Organisme de formation
Ch. des Fleurettes 28
Lausanne, Suisse
Tél. 021 626 08 00
www.cyberiade.ch

Formation continue

Coding Intelligence (EN)

Lire, écrire et corriger du code avec méthode



Résumé

Préparation au test Tosa Coding Intelligence. Le cours est en français, les questions du test restent en anglais. On part de la logique d'un programme, puis on le découpe en blocs réutilisables, on le teste, on le corrige à plusieurs et on juge ce que renvoie un outil d'IA. Le score, de 1 à 1 000, situe le niveau d'Initial à Expert.

Objectifs

Lire un programme et prévoir son résultat, y compris une fonction qui s'appelle elle-même.
Écrire une logique simple : valeurs, conditions, boucles, fonctions et structures.
Tester, trouver l'origine d'une erreur et la corriger pour qu'elle ne revienne pas.
Travailler à plusieurs : versions, revue de code et étapes d'un projet.
Juger un code produit par un outil d'IA, et l'effet du programme sur les personnes.

Public cible

Étudiants et adultes qui utilisent l'intelligence artificielle pour produire ou relire du code. Le test se passe en anglais.

C'est une certification complémentaire à une formation de type CFC ou à une formation continue, pour étayer votre CV et mettre vos connaissances à jour.

Contenu

1. Situer le test

- Voir ce que mesure Coding Intelligence : les bases de la programmation, sans langage imposé
- Savoir que les questions sont en anglais
- Repérer le format : 60 minutes, 35 questions, score de 1 à 1 000, certificat à partir de 551
- Distinguer les cinq niveaux, de Beginner à Expert

2. Comprendre un programme

- Expliquer comment une machine suit des instructions, et à quoi sert le code
- Distinguer le matériel du logiciel
- Distinguer un code compilé d'un code interprété, et reconnaître un langage de haut niveau
- Situer un langage d'interface, pour ce que la personne voit à l'écran
- Relier un programme à la mémoire, et une donnée qui reste après l'arrêt à un fichier
- Nommer les pièces d'un système de fichiers
- Reconnaître une façon d'organiser un programme, et classer les langages

3. Lire et prévoir

- Suivre la logique d'un court programme et annoncer le résultat
- Repérer un motif qui revient
- Trouver une erreur de logique dans une condition
- Découper un problème, et expliquer ce qui fait un algorithme
- Repérer un appel de fonction dont les arguments ne conviennent pas

Organisme de formation
Ch. des Fleurettes 28
Lausanne, Suisse
Tél. 021 626 08 00
www.cyberiade.ch

- Expliquer ce qu'est l'abstraction, dans une interface ou un plan de logiciel
- Suivre une fonction qui s'appelle elle-même
- Choisir le pseudo-code qui trouve le deuxième plus grand nombre
- Choisir le parcours le plus économe pour trouver un maximum

4. Écrire et corriger une séquence

- Expliquer à quoi sert un commentaire
- Repérer ce qui réagit à une action de la personne, et lire une saisie
- Prévoir l'ordre d'une boucle, et écrire une fonction simple
- Corriger un résultat faux
- Choisir le bon retrait pour qu'une condition s'exécute
- Compléter des boucles imbriquées
- Repérer un passage qu'on peut alléger en séparant les rôles

5. Découper en fonctions

- Expliquer pourquoi on réutilise un bloc, et à quoi sert un guide de style
- Nommer une variable pour qu'elle se lise
- Passer une valeur à une fonction, et distinguer le paramètre de l'argument
- Se servir d'une bibliothèque
- Réduire la copie du même code, et regrouper ce qui va ensemble
- Suivre une suite d'appels et de boucles jusqu'au résultat

6. Ranger les données

- Choisir un type, une constante ou un booléen, et convertir une valeur
- Lire une liste et ce qu'elle contient
- Choisir une structure clé-valeur pour une recherche
- Prévoir une valeur qui change pendant que le programme tourne
- Choisir la structure d'un retour en arrière, comme le bouton précédent d'un navigateur
- Reconnaître une pile, dernier entré, premier sorti
- Transformer une pile en file
- Distinguer un arbre d'un graphe, et comparer le coût d'une opération

7. Tester et améliorer

- Distinguer un test qui vérifie le comportement d'un test qui vérifie autre chose
- Situer les niveaux de test
- Classer les erreurs, et expliquer à quoi sert un débogueur
- Lire une erreur quand les clés d'une structure ne se correspondent pas
- Trouver une erreur de calcul, une borne, une faute de nom ou une boucle mal formée
- Repérer un décalage d'un cran dans les bornes d'un tableau
- Suivre la profondeur d'une fonction qui s'appelle elle-même
- Voir une récursion qui ne s'arrête pas, faute de cas d'arrêt
- Comparer deux façons d'ôter les doublons, et expliquer ce qui coûte le plus

8. Travailler à plusieurs

- Expliquer à quoi servent un dépôt, une revue de code et une demande d'intégration
- Voir comment une branche se sépare, puis se réunit
- Réduire les conflits au moment de réunir le travail
- Poser le périmètre d'un projet, et ordonner les étapes de sa vie
- Repérer les dépendances pour ordonner les tâches

Organisme de formation
Ch. des Fleurettes 28
Lausanne, Suisse
Tél. 021 626 08 00
www.cyberiade.ch

- Prioriser une revue selon le risque, et réagir à une faille avant une mise en ligne

9. Coder avec une IA

- Expliquer ce qu'un outil d'IA sait faire, et ce qu'il ne remplace pas
- Apparier une tâche et une demande
- Lire les limites d'un code produit par l'outil
- Rester responsable du code que l'on garde
- Reconnaître une réponse inventée, et classer les demandes selon leur précision

10. Agir dans le numérique

- Situer le modèle client-serveur, le rôle de TCP/IP et celui du chiffrement
- Distinguer un code ouvert, le domaine public, une licence et le droit d'auteur
- Attribuer un code qui n'est pas le nôtre
- Voir ce qu'une fuite de données coûte, et comment y répondre
- Repérer une interface trompeuse, et un obstacle pour qui distingue mal les couleurs
- Relier la consommation d'un centre de données à l'effet du numérique
- Vérifier un contenu produit par une IA avant de le tenir pour vrai
- Respecter le consentement, y compris quand les données touchent la santé

11. Coder de façon responsable

- Valider une saisie, et refuser un mot de passe stocké en clair
- Repérer un biais dans un algorithme, et dans des données utilisées pour recruter
- Relire un incident pour que la même panne ne revienne pas
- Expliquer où un algorithme pèse sur l'information, l'emploi et les décisions
- Rester responsable du code qu'un outil d'IA a proposé

Tarifs

Cours individuel selon calendrier	**CHF 400.–/jour**
Cours en entreprise	**Sur demande**
Coaching privé	**CHF 100.–/heure**

La durée à prévoir se discute en fonction de votre niveau initial et du niveau attendu (score).

Certification	**CHF 65.–**
Certification + 1 évaluation	**CHF 80.–**
Certification + 2 évaluations	**CHF 92.–**

Lieu

Chemin des Fleurettes 36, 1007 Lausanne
En ligne

Modalités

Cours en ligne
Cours en présentiel