

# Programmation structurée

---

La **programmation structurée** constitue un sous-ensemble de la programmation impérative. C'est un paradigme important de la programmation, apparu vers 1970. Elle dérive de travaux de Nicklaus Wirth pour son Algol W et reçut son coup d'envoi avec l'article fondateur de Dijkstra dans *Communications of the ACM* nommé *GO TO statement considered harmful* (*L'instruction GOTO est considérée nocive*)<sup>1, 2</sup>.

Elle est en effet célèbre pour son essai de suppression de l'instruction *goto* ou du moins pour la limitation de son usage à des cas inhabituels et graves (que l'on nommerait plutôt des exceptions).

## Sommaire

---

**Usage**

**Méthodologie associée**

**Antonyme**

**Références**

## Usage

---

La programmation structurée est possible dans n'importe quel langage de programmation procédural, mais certains comme le FORTRAN IV s'y prêtaient très mal. Vers 1970, la programmation structurée devint une technique populaire, et les langages de programmation procéduraux intégrèrent à l'image d'Algol et en le dépassant des mécanismes rendant aisée la programmation structurée (faisant tomber en désuétude des langages parfois novateurs, mais qui facilitaient une programmation dé-structurée). Parmi les langages de programmation les plus structurants, on trouve PL/I, Pascal et, plus tardivement pour les projets de très grande taille, Ada.

Pour l'écriture de fragments assez courts, la programmation structurée recommande une organisation hiérarchique simple du code. On peut le faire dans la plupart des langages de programmation modernes par l'utilisation de structures de contrôles *while*, *repeat*, *for*, *if .. then .. else*. Il est également recommandé de n'avoir qu'un point d'entrée pour chaque boucle (et un point de sortie unique dans la programmation structurée originelle), et quelques langages l'imposent. Cette technique est toutefois peu suivie, car elle encourage l'utilisation de plus nombreuses variables pour remplacer les sorties immédiates par des états ; or la multiplication des variables augmente la charge cognitive pour comprendre un fragment de code.

Les programmeurs décomposent leur code en modules (appelés fonctions et procédures dans certains langages) ne dépassant guère 60 lignes, afin d'être présente en entier sous les yeux. On recommande aux programmes d'éviter l'usage des variables globales afin de prévenir les effets de bord (*side effects*) : les sous-programmes utilisent donc des variables locales et peuvent agir sur des arguments fournis explicitement en paramètre, par valeur (leur contenu) ou par référence (l'adresse de ce contenu). Ces techniques aident à créer des petits morceaux de code, faciles à comprendre isolément et sans nécessité d'avoir à en connaître le contexte.

## Méthodologie associée

---

La programmation structurée est souvent utilisée conjointement à la méthodologie de développement par décompositions successives (*top-down design*), mais ne se confond pas avec elle : Smalltalk, bon langage de conception *top-down*, n'est par exemple guère structuré. Dans cette approche les programmeurs décomposent la structure, à large échelle, d'un programme en termes d'opérations plus petites, codent et testent ces petites opérations, et les assemblent pour réaliser le programme (voir aussi *Cycle en V*). Cela n'est évidemment possible que sur des domaines applicatifs déjà bien connus, sans quoi la décomposition risque dès le départ de ne pas être la bonne, ce qui est très difficile à modifier ensuite.

À la fin du xx<sup>e</sup> siècle la plupart des programmeurs avaient adopté la programmation structurée. Dijkstra rappelait qu'un programme devait d'abord être compris par le programmeur et ses collègues chargés de la maintenance, et que si cette tâche était accomplie, le reste — le faire exécuter à la machine — n'était plus que formalité.

## Antonyme

---

- Programmation spaghetti

## Références

---

1. <https://homepages.cwi.nl/~storm/teaching/reader/Dijkstra68.pdf>
2. <http://www.cs.utexas.edu/users/EWD/ewd02xx/EWD215.PDF>

---

Ce document provient de « [https://fr.wikipedia.org/w/index.php?title=Programmation\\_structurée&oldid=149702860](https://fr.wikipedia.org/w/index.php?title=Programmation_structurée&oldid=149702860) ».

**La dernière modification de cette page a été faite le 20 juin 2018 à 20:31.**

Droit d'auteur : les textes sont disponibles sous licence Creative Commons attribution, partage dans les mêmes conditions ; d'autres conditions peuvent s'appliquer. Voyez les conditions d'utilisation pour plus de détails, ainsi que les crédits graphiques. En cas de réutilisation des textes de cette page, voyez comment citer les auteurs et mentionner la licence.

Wikipedia® est une marque déposée de la Wikimedia Foundation, Inc., organisation de bienfaisance régie par le paragraphe 501(c)(3) du code fiscal des États-Unis.