

下記に掲載を希望されるアルゴリズムを書き込んでください。

書き込み方法は、次のようになります。

1. ページ上部の [編集] をクリックし、[このページを編集] を選択します。
2. ページの編集画面が表示されるので、先頭に「-」(半角のマイナス)を付けて「- () (はアルゴリズム) を現状のリストに追加する形で入力します。
3. [ページ保存] ボタンをクリックします。

- 自然言語の解析法 (他言語との翻訳、仮名漢字変換、文を文節や品詞に分解、等)
- バブルソート
- バイナリソート
- ベジアンフィルタ
- 文書比較アルゴリズム ([ここ](#)とか)
- [LLL algorithm](#)
- プレゼンハムの線分描画アルゴリズム (様々な高速化や誤差修正方法など)
- 線形計画法の代表的解法
- 典型的なパーサのアルゴリズム
- モンテカルロ法
- マルコフ連鎖モンテカルロ法
- 遺伝的アルゴリズム
- 動的計画法
 - 枝刈りなどのゲームでよく使うアルゴリズム
- グラフの連結度増大アルゴリズム
- PageRank
- suffix array
- B tree (B-tree/B-+tree/B-*tree)
- パトリシア木
- Double Array
- A* (A-star: 経路探索)
- CRF (Conditional Random Fields)
- GA (Genetic Algorithm: 遺伝的アルゴリズム)
- SVM (support vector machine: サポートベクターマシン)
- PKE (公開鍵暗号)
- Non blocking queueing
- [AKS素数判定法](#)
- k-最短路(Dijkstraの応用よりは、できればEppsteinの)
- Viterbi algorithm
- MT(Mersenne Twister)およびMTの5 ~ 128bit程に拡張した場合のアプローチ([ここ](#)とか)
- SFMT(SIMD-oriented Fast Mersenne Twister)
- WELL(Well Equidistributed Long-period Linear)
- Xorshift RNGs
- Bellman-Ford (BF) algorithm
- 平面の三角形分割 (デローニー分割)、空間の四面体分割
- 巡回セールスマン問題のヒューリスティックな解法
- 迷路の作り方
- 最小完全ハッシュ関数の構築や様々な場面でのアプローチ
- 各種予測・補間アルゴリズムの紹介と比較
- フリーセル (Windows標準添付のゲーム) の解法
- 符号のリスト復号アルゴリズム (Guruswami-SudanやGoldreich-Levin等)
- Karatsuba法
- FFT
- ペアリング演算の高速化
- Wait-free キュー (java.util.concurrent.ConcurrentLinkedQueue)
- 修正マージソート modified-merge-sort
- グレブナ基底計算
- 最大流 (Goldberg-Rao)
- 劣モジュラ関数最小化
- 記号積分 (Risch)
- 線型時間のグラフ平面性判定
- 内点法
- persistentなデータ構造のamortized costを評価するときのbanker's methodとphysicist's methodを解説してほしい
- Tomasulo's Algorithm
- Burrows Wheeler's Transform
- 高速ウェーブレット変換
- 誤り有文字列マッチング (本"Flexible Pattern Matching in Strings", G. Navarroでカバーされているもの)
- いろんな人 (SIAM関連) が選んだ20世紀のTOP 10 アルゴリズムで上で述べられていないもの ("<http://amath.colorado>).

[edu/resources/archive/topten.pdf](#)")

- Hindley – Milner type inference algorithm
- グラフ同一性判定
- funnelsort
- sublinear time algorithm
- thorupのアルゴリズム(無向グラフの最短路問題を解く線形時間アルゴリズム)
- Gomory-Hu木
- Coppersmith – Winogradのアルゴリズム(行列乗算)
- フィボナッチヒープ
- [レジスタ割り付けアルゴリズム \(グラフ彩色, linear scan等\)](#)
- Knuth-Morris-Pratt Algorithm
- Levenshtein Distance
- Aho-Corasick algorithm
- 3D CGゲームにおける回転のための四元数の利用
- [フォトンマッピングやPRT等の大域照明アルゴリズム](#)
- [物体の特徴を保った写真の伸縮アルゴリズム](#)
- [漫画の彩色補助アルゴリズム](#)
- [顔認識](#)や[OpenCV](#), [ARToolKit](#)等, いわゆるcomputer visionのエッセンスを分かりやすく
- [ビットマップ画像の輪郭をベクタ画像として取り出すアルゴリズム](#)
- [メッシュ生成アルゴリズム](#)のうち使いやすくメジャーなもの
- [逆運動学 \(inverse kinematics\) アルゴリズム](#)
- グラフを見栄え良く描画するアルゴリズム (参考: [チュートリアル](#)のPDF)
- 縮尺の小さい地図を見栄え良く描画する[総描 \(generalization\) アルゴリズム](#)
- [任意の2ファイルにそれぞれ適当なsuffixを付加してMD5ハッシュが衝突するようにするアルゴリズム](#)
- [Tree automaton](#)に関して重要な知見を簡潔に (できれば[生垣automaton](#)にも一言)
- Diamond-S algorithm
- Byzantine consensus algorithm
- Purely functional data structures (<http://www.cs.cmu.edu/~rwh/theses/okasaki.pdf>)
- PPM圧縮法
- リニアハッシュ
- マトロイドのアルゴリズム
- グリーディ法とグリードイドとの関係