

Algorithmes classiques

La plupart des algorithmes que je cite ci-dessous sont tirés du Cormen [[CLRS10](#)].

Algorithmes sur les graphes :

- PRIM pour trouver l'arbre couvrant minimal
- KRUSKAL pour trouver l'arbre couvrant minimal
- FORD FULKERSON pour trouver le flot maximal ou la coupe minimal
- BELLMAN FORD pour trouver le plus court chemin entre deux sommets
- DIJKSTRA pour trouver le plus court chemin entre deux sommets
- FLOYD WARSHALL pour trouver le plus court chemin entre toute paire de deux sommets
- KOSARAJU pour trouver les composantes fortement connexes
- TARJAN pour trouver les composantes fortement connexes
- TRI TOPOLOGIQUE

Algorithmes de tri :

- TRI BULLE
- TRI PAR SÉLECTION
- TRI PAR INSERTION
- TRI RAPIDE
- TRI FUSION
- TRI PAR TAS
- TRI PAR DÉNOMBREMENT
- TRI PAR BASE
- TRI PAR PAQUETS
- TRI EXTERNE
- TRI TOPOLOGIQUE

Algorithmes du texte :

- MORRIS PRATT pour trouver les motifs d'un texte
- KNUTH MORRIS PRATT pour trouver les motifs d'un texte
- COCKE YOUNGER KASAMI pour montrer qu'un mot appartient à une grammaire algébrique
- ANALYSE LL(1) pour faire de l'analyse syntaxique
- PLUS LONGUE SOUS SÉQUENCE COMMUNE
- DISTANCE D'ÉDITION

Algorithme sur des automates :

- HOPCROFT pour trouver l'automate minimal

Algorithme pour la logique :

- ALGORITHME D'UNIFICATION pour trouver l'unificateur principal de deux termes de logique du premier ordre

Une structure de dictionnaire est une structure de données avec un type abstrait demandant les opérations de RECHERCHE, d'INSERTION et de SUPPRESSION.

Structure de **dictionnaire** :

- LISTE
- TABLEAU TRIÉ
- ARBRE BINAIRE DE RECHERCHE
- ARBRE BINAIRE DE RECHERCHE OPTIMAL
- AVL
- ARBRE ROUGE NOIR
- ARBRE SPLAY
- B-ARBRE