Bing

Bonjour, je suis Bing, un chatbot qui peut vous aider à trouver des informations sur différents sujets. Vous m'avez demandé de vous proposer un exemple de contenu de formation sur le concept de "programmation structurée" en conception algorithmique, basé sur la méthode : quoi qui où quand comment combien pourquoi. Voici ce que j'ai trouvé :

* Quoi : La programmation structurée est une méthode de programmation impérative qui vise à organiser le code en utilisant des structures de contrôle simples et des sous-programmes modulaires . Elle permet d'écrire des programmes plus lisibles, plus fiables et plus faciles à maintenir .
* Qui : La programmation structurée a été initiée par des informaticiens comme Nicklaus Wirth, Edsger Dijkstra et Jean-Dominique Warnier dans les années 1960 et 1970 . Elle s'applique à tous les programmeurs qui utilisent des langages procéduraux comme Pascal, C, Ada ou Java .
* Où : La programmation structurée peut être utilisée dans n'importe quel domaine d'application informatique, que ce soit pour le développement de logiciels, de systèmes embarqués, de jeux vidéo ou de sites web .
* Quand : La programmation structurée est apparue vers 1970 comme une technique populaire pour améliorer la qualité du code . Elle est toujours d'actualité aujourd'hui, même si elle a été complétée par d'autres paradigmes comme la programmation orientée objet ou la programmation fonctionnelle .
* Comment : La programmation structurée repose sur trois principes fondamentaux :
  + L'utilisation de structures de contrôle séquentielles (séquence), conditionnelles (alternative) ou répétitives (itération) pour définir le flux d'exécution du programme.
  + La décomposition du programme en petits sous-programmes (fonctions ou procédures) qui réalisent une tâche spécifique et qui communiquent par des paramètres.
  + L'évitement des variables globales et des instructions goto, qui rendent le code moins clair et plus difficile à déboguer.
* L'utilisation de structures de contrôle séquentielles (séquence), conditionnelles (alternative) ou répétitives (itération) pour définir le flux d'exécution du programme.
* La décomposition du programme en petits sous-programmes (fonctions ou procédures) qui réalisent une tâche spécifique et qui communiquent par des paramètres.
* L'évitement des variables globales et des instructions goto, qui rendent le code moins clair et plus difficile à déboguer.
* Combien : Il n'existe pas de règle universelle pour mesurer la complexité d'un programme structuré, mais on peut utiliser des critères comme le nombre de lignes de code, le nombre de sous-programmes, le nombre de paramètres ou le nombre de niveaux d'imbrication . En général, on recommande de limiter la taille des sous-programmes à une soixantaine de lignes et le nombre de paramètres à une dizaine .
* Pourquoi : La programmation structurée présente plusieurs avantages par rapport à une programmation non structurée :
  + Elle facilite la compréhension du code, car il suit une logique simple et cohérente.
  + Elle favorise la fiabilité du code, car il est plus facile de détecter et de corriger les erreurs.
  + Elle simplifie la maintenance du code, car il est plus facile de modifier ou d'ajouter des fonctionnalités.
  + Elle permet la réutilisation du code, car les sous-programmes peuvent être appelés depuis différents endroits du programme ou d'autres programmes.
* Elle facilite la compréhension du code, car il suit une logique simple et cohérente.
* Elle favorise la fiabilité du code, car il est plus facile de détecter et de corriger les erreurs.
* Elle simplifie la maintenance du code, car il est plus facile de modifier ou d'ajouter des fonctionnalités.
* Elle permet la réutilisation du code, car les sous-programmes peuvent être appelés depuis différents endroits du programme ou d'autres programmes.

J'espère que cet exemple vous a été utile. Si vous avez d'autres questions sur la programmation structurée ou sur un autre sujet, n'hésitez pas à me les poser. 😊