	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)	<p>Notes à l'intention de l'enseignant</p> <p>Au fur et à mesure que les élèves avancent dans leurs programmes de cours Secondaire 2, 3 et 4 (S2, S3 et S4), le travail d'équipe prend de plus en plus d'importance dans les cours d'informatique.</p> <p>S2 : Initiation au travail d'équipe, chaque élève doit écrire et documenter un code pour une petite portion du programme. Ils échangent alors leurs codes et continuent le travail avec le code d'un autre élève.</p> <p>S3 : Afin de mettre l'accent sur la communication et les aptitudes au travail d'équipe, s'assurer que les projets de groupe ont moins d'envergure que les projets individuels.</p> <p>S4 : Structurer les projets de groupe de manière à concevoir un programme écrit par plus d'un élève. Chaque projet implique la division des responsabilités, la conception descendante, le test en cours, la révision structurée, la gestion de projet, la conception de l'interface, la documentation et la présentation du produit.</p> <p>S'assurer que chaque projet comprend des occasions d'apprentissage autodirigé.</p>
RAS— Résultat d'apprentissage spécifique	RAS 1.1 Travail d'équipe	RAS 1.1	RAS 1.1	
Titre pour un regroupement de RAS.	1.1.1 développer des habiletés de communication interpersonnelle par le travail d'équipe, par la réflexion en groupe et par l'échange d'information parmi les membres du groupe.	1.1.1	1.1.1	
Le numéro du RAS s'applique à chaque cours – le premier chiffre indique le RAG – le second chiffre indique le titre – le troisième chiffre indique le RAS	1.1.2 participer à au moins un projet de groupe durant le cours.	1.1.2	1.1.2 participer à plusieurs projets de groupe durant le cours, en travaillant comme membre d'équipe mais aussi en réalisant des tâches individuelles au sein du groupe.	
Le RAS devrait être atteint à la fin d'un cours donné.	RAS 1.2 Société et Environnement	RAS 1.2	RAS 1.2	
	1.2.1 décrire l'évolution des technologies de l'information et de la communication en mettant l'accent sur l'évolution du matériel informatique.	1.2.1	1.2.1 examiner les tendances actuelles et les prévisions pour l'avenir des technologies de l'information et de la communication.	
	1.2.2 discuter des implications de l'évolution progressive du matériel informatique pour l'environnement et la société.	1.2.2	1.2.2 discuter les sujets d'actualité ayant trait à l'impact des technologies de l'information et de la communication sur l'environnement et la société.	

← La lecture verticale de chaque page permet de voir les résultats d'apprentissage attendus à la fin de chaque cours. →

Une **flèche** (→) indique qu'un RAS d'un cours précédent doit être revu, approfondi et maintenu dans le ou les cours suivants.

Les **notes à l'intention de l'enseignant** précisent l'objectif du RAS et suggèrent des moyens d'atteindre ces résultats en termes généraux.

Résultat d'apprentissage général 1

Relations humaines : l'élève fait preuve de tolérance, d'esprit d'équipe, de leadership et d'une conduite responsable, éthique et morale.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 1.1 Travail d'équipe	1.1.1 développer des habiletés de communication inter-personnelle par le travail d'équipe, par la réflexion collective et par l'échange d'information parmi les membres du groupe.	1.1.1 ⇒	1.1.1 ⇒
	1.1.2 participer à au moins un projet de groupe durant le cours.	1.1.2 ⇒	1.1.2 participer à plusieurs projets de groupe durant le cours, en travaillant comme membre d'équipe mais aussi en réalisant des tâches individuelles au sein du groupe.
RAS 1.2 Société et Environnement	1.2.1 décrire l'évolution des technologies de l'information et de la communication en mettant l'accent sur l'évolution du matériel informatique.	1.2.1 décrire l'évolution des technologies de l'information et de la communication en mettant l'accent sur l'histoire des logiciels et l'évolution des langages de programmation jusqu'à aujourd'hui.	1.2.1 examiner les tendances actuelles et les prévisions pour l'avenir des technologies de l'information et de la communication.
	1.2.2 discuter des implications de l'évolution progressive du matériel informatique pour l'environnement et la société.	1.2.2 ⇒	1.2.2 discuter des sujets d'actualité ayant trait à l'impact des technologies de l'information et de la communication sur l'environnement et la société.

Notes à l'intention de l'enseignant

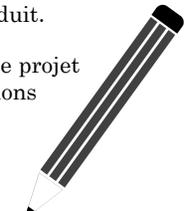
Au fur et à mesure que les élèves avancent dans leurs programmes de cours Secondaire 2, 3 et 4 (S2, S3 et S4), le travail d'équipe prend de plus en plus d'importance dans les cours d'informatique.

S2 : Pour initier au travail d'équipe, chaque élève doit écrire et documenter un code pour une petite portion du programme. Ils échangent alors leurs codes et continuent le travail avec le code d'un autre élève.

S3 : Afin de mettre l'accent sur la communication et les aptitudes au travail d'équipe, s'assurer que les projets de groupe ont moins d'envergure que les projets individuels.

S4 : Structurer les projets de groupe de manière à concevoir un programme écrit par plus d'un élève. Chaque projet implique la division des responsabilités, la conception descendante, le test en cours, la révision structurée, la gestion de projet, la conception de l'interface, la documentation et la présentation du produit.

S'assurer que chaque projet comprend des occasions d'apprentissage autodirigé.



(suite...)

Résultat d'apprentissage général 1 (suite)

Relations humaines : l'élève fait preuve de tolérance, d'esprit d'équipe, de leadership et d'une conduite responsable, éthique et morale.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 1.3 Comportement éthique	1.3.1 discuter les effets du délit informatique, du piratage, du plagiat des codes, du vol de logiciel, de la propagation des virus et de la destruction volontaire de données.	1.3.1 ⇒	1.3.1 ⇒
	1.3.2 utiliser des pratiques responsables et sécuritaires en utilisant les ordinateurs et les réseaux.	1.3.2 ⇒	1.3.2 ⇒

Notes à l'intention de l'enseignant

Les cours d'informatique doivent être consacrés à l'enseignement plus qu'à écrire du code de programme. Les représentants de l'industrie répètent sans cesse que les relations humaines sont l'aspect le plus important de la programmation informatique réussie. Les programmes sont créés pour répondre à un besoin humain. On doit insister pour développer et étoffer les relations humaines tout au long du cours d'informatique.



Résultat d'apprentissage général 2

Littératie et communication : l'élève fait preuve de qualités de communication efficace dans ses composantes orales et écrites.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 2.1 Documentation	<p>2.1.1 expliquer le besoin et la raison d'être de la documentation.</p> <p>2.1.2 documenter le code du programme pour tous ses travaux.</p> <p>2.1.3 nommer de façon appropriée les variables en utilisant les conventions acceptées.</p>	<p>2.1.1 ⇒</p> <p>2.1.2 ⇒</p> <p>2.1.3 ⇒</p>	<p>2.1.1 ⇒</p> <p>2.1.2 ⇒</p> <p>2.1.3 ⇒</p>
RAS 2.2 Présentation orale	<p>2.2.1 présenter un exposé informel d'idées et du code à l'enseignant et aux autres élèves.</p>	<p>2.2.1 concevoir et faire une présentation orale individuelle à caractère formel sur la résolution d'un problème.</p>	<p>2.2.1 concevoir et participer à une présentation orale de groupe à caractère formel sur la résolution d'un problème.</p>
RAS 2.3 Carrières	<p>2.3.1 explorer les carrières en technologies de l'information et de la communication ayant un lien avec l'informatique.</p>	<p>2.3.1 faire une recherche sur une carrière en technologies de l'information et de la communication ayant un lien avec l'informatique.</p>	<p>2.3.1 faire une recherche sur une carrière en technologies de l'information et de la communication ayant un lien avec l'informatique et la gestion de projet.</p> <p>2.3.2 faire une recherche sur les différentes options de formation postsecondaire pour des études ultérieures en informatique.</p>

Notes à l'intention de l'enseignant

La documentation décrit l'intention du programmeur et dirige l'utilisation du produit.
 S2 : Insiste sur l'importance d'une documentation appropriée dès le début en mettant l'accent sur l'explication des fonctionnements du code du programme.
 S3 : La documentation pour tous les travaux devrait inclure une description des données d'entrée et de sortie pour chaque module à l'intérieur du programme.
 S4 : La documentation devrait inclure une communication écrite séparée décrivant le problème que le programme règle et renseignant l'utilisateur sur l'utilisation du programme. Par exemple : organigramme, manuel d'utilisateur, fichier d'aide, foire aux questions.

Les autres élèves, les membres de la communauté et les Aînés peuvent profiter de cette recherche si l'élève partage ses connaissances au moyen d'exposés ou de sites Internet. La recherche peut comprendre les carrières et les occasions qui sont viables dans les petites communautés ou dans celles qui sont éloignées.



(suite...)

Résultat d'apprentissage général 2 (suite)

Littératie et communication : l'élève fait preuve de qualités de communication efficace dans ses composantes orales et écrites.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
RAS 2.4 Gestion de projet	L'élève pourra	L'élève pourra	L'élève pourra
			2.4.1 définir le terme <i>projet</i> .
			2.4.2 décrire les rôles respectifs du gestionnaire de projet et du chef de projet et distinguer les deux rôles.
			2.4.3 identifier les principales raisons de la réussite ou de l'échec d'un projet.
			2.4.4 préparer un plan de projet comprenant un échéancier, un plan d'action, une estimation des ressources requises, un budget et une analyse de risque.
			2.4.5 résoudre les problèmes liés à un projet.
			2.4.6 motiver les membres de l'équipe.
			2.4.7 faire un rapport d'étape de projet.
			2.4.8 évaluer les résultats d'un projet.

**Notes à l'intention
de l'enseignant**



Résultat d'apprentissage général 3

Résolution de problème : l'élève démontre des aptitudes à résoudre des problèmes en cherchant des solutions aux défis informatiques.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 3.1 Apprendre à apprendre	3.1.1 développer les aptitudes et les stratégies pour l'apprentissage autonome.	3.1.1 ⇒	3.1.1 ⇒ 3.1.2 rechercher de l'information à propos d'un langage de programmation qui lui est inconnu et étudier la syntaxe de façon indépendante en créant un nouveau programme ou en réécrivant un programme existant au moyen d'un langage différent.
	3.2.1 utiliser un modèle structuré pour résoudre les problèmes.	3.2.1 ⇒	3.2.1 ⇒
RAS 3.2 Raisonnement et logique	3.2.2 définir les termes <i>algorithme</i> , <i>pseudocode</i> et <i>affinement progressif</i> .	3.2.2 utiliser l'affinement progressif.	3.2.2 ⇒

Notes à l'intention de l'enseignant

Fournir aux élèves des occasions et des expériences qui sont sources de motivation et d'enrichissement personnel. Stimuler la curiosité, la confiance en soi et la pensée positive des élèves les aidera à devenir des personnes plus autonomes, confiantes dans leurs habiletés et qui valorisent l'apprentissage.

Les élèves ont besoin d'avoir accès aux ressources pédagogiques de l'école ou de la communauté et d'avoir l'occasion d'apprendre de leurs pairs et de leurs mentors.

Modéliser la résolution de problèmes dans les cours de S2 à S4 pour mettre l'accent sur l'analyse et la planification. Reconnaître que certaines personnes préfèrent aborder un problème de façon globale plutôt que dans ses composantes. Réduire l'information et l'appui aux élèves du S3 et S4.

S2 : Présenter le problème et un plan de résolution. Demander aux élèves d'exécuter le plan.
(suite...)



(suite...)

Résultat d'apprentissage général 3 (suite)

Résolution de problème : l'élève démontre des aptitudes à résoudre des problèmes en cherchant des solutions aux défis informatiques.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 3.2 Raisonnement et logique	3.2.3 identifier la suite logique d'étapes pour résoudre un problème.	3.2.3 ⇒	3.2.3 décider de la méthode la plus appropriée pour résoudre un problème.
	3.2.4 confirmer la logique d'un algorithme en traçant à la main un programme d'ordinateur.	3.2.4 ⇒	3.2.4 ⇒
	3.2.5 utiliser la logique booléenne pour tirer des analogies entre les programmes d'ordinateur et les tâches quotidiennes.	3.2.5 ⇒	3.2.5 ⇒
	3.2.6 représenter graphiquement la solution des problèmes numériques et non-numériques en utilisant – les diagrammes – les pseudocodes	3.2.6 ⇒	3.2.6 ⇒
			3.2.7 décrire et reconnaître la différence entre la conception descendante et ascendante.

Notes à l'intention de l'enseignant

S3 : Présenter le problème. Amener les élèves à analyser le problème, à créer un plan pour le résoudre, et à concevoir et à exécuter la solution.

S4 : Amener les élèves à identifier les problèmes et à les résoudre collaborativement avec des rôles bien définis et des tâches réparties entre les membres de l'équipe. À la fin du S4, les élèves seront capables de compléter toutes les étapes du modèle de résolution de problème par eux-mêmes ou comme membres d'une équipe.

Les élèves pourraient avoir utilisé une approche en cinq étapes comme celle qui suit pour résoudre un problème de mathématiques :

1. Comprendre le problème.
 - analyser le problème
2. Élaborer un plan.
 - organiser les étapes pour résoudre le problème
 - envisager une solution
 - créer un prototype
 - créer un algorithme
3. Réaliser le plan.
 - encoder la solution
4. Revenir en arrière pour faire une vérification.
 - tester le code
5. Communiquer les solutions.
 - présenter le produit



Résultat d'apprentissage général 4

Technologie : l'élève développe des aptitudes à utiliser, à gérer et à comprendre les technologies de l'information et de la communication en explorant les langages de programmation et les dispositifs commandés par ordinateur.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 4.1 Conventions et normes de programmation	4.1.1 mettre en place un répertoire et un système de gestion des fichiers pour le stockage, la récupération et la portabilité.	4.1.1 ⇒	4.1.1 ⇒
	4.1.2 expliquer la nécessité de standards et de conventions de programmation.	4.1.2 ⇒	4.1.2 rechercher les standards et conventions de programmation actuels.
	4.1.3 utiliser les standards et les conventions de programmation acceptés pour – la documentation – le style et la lisibilité – le nommage des variables, sous-programmes, etc.	4.1.3 ⇒	4.1.3 ⇒
			4.1.4 développer et documenter des standards d'équipe comme éléments de projet de groupe.

Notes à l'intention de l'enseignant

Modéliser la gestion de fichiers en créant une norme de classe pour l'organisation et le nommage des fichiers.



(suite...)

Résultat d'apprentissage général 4 (suite)

Technologie : l'élève développe des aptitudes à utiliser, à gérer et à comprendre les technologies de l'information et de la communication en explorant les langages de programmation et les dispositifs commandés par ordinateur.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 4.2 Programmation : structures de données	4.2.1 reconnaître qu'une structure de données particulière permet l'exécution de certaines opérations avec efficacité.	4.2.1 reconnaître qu'un tableau uni-dimensionnel apporte la solution à certains problèmes et facilite la résolution d'autres problèmes.	4.2.1 ⇒
	4.2.2 identifier et utiliser les types de données appropriés.	4.2.2 ⇒	4.2.2 ⇒
	4.2.3 obtenir des données d'entrée de l'utilisateur de programme.	4.2.3 valider les données d'entrée de l'utilisateur.	4.2.3 ⇒
	4.2.4 identifier les troncatures et les erreurs d'arrondi liées aux types de données.	4.2.4 ⇒	4.2.4 ⇒
	4.2.5 formater la sortie de données numérique et textuelle.	4.2.5 ⇒	4.2.5 ⇒
	4.2.6 utiliser une chaîne de données pour l'entrée, la concaténation et la sortie de texte.	4.2.6 manipuler le texte y compris par l'extraction, la concaténation et la comparaison.	4.2.6 utiliser un tableau bidimensionnel.
	4.2.7 décrire le lien entre chaque type de données et la quantité de mémoire utilisée.	4.2.7 décrire comment un tableau est stocké intérieurement et comment accéder à un élément individuel.	4.2.7 décrire comment un tableau bidimensionnel est stocké intérieurement.

**Notes à l'intention
de l'enseignant**



(suite...)

Résultat d'apprentissage général 4 (suite)

Technologie : l'élève développe des aptitudes à utiliser, à gérer et à comprendre les technologies de l'information et de la communication en explorant les langages de programmation et les dispositifs commandés par ordinateur.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 4.2 Programmation : structures de données	4.2.8 utiliser des structures simples pour le stockage et la récupération des données.	4.2.8 utiliser les tableaux parallèles pour stocker les données associées.	4.2.8 créer une structure de données définie par l'utilisateur (telle que des enregistrements) qui consiste en une combinaison d'autres objets de données. 4.2.9 créer, stocker, récupérer et manipuler les fichiers séquentiels et les fichiers à accès direct (comme les fichiers bases de données et les fichiers délimités par une virgule).
RAS 4.3 Programmation : structures de contrôle	4.3.1 identifier, utiliser et tracer des structures de contrôle incluant... – la programmation linéaire (séquentielle) – le branchement (IF/THEN/ELSE) – la boucle (définie et indéfinie)	4.3.1 identifier, utiliser et tracer des structures de contrôle incluant... – le branchement multiple au-delà de IF/THEN/ELSE (par exemple : « case », « select case », « switch »)	4.3.1 identifier, utiliser et tracer des structures de contrôle incluant... – la récursivité
RAS 4.4 Programmation : débogage	4.4.1 faire la différence entre les erreurs de compilation, les erreurs d'exécution et les erreurs de raisonnement. 4.4.2 détecter et corriger les erreurs de compilation, les erreurs de durée d'exécution et les erreurs de logique. 4.4.3 repérer les erreurs par le traçage à la main et en utilisant des outils de débogage.	4.4.1 définir un échantillon de données afin de tester un programme. 4.4.2 → 4.4.3 →	4.4.1 → 4.4.2 → 4.4.3 →

Notes à l'intention de l'enseignant

La boucle peut être utilisée pour créer une animation graphique.

Le débogage est une composante de chaque travail de programmation. Encourager les élèves à tester le programme des autres élèves.

Dans les projets d'équipe, les élèves devraient solliciter l'aide des autres membres du groupe avant de demander l'aide de l'enseignant. Cette façon de faire vise à améliorer la communication entre les élèves et les aptitudes à résoudre les problèmes.



(suite...)

Résultat d'apprentissage général 4 (suite)

Technologie : l'élève développe des aptitudes à utiliser, à gérer et à comprendre les technologies de l'information et de la communication en explorant les langages de programmation et les dispositifs commandés par ordinateur.

	Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
	L'élève pourra	L'élève pourra	L'élève pourra
RAS 4.5 Programmation : code réutilisable	4.5.1 utiliser un code déjà existant fourni par l'enseignant pour construire un programme de plus grande taille.	4.5.1 créer un code réutilisable.	4.5.1 créer du code que d'autres élèves peuvent utiliser dans un projet d'équipe de grande envergure. 4.5.2 définir et utiliser les bibliothèques.
RAS 4.6 Programmation : sous-programmes	4.6.1 utiliser les fonctions intégrées avec les paramètres, incluant les fonctions graphique/dessin. 4.6.2 utiliser et créer des sous-programmes qui ne transfèrent pas les paramètres.	4.6.1 ⇒ 4.6.2 ⇒ 4.6.3 reconnaître les sous-programmes qui transfèrent les paramètres. 4.6.4 reconnaître la portée d'une variable (par exemple, distinguer entre variables locales et globales).	4.6.1 ⇒ 4.6.2 ⇒ 4.6.3 créer des sous-programmes qui transfèrent les paramètres. 4.6.4 ⇒

Notes à l'intention de l'enseignant

L'enseignant peut faciliter l'écriture de programmes plus complexes en fournissant des segments de code déjà écrit que les élèves ne comprendraient peut-être pas encore. L'élève devrait progresser de l'utilisation de code à la planification de création de code réutilisable et finalement au partage de code avec les autres.



(suite...)

Résultat d'apprentissage général 4 (suite)

Technologie : l'élève développe des aptitudes à utiliser, à gérer et à comprendre les technologies de l'information et de la communication en explorant les langages de programmation et les dispositifs commandés par ordinateur.

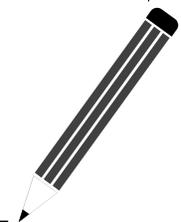
	Secondaire 2 Informatique (20S) L'élève pourra	Secondaire 3 Informatique (30S) L'élève pourra	Secondaire 4 Informatique (40S) L'élève pourra
RAS 4.7 Programmation : objets		<p>4.7.1 définir le terme <i>objet</i>.</p> <p>4.7.2 utiliser la notation point (par exemple : appeler une méthode ou accéder à une propriété).</p>	<p>4.7.1 définir le terme <i>classe</i>.</p> <p>4.7.2 décrire la programmation orientée objet en termes d'encapsulation, d'héritage et de polymorphisme.</p> <p>4.7.3 créer un programme qui utilise les objets et qui crée des classes au moyen d'un langage de programmation orientée objet.</p> <p>4.7.4 limiter la portée des identificateurs publics et privés de façon appropriée.</p>
RAS 4.8 Programmation : algorithmes de tri et de recherche		<p>4.8.1 programmer des algorithmes simples (comme le tri par permutation, le tri par insertion et le tri par sélection).</p> <p>4.8.2 comparer le fonctionnement d'algorithmes.</p> <p>4.8.3 utiliser les algorithmes de recherche y compris linéaire et binaire.</p>	<p>4.8.1 programmer des algorithmes avancés (comme le tri par segmentation, la recherche dichotomique et la génération d'un nombre aléatoire).</p> <p>4.8.2 comparer l'efficacité d'algorithmes.</p> <p>4.8.3 →</p>

Notes à l'intention de l'enseignant

Le but est d'introduire assez vite le langage de la programmation orientée objet et d'amener les élèves à utiliser des objets le plus tôt possible. Les élèves n'ont pas à concevoir leurs propres classes avant le niveau 40S.

Les éléments-clés de la programmation orientée objet sont : l'encapsulation, l'héritage et le polymorphisme. Ce ne sont pas tous les langages qui conviennent à la vraie programmation orientée objet.

Le but est d'en apprendre sur les algorithmes et non de mettre l'accent sur la programmation de tris ou de recherches.



Annexes

Plans de cours proposés

Le tableau qui suit présente une vue d'ensemble des plans de cours proposés pour chaque cours d'informatique au secondaire.

Secondaire 2 Informatique (20S)	Secondaire 3 Informatique (30S)	Secondaire 4 Informatique (40S)
<p>Ce cours d'introduction à la programmation s'adresse à une grande clientèle cible. Les élèves acquièrent des connaissances, des aptitudes et des attitudes qui peuvent s'appliquer à des situations bien au-delà de l'informatique.</p> <p>Sujets :</p> <ul style="list-style-type: none"> • Stratégies pour l'apprentissage autonome, la communication et le travail d'équipe • Histoire du matériel informatique • Influence des TIC sur la société et l'environnement • Éthique • Exploration des carrières en TIC • Standards et conventions de programmation <ul style="list-style-type: none"> • Résolution de problèmes • Concepts de programmation <ul style="list-style-type: none"> – Documentation – Variables et types de données – Entrée/sortie (de données) – Débogage – Branchement (IF/THEN/ELSE), branchement imbriqué – Boucle (définie et indéfinie) – Utilisation des fonctions/méthodes – Fonctions graphiques pré-définies – Écriture des sous-programmes sans paramètres 	<p>Ce cours porte un regard approfondi sur la résolution de problèmes au moyen de la programmation.</p> <p>Sujets :</p> <ul style="list-style-type: none"> • Stratégies pour l'apprentissage autonome, la communication et le travail d'équipe • Histoire des logiciels • Influence des TIC sur la société et l'environnement • Éthique • Recherche de carrières en TIC • Standards et conventions de programmation <ul style="list-style-type: none"> • Résolution de problèmes • Concepts de programmation <ul style="list-style-type: none"> – Documentation – Variables et types de données – Entrée/sortie (de données) – Débogage – Branchement multiple – Boucle – Sous-programmes avec paramètres – Chaînes – Tableau—unidimensionnel, parallèle – Algorithmes—tri simple 	<p>Ce cours de programmation à partir d'un projet reflète la manière de réaliser un projet dans l'industrie. Le cours est structuré de façon à donner aux élèves l'occasion d'écrire des programmes et d'affiner leurs compétences en gestion de projet dans un environnement de programmation en équipe.</p> <p>Sujets :</p> <ul style="list-style-type: none"> • Stratégies pour l'apprentissage autonome, la communication et le travail d'équipe • Tendances actuelles en TIC • Aspects des TIC liés à la société et à l'environnement • Éthique • Recherche de carrières en TIC et présentation des résultats • Études futures en informatique • Gestion de projet • Projet d'équipe : concevoir une simulation par ordinateur • Apprentissage d'un nouveau langage de programmation de façon indépendante <ul style="list-style-type: none"> • Résolution de problèmes • Concepts de programmation <ul style="list-style-type: none"> – Fichiers à accès linéaire et direct – Écriture d'un programme simple au moyen de deux langages différents – Programmation orientée objet – Récursivité – Tableau—bidimensionnel – Analyse d'algorithmes – Tri et recherche

Lexique français-anglais

Affinement progressif	Stepwise refinement	Construits	Constructs
Algorithme	Algorithm	Débogage	Debugging
Algorithmes de recherche	Search algorithms	Défilement	Scrolling
Algorithmes de recherche binaire	Binary search algorithms	Défini par l'utilisateur (structure de données)	User-defined (data structure)
Algorithmes de recherche linéaire	Sequential search algorithms	Délit informatique	Computer crime
Algorithmes de tri	Sort algorithms	Données associées	Related data
Algorithmes de tri avancé	Advanced sort algorithms	Données d'entrée	Data input
Algorithmes de tri simple	Simple sort algorithms	Données de sortie	Data output
Animation graphique	Graphic animation	Encapsulation	Encapsulation
Appeller une méthode	Invoke a method	En ligne	Online
Bibliothèques	Libraries	Enregistrements	Records
Boucle (définie et indéfinie)	Looping (definite and indefinite)	Erreurs d'arrondi	Round-off errors
Branchement (IF/THEN/ELSE)	Branching (IF/THEN/ELSE)	Erreurs de compilation	Compile-time errors
Branchement imbriqué	Nested branching	Erreurs de durée d'exécution	Run-time errors
Branchement multiple	Multiple branching	Erreurs de logique	Logic errors
Casse	Case	Extraction	Extraction
Chaîne de données	String data	Fichiers à accès aléatoire	Random-access files
Classe	Class	Fichiers à accès direct	Random-access files
Code réutilisable	Reusable code	Fichiers délimités par une virgule	Comma-delimited file
Comparaison	Comparison	Fichiers séquentiels	Sequential files
Concaténation	Concatenation	Fonction intégrée	Built-in function
Conception ascendante	Bottom-up (design)	Fonctions graphiques/dessins	Graphic/drawing functions
		Génération d'un nombre aléatoire	Random number generation

Informatique au secondaire 2 (20S), secondaire 3 (30S) et secondaire 4 (40S)

Conception de l'interface	Interfacing	Sous-programmes	Sub-programs
Conception descendante	Top-down (design)	Standards de programmation	Programming standards
Héritage	Inheritance	Stockage	Storage
Identificateur privé	Private identifier	Structures de données	Data structures
Identificateur public	Public identifier	Structures de contrôle	Control structures
Module	Module	Système de gestion des fichiers	File management system
Nommage	Naming	Tableau bidimensionnel	Two-dimensional array
Nommage des variables	Naming of variables	Tableau unidimensionnel	One-dimensional array
Notation point	Dot notation	Tableaux parallèles	Parallel arrays
Objet	Object	Test en cours	On-going testing
Organigramme	Organization chart	Traçage à la main	Hand-tracing
Paramètre de référence	Reference parameter	Transférer les paramètres	Pass parameters
Paramètre de valeur	Value parameter	Tri par insertion	Insertion sort
Paramètres	Parameters	Tri par permutation	Bubble sort
Piratage informatique	Computer hacking	Tri par remontée des bulles	Bubble sort
Polymorphisme	Polymorphism	Tri par segmentation	Quick sort
Portabilité	Portability	Tri par sélection	Selection sort
Portée d'une variable	Scope of a variable	Troncature	Truncation
Programmation linéaire	Linear (sequential) steps	Types de données	Data types
Programmation orientée objet	Object-oriented programming	Variables	Variables
Pseudocode	Pseudocode	Variables globales	Global variables
Recherche dichotomique	Bisection search	Variables locales	Local variables
Récupération	Retrieval		
Récupération des données	Retrieval of data		
Récursivité	Recursion		
Répertoire	Directory		
Révision structurée	Walk-throughs		
Segments de code déjà écrit	Pre-written segments of code		
Sortie de données numériques	Numeric output		
Sortie de données textuelles	Textual output		

Lexique anglais-français

Advanced sort algorithms	Algorithmes de tri avancé	Encapsulation	Encapsulation
Algorithm	Algorithme	Extraction	Extraction
Binary search algorithms	Algorithmes de recherche binaire	File management system	Système de gestion des fichiers
Bisection search	Recherche dichotomique	Global variables	Variables globales
Bottom-up (design)	Conception ascendante	Graphic animation	Animation graphique
Branching (IF/THEN/ELSE)	Branchement (IF/THEN/ELSE)	Graphic/drawing functions	Fonctions graphiques/dessins
Bubble sort	Tri par remontée des bulles (ou tri par permutation)	Hand-tracing	Traçage à la main
Built-in function	Fonction intégrée	Inheritance	Héritage
Class	Classe	Insertion sort	Tri par insertion
Comma-delimited file	Fichiers délimités par une virgule	Interfacing	Conception de l'interface
Comparison	Comparaison	Invoke a method	Appeller une méthode
Compile-time errors	Erreurs de compilation	Libraries	Bibliothèques
Computer crime	Délit informatique	Linear (sequential) steps	Programmation linéaire
Computer hacking	Piratage informatique	Local variables	Variables locales
Concatenation	Concaténation	Logic errors	Erreurs de logique
Constructs	Construits	Looping (definite and indefinite)	Boucle (définie et indéfinie)
Control structures	Structures de contrôle	Module	Module
Data input	Données d'entrée	Multiple branching	Branchement multiple
Data output	Données de sortie	Naming	Nommage
Data structures	Structures de données	Naming of variables	Nommage des variables
Data types	Types de données	Nested branching	Branchement imbriqué
Dot notation	Notation point	Numeric output	Sortie de données numériques
		Object	Objet
		Object-oriented programming	Programmation orientée objet

Informatique au secondaire 2 (20S), secondaire 3 (30S) et secondaire 4 (40S)

Debugging	Débogage	Scope of a variable	Portée d'une variable
Directory	Répertoire	Scrolling	Défilement
One-dimensional array	Tableau unidimensionnel	Search algorithms	Algorithmes de recherche
On-going testing	Test en cours	Selection sort	Tri par sélection
Online	En ligne	Sequential files	Fichiers séquentiels
Organisation chart	Organigramme	Sequential search algorithms	Algorithmes de recherche linéaire
Parallel arrays	Tableaux parallèles	Simple sort algorithms	Algorithmes de tri simple
Parameters	Paramètres	Sort algorithms	Algorithmes de tri
Pass parameters	Transférer les paramètres	Stepwise refinement	Affinement progressif
Polymorphism	Polymorphisme	Storage	Stockage
Portability	Portabilité	String data	Chaîne de données
Pre-written segments of code	Segments de code déjà écrit	Sub-programs	Sous-programmes
Private identifier	Identificateur privé	Textual output	Sortie de données textuelles
Programming standards	Standards de programmation	Top-down (design)	Conception descendante
Pseudocode	Pseudocode	Truncation	Troncature
Public identifier	Identificateur public	Two-dimensional array	Tableau bidimensionnel
Quick sort	Tri par segmentation	User-defined (data structure)	Défini par l'utilisateur (structure de données)
Random number generation	Génération d'un nombre aléatoire	Value parameters	Paramètres de valeur
Random-access files	Fichiers à accès direct (ou fichiers à accès aléatoire)	Variables	Variables
Records	Enregistrements	Walk-throughs	Révision structurée
Recursion	Récurtivité		
Reference parameter	Paramètre de référence		
Related data	Données associées		
Retrieval	Récupération		
Retrieval of data	Récupération des données		
Reusable code	Code réutilisable		
Round-off errors	Erreurs d'arrondi		
Run-time errors	Erreurs de durée d'exécution		

Références

Références

- Education et Formation professionnelle Manitoba.
Informatique cours 205. Winnipeg (Manitoba) : Éducation et Formation professionnelle Manitoba, 1985.
- . *Informatique cours 305*. Winnipeg (Manitoba) : Éducation et Formation professionnelle, 1985.
- . *La technologie comme compétence de base : Vers l'utilisation, la gestion et la compréhension des technologies de l'information*. Ouvrage de référence pour les concepteurs de programmes d'études, les enseignants et les administrateurs. Winnipeg (Manitoba) : Éducation et Formation professionnelle Manitoba, 1998. Disponible en ligne à l'adresse
<<http://www.edu.gov.mb.ca/ks4/docs/supports/tfs/index.html>>
- Éducation et Jeunesse Manitoba. *Guide des matières enseignées - Systèmes informatisés de transmission des dossiers du personnel professionnel*. Publication annuelle. Disponible en ligne à l'adresse
<<http://www.edu.gov.mb.ca/ks4/docs/policy/sth/index.html>>.
- Morvan, Pierre. *Dictionnaire de l'informatique*. Paris : Larousse, © 1996.
- Office québécoise de la langue française. *Le grand dictionnaire terminologique*. [en ligne].
<<http://w3.granddictionnaire.com>>

Notes



Printed in Canada
Imprimé au Canada