Programme d'examens de génie logiciel

Programme d'examens de génie logiciel

Groupe A - Examens obligatoires (huit exigés)

19-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.

Manuels de référence (l'édition la plus récente est recommandée):

- Adam Drozdek, Data Structures and Algorithms in C++, Fourth Edition, Cengage Learning, 2013.

19-Log-A2 Architecture informatique et systèmes d'exploitation

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; machines virtuelles.

Manuels de référence (l'édition la plus récente est recommandée):

- David A. Patterson, John L. Hennessy, Morgan Kaufman, Computer Organization & Design, 1998.
- David A. Patterson, John L. Hennessy, Computer Organization and Design MIPS Edition: The Hardware/Software Interface, 2013 edition.
- Saltzer and Kaashoek, Principles of Computer System Design: An Introduction.

19-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. Typage statique et dynamique. Types mutables et immuables. Modularité et décomposition. Conception orientée-fonction. Conception orientée-objet. Sous-typage. Éléments. Conception d'interfaces. Conception de niveau module. Notations : UML et autres notations. Concepts de base des schémas de conception. Introduction aux tests : tests unitaires, tests boîte noire c. tests boîte grise, couverture des tests.

Manuels de référence (l'édition la plus récente est recommandée):

- Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, Second Edition, Pearson 2003, ISBN 0-13-305699-6.
- Shari Lawrence Pfleeger, Joanne M. Atlee, Software Engineering, Third Edition, Pearson 2006, ISBN 0-13-146913-4.
- Robert C. Martin, Agile Software Development, Principles, Patterns, and Practices, Prentice Hall, 2002.

19-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 personne-machine). Fiabilité et tolérance aux failles. Exceptions et traitement des exceptions. Accès simultané, synchronisation, communication et 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.

Manuels de référence (l'édition la plus récente est recommandée):

- A. Burns, Andy Wellings, Real-Time Systems and Programming Languages: Ada, Real-Time Java and C/Real-Time POSIX Addison–Wesely, 2009 ISBN-13: 978-0321417459. Giorgio Buttazzo, Hard Real-Time Computing Systems, Springer ISBN 978-1-4614-0676-1

19-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 (par ex. : cas d'utilisation) et langages de spécification. Validation, examens et inspections, prototypage, validation des exigences non fonctionnelles. Conception d'essais de réception.

Manuels de référence (l'édition la plus récente est recommandée):

- Shari Lawrence Pfleeger, Joanne M. Atlee, Software Engineering, Third Edition, Pearson 2006, ISBN 0-13-146913-4.
- Karl E. Wiegers, Software Requirements, Microsoft Press, 1999.

19-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.

Manuels de référence (l'édition la plus récente est recommandée):

- Daniel Galin, Software Quality Assurance: From Theory to Implementation, Pearson, 2003, ISBN-13: 978-0201709452.

19-Log-A7 Processus de développement logiciel

Cycles de vie des logiciels. Modèles de processus logiciels. 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. Gestion de la configuration. Gestion des défauts.

Manuels de référence (l'édition la plus récente est recommandée):

- Shari Lawrence Pfleeger, Joanne M. Atlee, Software Engineering, Third Edition, Pearson 2006, ISBN 0-13-146913-4.
- Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, Second Edition, Pearson 2003, ISBN 0-13-305699-6.

21-Log-A8 Mathématiques discrètes

Logique : équivalences propositionnelles, prédicats et quantificateurs, séries, opérations en série, fonctions, suites et cumuls, croissance des fonctions. Algorithmes : complexité des algorithmes, entiers relatifs et division, matrices. Méthodes de preuve : raisonnement par récurrence, définition récursive. Concepts de base du dénombrement : principe des boîtes, permutations et combinaisons, probabilité discrète. Relations de récurrence : inclusion-exclusion. Relations et leurs propriétés : relations de représentation, relations d'équivalence. Introduction aux graphes : terminologie des graphes, représentation des graphes et isomorphisme de graphe, connectivité, trajets d'Euler et de Hamilton. Introduction au tri.

Manuels de référence (l'édition la plus récente est recommandée):

- Pas de manuels référencés

Groupe B - Examens facultatifs (trois exigés)

19-Log-B1 Conception logicielle avancée

Paradigmes de conception logicielle : orienté-objet, orienté-service, à base de composants, en mode agent, programmation fonctionnelle, client-serveur (y compris des protocoles comme REST), virtualisation. Cadres et systèmes basés sur des composants distribués. Modèles de conception. Conception logicielle guidée par modèle. Architecture logicielle. Représentation de l'architecture.

Manuels de référence (l'édition la plus récente est recommandée):

- Bass, L., Clements, P., & Kazman, R., Software Architecture in Practice, Third edition, 2012, Reading, MA: Addison-Wesley.
- Gorton, I., Essential Software Architecture, Second Edition, 2006, Berlin: Springer.
- Bernd Bruegge, Allen H. Dutoit, Object-Oriented Software Engineering, Second Edition, Pearson 2004.
- Joshua Bloch, Effective java, Addison-Wesley, ISBN-13: 978-0134685991.

19-Log-B2 Interface utilisateur

Principes psychologiques de l'interaction personne-machine. Évaluation des interfaces utilisateurs. 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.

Manuels de référence (l'édition la plus récente est recommandée):

- J. Tidwell, O'Reilly, Designing Interfaces: Patterns for Effective Interaction Design, ISBN-13: 978-1449379704.
- J. Johnson, Morgan Kauffman, Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines, ISBN-13: 978-0124079144.

19-Log-B3 Sécurité

Risques, menaces et vulnérabilités de sécurité. Confidentialité, intégrité et respect de la vie privée. Cryptographie, contrôle de l'accès, assurance, imputabilité. Ingénierie de systèmes sécurisés, approches architecturales (par ex. : confinement, machines virtuelles, informatique sécurisée/de confiance). Techniques d'analyse (par ex. : analyse et tests statiques, exploration de modèle. Répercussions sur la conception et la convivialité de l'interface utilisateur.

Manuels de référence (l'édition la plus récente est recommandée):

- R. J. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, Second Edition, Wiley, 2008.

19-Log-B4 Systèmes fiables

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. Fiabilité matérielle et fiabilité logicielle. Techniques de prédiction des défaillances restantes, comprenant : injection de fautes, analyse de l'arbre de classification, couverture du code. Techniques 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 de processus; stabilisation des exigences, de la conception, du code et des artéfacts d'essais. Détection des défaillances actives et passives. Programmation N-versions, points de reprise avant et arrière, blocs récupérateurs et techniques d'arbitrage. Traitement et correction des fautes, exceptions, tolérance aux fautes. Survivance, fonctions essentielles et fonctionnements en mode dégradé. Protection de l'intégrité des données.

Manuels de référence (l'édition la plus récente est recommandée):

- Nancy G. Leveson, Engineering a Safer World: Systems Thinking Applied to Safety, 2012, MIT Press, ISBN-13: 978-0262016629.

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

Modélisation mathématique des logiciels, comprenant des sujets tels que la logique de programmation, l'algèbre de processus et les spécifications basées sur un modèle, les langages OCL et les spécifications algébriques. Raisonnement mathématique utilisant de tels modèles, dont les preuves d'exactitude du programme. Outils de contrôle statique de l'exactitude du logiciel par rapport à ses spécifications.

Manuels de référence (l'édition la plus récente est recommandée):

- V. S. Alagar and K. Periyasamy, Springer-Verlag, Specification of Software Systems, 1998, ISBN 0-387-98430-5.

19-Log-B6 Gestion de projets logiciels

Cycles de développement logiciel (séquentiel, itératif, en spirale, agile). Gestion des coûts du logiciel : estimation de la taille et des efforts. Gestion des risques. Gestion de la qualité. Gestion des actifs logiciels (gestion de la configuration, logiciels ouverts et enjeux connexes liés au protocole IP). Gouvernance du développement logiciel, en particulier dans les environnements réglementés. Production et déploiement de logiciels.

Manuels de référence (l'édition la plus récente est recommandée):

- Pas de manuels référencés

19-Log-B7 Rétro-ingénierie, maintenance et évolution

Maintenance de logiciels : corrective, perfective et adaptative. Techniques de rétro-ingénierie 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. Dette technique.

Manuels de référence (l'édition la plus récente est recommandée):

- Pas de manuels référencés

19-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. Infonuagique. Sécurité.

Manuels de référence (l'édition la plus récente est recommandée):

- Pas de manuels référencés

19-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.

Manuels de référence (l'édition la plus récente est recommandée):

- Pas de manuels référencés

19-Log-B10 Réseautique et communications

Communications de données, comprenant : signaux, modulation et réception. Modèles de canal et capacité de canal. Codes de détection et de correction d'erreurs. Taux d'erreur binaire. Protocoles de transmission de données, comprenant : semi-duplex/duplex intégral, asynchrone/synchrone, de point à point/multipoint. Jeux de caractères, options de commutation, y compris le circuit et le paquet. Architecture de réseau en couches. Protocoles de liaison de données et de couches de réseau. Protocoles de transport. Réseaux locaux et réseaux étendus. Éléments de la théorie 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.

Manuels de référence (l'édition la plus récente est recommandée):

- B. P. Lathi and Z. Ding, Modern Digital and Analog Communication Systems, Fifth edition, Oxford University Press, 2018.
- B. A. Forouzan, Data Communications and Networking, Fifth edition, McGraw-Hill, 2013.
- A. Leon-Garcia and I. Widjaja, Communication Networks: Fundamental Concepts and Key Architectures, McGraw-Hill, 2004.
- D. Bertsekas and R. Gallager, Data Networks, Second edition, Upper Saddle River, NJ: Prentice Hall, 1991.

19-Log-B11 Systèmes de commande de processus

Modèles à temps discret de phénomènes physiques continus. Transformée Z et 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 utilisant le critère de Nyquist et conception dans le domaine fréquence. Théorème d'échantillonnage, crénelage et filtrage anticrénelage. Conception de contrôleurs numériques. Implémentation logicielle de contrôleurs numériques. Interfaçage de contrôle informatique.

Manuels de référence (l'édition la plus récente est recommandée):

- Pas de manuels référencés

19-Log-B12 Calcul technique : Numération

Représentation des nombres et arrondissement des nombres à virgule flottante. Restrictions des calculs à

virgule flottante. Systèmes linéaires : méthodes directe et itérative, conditionnement, systèmes structurés. Zéros de fonctions. Quadrature. Méthodes d'ajustement de données. Équations différentielles ordinaires : problèmes de valeur initiale, prédicteur-correcteur, problèmes de valeur limite, systèmes d'équations différentielles ordinaires. Équations différentielles partielles simples. Optimisation continue.

Manuels de référence (l'édition la plus récente est recommandée):

- Laurene V. Fausett, Numerical Methods, Pearson, 2003.
- Singiresu S. Rao, Applied Numerical Methods for Engineers and Scientists, Prentice Hall, 2002.

19-Log-B13 Analyse et simulation de la performance

Techniques de base d'évaluation de la performance des systèmes, plus spécifiquement : méthodes et outils de mesure, conception et analyse expérimentales, modélisation (comprenant les systèmes de mise en file d'attente et les réseaux de systèmes de mise en file d'attente), simulation d'événements discrets, vérification et validation des modèles de simulation, analyse des sorties de la simulation, méthodes statistiques (comparaison de systèmes à l'aide de données-échantillons, test d'hypothèse et mesures de confiance).

Manuels de référence (l'édition la plus récente est recommandée):

- Mor Harchol-Balter, Performance Modeling and Design of Computer Systems: Queueing Theory in Action, Cambridge University Press, 2013, ISBN: 9781107027503.

19-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. Techniques d'analyse des risques : FMEA, HAZOP, FTA, ETA. Niveau d'intégrité de la sûreté et cas types, utilisation de la notation de structuration des buts (GSN). 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. Facteurs humains intervenant dans la conception visant la sûreté. Choix du langage de programmation et de sous-ensembles sûrs. Méthodes formelles, algébriques, spécification basée sur les modèles et processus, langages de spécification formels, preuves de raffinement, preuves de vérification. Techniques STAMP/STPA (Systems-Theoretic Accident Model and Processes/Systems-Theoretic Process Analysis). Tolérance aux fautes, redondance et défaillances de mode commun, programmation N-versions et blocs récupérateurs. Normes de sûreté. Certification.

Manuels de référence (l'édition la plus récente est recommandée):

- Nancy Leveson, Engineering a safer world, MIT Press, 2012.
- Clifton Ericsson, Software safety primer, CreateSpace, 2013.

19-Log-B15 Intelligence artificielle

Intelligence artificielle, définition et applications. Résolution de problèmes : recherche, recherche adversariale et résolution de contraintes. Connaissance et raisonnement : agents, logique, planification, représentation des connaissances. Incertitude : calcul et raisonnement probabilistes, problèmes de décision. Apprentissage : à partir d'exemples, modèles d'apprentissage, apprentissage par renforcement, réseaux neuronaux, apprentissage profond. Communication : traitement du langage naturel, perception, robotique.

Manuels de référence (l'édition la plus récente est recommandée):

- Peter Norvig and Stuart J. Russell, Artificial Intelligence: A Modern Approach, Pearson, ISBN: 0-13-604259-7.

19-Log-B16 Langages de programmation, sémantique et implémentation

Paradigmes de programmation (procédural, orienté-objet, logique et fonctionnel). Éléments de structuration

(modules, objets, héritage, polymorphisme). Exemples explicites d'une variété de langages. Syntaxe abstraite. Systèmes de types. Interprétation, compilation, génération de code, transformation de code, analyse de code. Structure et éléments des compilateurs. Soutien de l'exécution. Optimisation du code.

Manuels de référence (l'édition la plus récente est recommandée):

- Benjamin C. Pierce, Types and Programming Languages, MIT Press, ISBN 0-262-16209-1.
- Daniel P. Friedman and Mitch Wand, Essentials of Programming Languages, MIT Press Third Edition, ISBN-13: 978-0262062794.
- Robert Harper, Practical Foundations for Programming Languages, Cambridge University Press, Second Edition, ISBN: 9781107150300.
- Keith Cooper and Linda Torczon, Morgan Kaufmann, Engineering: A Compiler, ISBN-13: 978-0120884780.

19-Log-B17 Visualisation de données

Abstraction de données. Abstractions de tâches. Analyse et exploration de données. Découverte de formes. Système de visualisation humain, perception. Présentations visuelles, conception visuelle. Types de diagrammes. Cartes et réseaux. Outils de visualisation de données.

Manuels de référence (l'édition la plus récente est recommandée):

- Tamara Munzner, Visualization Analysis and Design, CRC Press, 2014, ISBN-13: 978-1466508910.