

INTRODUCTION

Le programme d'examens du Bureau canadien des conditions d'admission en génie d'Ingénieurs Canada englobe 19 domaines du génie.

Le programme d'examens de chaque domaine se divise en deux catégories d'examens : les examens obligatoires et les examens facultatifs. L'ensemble complet d'examens de génie logiciel comprend dix examens de trois heures. Les examens exigés du candidat seront déterminés en fonction d'une évaluation de ses acquis universitaires. Le candidat pourrait être tenu, à la discrétion de l'association/ordre constituant, de subir des examens dans un domaine autre que le sien.

Avant de subir les examens liés à son domaine, le candidat devra avoir réussi les examens sur les matières de base ou avoir été dispensé de les subir.

L'association/ordre constituant fournira les détails portant sur la date de l'examen, les manuels, le matériel fourni ou exigé et indiquera si les examens ont lieu à livre ouvert ou non.

EXAMENS DE GÉNIE LOGICIEL

GROUPE A

EXAMENS OBLIGATOIRES (SEPT EXAMENS REQUIS)

04-Log-A1 Algorithmes et structures de données

Structures de données fondamentales et algorithmes connexes. Piles et files d'attente, arbres, tables, listes, tableaux, chaînes, ensembles; fichiers et méthodes d'accès. Arbres balancés, structures multi-clés. Recherche. Tri. Techniques de conception d'algorithmes, telles que : diviser pour régner, méthode de l'algorithme glouton, équilibrage, programmation dynamique. Algorithmes se rapportant aux opérations ensemblistes. Graphes, algorithmes de graphes : recherche en profondeur d'abord et en largeur d'abord, arbre recouvrant de longueur minimum, algorithme du plus court chemin. Mesures empiriques et théoriques du rendement des algorithmes. Analyse de la complexité. Problèmes durs, problèmes NP-complets et problèmes insolubles.

04-Log-A2 Structures de calcul

Éléments de base de l'architecture d'ordinateur, comprenant : algèbre booléenne, portes, logique combinatoire et séquentielle, représentation des données de niveau machine; organisation machine, programmation en langage assembleur/machine; organisation de la mémoire, caches, tas, piles; E/S série et parallèle, interruptions, protocoles de bus et accès direct en mémoire (DMA). Éléments de base des systèmes d'exploitation, comprenant : accès simultané, ordonnancement des processus, gestion de la mémoire; protection, accès et authentification; enchaînement et chargement. Éléments de base des bases de données, comprenant : architecture de base de données, langages d'interrogation, transactions. Organisation du système de fichiers et méthodes d'accès (accès séquentiel, séquentiel indexé, hachage extensible, arbres équilibrés), organisation d'index.

04-Log-A3 Conception logicielle

Rôle de l'activité de conception logicielle. Attributs de qualité de la conception logicielle (exactitude, fiabilité, maintenabilité, portabilité, robustesse). Principes de conception logicielle (séparation d'aspects, abstraction, masquage d'information). Architecture logicielle, structures architecturales et vues. Modularité et décomposition. Conception orientée-fonction. Conception orientée-objet. Éléments. Conception d'interfaces. Conception de niveau module. Spécification de conception. Notations (graphiques et langages). Langage UML. Concepts de base des schémas de conception.

04-Log-A4 Systèmes temps réel

Définition et caractéristiques des systèmes temps réel. Systèmes temps réel « durs » et « doux ». Réactions dynamiques de processus physiques simples. Conception de systèmes temps réel (exigences, méthodes de conception, implémentation, tests, interaction homme-machine). Fiabilité et tolérance aux failles. Exceptions et traitement des exceptions. Accès simultané. Synchronisation et communication. Contrôle des ressources. Ordonnancement (exécutif cyclique, taux monotone et priorité des échéances, protocoles de plafond des priorités). Systèmes d'exploitation temps réel. Systèmes intégrés simples.

04-Log-A5 Exigences et spécifications

Sources et techniques d'obtention d'information. Paradigmes de modélisation, comprenant : modélisation de l'information, modélisation comportementale, modélisation du domaine, modélisation fonctionnelle, modélisation des contraintes. Exigences de qualité (par ex. : performance, convivialité, fiabilité, maintenabilité); expression des exigences de qualité de sorte qu'elles soient testables. Priorisation, analyse des options de remplacement, négociation, analyse des risques et analyse des impacts. Gestion des exigences, gestion de la cohérence, analyse des interactions, traçabilité. Documentation des exigences et langages de spécification. Validation, examens et inspections, prototypage, validation des exigences non fonctionnelles. Conception d'essais de réception.

04-Log-A6 Assurance-qualité logicielle

Concepts de validation et de vérification. Cycle de vie des logiciels et application de la validation et de la vérification. Processus d'assurance-qualité logicielle. Définitions de la qualité des produits logiciels. Caractéristiques de qualité. Définitions et spécifications de la qualité ingénierie. Définition et classification des défauts logiciels. Définitions de l'aptitude à l'usage et de la qualité du point de vue du client. Coûts des logiciels, coûts et économie de la qualité. Examens, revues de projet et inspections : concepts généraux, essais unitaires (module/ensemble), essais des sous-systèmes / tests d'intégration, tests de régression, tests basés sur les états, tests fonctionnels traditionnels, tests/analyses logiques, facteurs à considérer lors des tests orientés-objets (polymorphisme et héritage); analyse et tests de la sûreté/des défaillances.

04-Log-A7 Processus logiciels

Cycles de vie des logiciels. Modèles de processus logiciels. Activités de chaque phase. Contrôle et gestion du cycle de vie de logiciels corrects, fiables, maintenables et rentables. Documentation des logiciels. Outils de gestion de projet. Gestion des risques. Communication et collaboration. Cause et effets de l'échec d'un projet. Estimation des coûts et ordonnancement. Facteurs influant sur la productivité et le succès. Mesures de productivité. Planification du changement. Gestion des attentes. Maintenance des logiciels. Gestion de la configuration.

GROUPE B

EXAMENS FACULTATIFS (TROIS EXAMENS REQUIS)

04-Log-B1 Conception orientée-objets avancée

Paradigme de l'orientation objets. Analyse orientée-objets : concepts de base, cas d'utilisation, analyse, stéréotypes et objets, schémas d'analyse. Langages de modélisation d'objets. Conception orientée-objets : concepts de base, stéréotypes et objets de conception, schémas de conception. Programmation orientée-objets : concepts de base, idiomes, principaux langages de programmation orientée-objets (C++, Java). Cadres d'applications. Outils GLAO orientés objets. Diagrammes des états de transition et des interactions. Tests de programmes orientés-objets.

Méthodes avancées d'analyse et de conception orientées-objets. Cadres et schémas de conception. Conception visant la réutilisabilité. Techniques avancées de programmation orientée-objets. Conception utilisant des bases de données orientées-objets et des architectures d'objets distribuées. Conception d'agents logiciels. Projet mettant en jeu l'analyse, la conception et l'implémentation orientées-objets.

04-Log-B2 Interface utilisateur

Principes psychologiques de l'interaction homme-machine. Évaluation des interfaces utilisateur. Ingénierie de la convivialité. Analyse des tâches, conception et prototypage orientés utilisateur. Modèles conceptuels et métaphores. Bien-fondé de la conception logicielle. Conception de fenêtres, de menus et de commandes. E/S vocales et en langage naturel. Temps de réponse et rétroaction. Couleurs, icônes et sons. Internationalisation et localisation. Architectures d'interfaces utilisateur et API. Études de cas et projet.

04-Log-B3 Sécurité/Sûreté

Caractéristiques des problèmes de sécurité. Chiffrement et déchiffrement de base; chiffrement sécurisé, protocoles et pratiques de chiffrement. Systèmes cryptographiques à clés publiques et signatures numériques. Conception logicielle visant la sécurité. Sécurité des systèmes d'exploitation, des bases de données, des ordinateurs personnels, des réseaux informatiques et des communications électroniques. Analyse des risques et planification de la sécurité. Dimension éthique de la sécurité informatique.

04-Log-B4 Fiabilité et tolérance aux fautes

Défaillances logicielles et matérielles. Défaillances, défaillances latentes et pannes. Caractérisation des fonctions de défaillance, distribution de probabilité des défaillances, fonction de l'intensité des défaillances. Définition et mesures de la fiabilité des logiciels. MTTF, MTBF, MTTR, disponibilité, maintenabilité. Utilisation des systèmes, profils opérationnels de Musa et incertitude de type 1. Élimination des défauts et incertitude de type 2. Stabilité de la fiabilité et croissance de la fiabilité. Distinction entre fiabilité matérielle et fiabilité logicielle. Fonction de densité de la probabilité des défaillances et fonction de fiabilité. Systèmes de prédiction de fiabilité, y compris les modèles de prédiction. Tests statistiques. Caractérisation de l'injection, de la détection et de la correction des fautes à diverses phases du cycle de vie. Techniques de prédiction des défaillances restantes, comprenant : injection de fautes, analyse de l'arbre de classification, couverture du code. Mesures générales du cycle de vie pour la production de logiciels fiables, comprenant : prévention des défauts; détection et élimination précoces des

défauts; conception visant la robustesse; utilisation de mesures issues des activités de validation et de vérification; stabilisation des exigences, de la conception, du code et des artéfacts d'essais. Détection des défaillances actives et passives. Programmation N-versions. Traitement et correction des fautes, exceptions, tolérance aux fautes. Analyse par arbre de défaillance. Survivance, fonctions essentielles et fonctionnements en mode dégradé.

04-Log-B5 Modélisation et vérification des logiciels (méthodes formelles)

Modélisation mathématique des logiciels, comprenant : logique, automates finis étendus, algèbre de processus, fonctions, et spécifications algébriques. Raisonnement mathématique de tels modèles, dont les preuves de cohérence, d'intégralité et d'exactitude. Outils de contrôle des types, vérification de bonne forme, simulation, vérification des invariants et des propriétés (par ex. : vérification des interblocages, vérification des modèles), génération de jeux d'essais et génération de code.

04-Log-B6 Gestion avancée de projets logiciels, méthodologies de cycle de vie

Processus de gestion de projets logiciels

Planification, intégration et contrôle des changements; gestion de l'étendue du projet; gestion de la qualité; gestion des coûts; gestion des risques; gestion du calendrier d'exécution; gestion des communications; gestion des ressources humaines et gestion des achats.

Méthodologies de cycle de vie

Modèles de développement (chute d'eau, modèle en V, modèle incrémentiel, spirale, etc.) et techniques (prototypage rapide, salle blanche, technique orientée-objet, etc.). Normes de développement des secteurs militaire/aérospatial et commercial. Activités propres à chaque phase : spécification du logiciel et analyse des exigences; comparaison des architectures logicielles et sélection; conception du logiciel; codage et tests unitaires; tests d'intégration; essai du système et mise en service. Activités liées au cycle de vie : gestion du référentiel, gestion de la qualité logicielle, gestion de la configuration logicielle.

Diagnostic d'un projet industriel réel.

04-Log-B7 Rétroingénierie, maintenance et évolution

Maintenance de logiciels : corrective, perfective et adaptative. Techniques de rétroingénierie de l'architecture et de la conception logicielles à des fins de compréhension des programmes. Réingénierie des systèmes et processus (technique et opérationnelle). Réusinage. Migration (technique et opérationnelle). Analyse d'impact. Gestion des versions et de la configuration. Modèles d'évolution des logiciels (théories, lois). Relations entre entités en évolution (par ex. : hypothèses, exigences, architecture, conception, code, jeux d'essais). Systèmes existants.

04-Log-B8 Systèmes distribués

Caractéristiques des systèmes distribués. Systèmes réseautés et systèmes centralisés. Concepts et mécanismes fondamentaux. Concepts architecturaux – distribution d'une application sur plusieurs plates-formes. Aperçu des configurations et topologies réseau. Systèmes client-serveur. Synchronisation des processus et communications inter-processus. Principes de tolérance aux fautes. Techniques de traitement transactionnel. Systèmes de fichiers distribués. Systèmes d'exploitation pour architectures distribuées. Sécurité.

04-Log-B9 Traitement parallèle

Modèles de calcul parallèle. Architecture superscalaire. Machines parallèles à mémoire partagée. Réseaux d'interconnexion et leurs propriétés topologiques. Ordinateurs massivement parallèles. Architectures hypercubiques. Mesure de performance des algorithmes parallèles. Évaluation parallèle d'expressions. Structures de recherche et de données parallèles. Traitement parallèle algébrique et géométrique.

04-Log-B10 Réseautique et communications

Communications de données, comprenant : signaux, modulation et réception. Protocoles de transmission de données, comprenant : semi-duplex/duplex intégral, asynchrone/synchrone, de point à point/multipoint, et orienté caractère/bit. Codes de détection et de correction d'erreurs. Jeux de caractères, options de commutation, à savoir le circuit, le message et le paquet. Modèle OSI, y compris les protocoles des couches physique, liaison de données et réseau (RS 232C, RS 449-422/23, HDLC, X.25). Protocoles de transport. Couche session et RNIS. Couches supérieures. Introduction aux structures et conceptions de réseau. Techniques de commutation. Méthodes d'accès aux ressources et de partage des ressources (accès aléatoire, interrogation, concentration et multiplexage). Réseaux de données publics. Réseaux locaux, comprenant : topologies en bus, anneau et arborescente, protocoles et matériel. Réseaux métropolitains et étendus. Éléments des systèmes de mise en file d'attente. Mesures de performance d'un réseau (longueur de la file d'attente, retard et débit). Normes et processus de normalisation.

04-Log-B11 Systèmes de commande de processus

Modèles à temps continu et à temps discret de phénomènes physiques continus. Transformée de Laplace, transformée Z, fonctions de transfert. Réponse, dans le domaine temps et dans le domaine fréquence, des systèmes de premier et de second ordres et d'ordres supérieurs. Stabilité et compensation de la rétroaction. Erreur en régime permanent et commande proportionnelle, intégrale et dérivée (PID). Conception de compensateurs par l'analyse du lieu des racines, critère de Nyquist et conception dans le domaine fréquence. Théorème d'échantillonnage, crénelage et implémentation numérique de contrôleurs continus. Pré-filtrage anticrénelage. Implémentation logicielle de compensateurs numériques. Interfaçage de contrôle informatique.

04-Log-B12 Calcul scientifique

Représentation des nombres et arrondissement des nombres à virgule flottante. Polynômes : représentation, division. Zéros de fonctions : bisection, méthode Newton-Raphson. Algèbre linéaire numérique : décomposition LU et élimination gaussienne, conditionnement, techniques itératives, systèmes structurés, valeurs propres. Différenciation numérique. Quadrature : règle de Simpson, quadrature gaussienne. Interpolation : Lagrange, splines. Lissage de données : méthode des plus petits carrés. Équations différentielles ordinaires : problèmes de valeur initiale, prédicteur-correcteur, Runge-Kutta, problèmes de valeur limite, systèmes d'équations différentielles ordinaires. Équations différentielles partielles simples : problèmes paraboliques, hyperboliques et elliptiques.

04-Log-B13 Analyse et simulation de la performance

Techniques de base d'évaluation de la performance des systèmes, plus spécifiquement : modélisation de la performance, simulation d'événements discrets, vérification et validation des modèles de simulation, analyse des sorties de la simulation, analyse de file d'attente simple à un

serveur et de réseaux de files d'attente, modélisation de systèmes et de réseaux informatiques, et d'autres systèmes de mise en file d'attente ou de non mise en file d'attente.

04-Log-B14 Systèmes critiques pour la sûreté

Analyse de la sûreté et des dangers/risques. Utilisation de logiciels dans les systèmes liés à la sûreté. Aspects juridiques et éthiques. FMEA, HAZOP, FTA, ETA. Évaluation des risques – fréquence et conséquences des événements dangereux. Niveau d'intégrité de la sûreté et cas types. Fiabilité des logiciels. Distinction entre sûreté et fiabilité des systèmes. Fiabilité des logiciels par la prévention des défaillances et la tolérance aux fautes. Aspects de la conception logicielle visant la sûreté et la tolérance aux fautes, y compris l'interface homme-machine. Choix du langage de programmation et de sous-ensembles sûrs. Méthodes formelles, algébrique, spécification basée sur les modèles et processus, langages de spécification formels, preuves de raffinement, preuves de vérification. Tolérance aux fautes, redondance et défaillances de mode commun, programmation N-versions et blocs récupérateurs. Normes de sûreté. Certification et cas types.

04-Log-B15 Intelligence artificielle/Systèmes intelligents

Intelligence artificielle, définition et applications. Concepts d'intelligence artificielle. Calcul des prédicats et son utilisation en intelligence artificielle. Aperçu des systèmes experts et à base de connaissances. Systèmes de production. Représentation des connaissances. Méthodes de recherche; planification. Traitement en langage naturel. Langages de programmation (LISP et Prolog) pour l'implémentation de l'IA et de systèmes experts. Représentation des connaissances. Systèmes à base de règles et systèmes à base d'objets. Vision.

04-Log-B16 Compilateurs

Caractéristiques fondamentales des langages de programmation, fonctions des compilateurs. Processeurs, préprocesseurs, éditeurs de liens et traducteurs. Structure des compilateurs, y compris des tables de compilation et d'exécution. Analyse lexicale et syntaxique. Génération de code intermédiaire. Génération de code objet. Diagnostic des erreurs. Optimisation du code.

04-Log-B17 Paradigme des langages de programmation

Principes des méthodologies de programmation et évolution des caractéristiques des langages de programmation (par ex. : modules, objets, héritage, polymorphisme, exceptions, modèles) soutenant ces méthodologies. Examen des principaux paradigmes des langages de programmation, y compris : procédural, logique, fonctionnel et orienté-objet. Connaissance de la programmation dans divers langages.

04-Log-B18 Infographie /imagerie/visualisation

Matériel, architecture et dispositifs graphiques. Algorithmes et systèmes vectoriels et à balayage de trames. Techniques de description et de génération d'images : approches par point, vecteur et trame. Transformation de l'image, comprenant : mise à l'échelle, translation, rotation, clippage et fenêtrage. Logiciels graphiques et structures de données. Transformations géométriques. Visualisation en trois dimensions. Courbes et splines. Génération d'images à l'aide de polyèdres solides. Éclairage et couleur. Lancer de rayon. Compilateurs graphiques. Structures de données graphiques. Graphiques interactifs. Normes graphiques, p. ex. : GKS, PHIGS, TIGA et X-WINDOWS. Technologie de réalité virtuelle.

Acquisition d'images. Techniques de seuillage, de détection des contours. Analyseurs de profils. Filtrage d'images. Morphologie mathématique. Segmentation d'images. Techniques de reconnaissance de formes. Techniques d'appariement. Reconstruction d'objets 3D. Compression d'images et normes connexes.