

Objectif

Ce cours présente les concepts fondamentaux des algorithmes et de la programmation.

A l'issue de ce cours, l'apprenant sera en mesure de concevoir des algorithmes et des programmes d'une manière structurée et bien pensée.

Algorithme

Un algorithme est une séquence finie d'instructions qui permet d'effectuer une tâche en vue de résoudre un problème.

Concepts

Séquence, condition, répétition, sous-programme (procédure), variables.

Computational Thinking

Une manière de penser pour formuler un problème et exprimer sa solution sous forme d'un algorithme.

Éléments du Computational Thinking

- *Décomposition* : comprendre que la résolution d'un problème complexe peut impliquer de le décomposer en un ensemble de problèmes plus simples.
- *Reconnaissance de formes* : savoir qu'un nouveau problème est susceptible d'être lié à d'autres problèmes que l'apprenant a déjà résolus.
- *Abstraction* : voir un problème et sa solution à plusieurs niveaux de détail. Simplifier un problème en considérant uniquement les détails qui comptent.
- *Généralisation* : se rendre compte qu'une solution à un problème peut être apportée de manière à résoudre une gamme de problèmes similaires.
- *Algorithmes* : réfléchir aux tâches à accomplir sous forme d'une séquence d'étapes logiques.

Pédagogie

Explorer : enquêter, essayer des choses par soi-même, déboguer en réaction aux feed-backs.

Envisager : avoir un objectif en tête, prédire le résultat du programme avant d'essayer.

Expliquer : expliquer ce qu'on fait, articuler les raisons de son approche.

Échanger : collaborer et partager, essayer de voir un problème du point de vue d'un autre, défendre sa propre approche et se comparer avec les autres.

Contenu

Algorithmes et programmation

Types de données

Opérations de base

Premiers programmes

Structure conditionnelle

Structure itérative

Procédures

Fonctions

Le type énuméré et le type intervalle

Le type set

Le type array

Le type string

Le type record

Pointeurs

Fichiers