



# アルゴリズムクイック リファレンス

Algorithms in a Nutshell

輪講計画／第1章：アルゴリズムで考える

@rian\_tkb

2018/04/19

# 今日やること

- ▶ 発表者の把握
- ▶ 各回の担当決め
- ▶ 2Q でのコマの調整
- ▶ 第1章：アルゴリズムで考える
  - ▶ 8ページくらいしかないので、デモンストレーション(?)も兼ねて今日やっちゃいます

# アルゴリズムクイックリファレンス 第2版

3

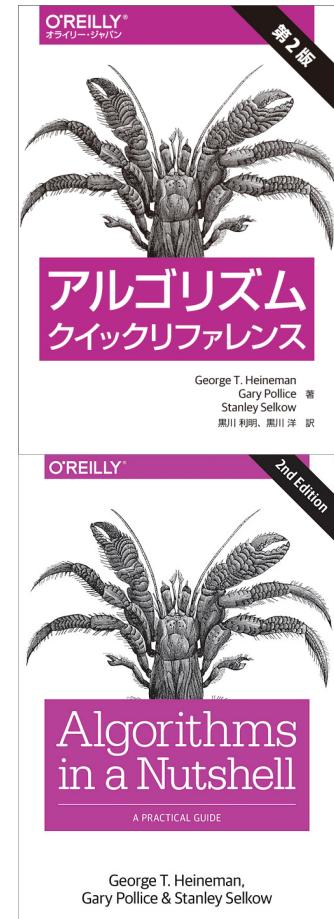
## ▶ 様々なアルゴリズムについて網羅的に解説した本

- ▶ リファレンスとあるが、ちゃんと読み物の体裁になっており、普通に頭から通して読んでいっても問題なさそう
  - ▶ 昨年の螺旋本よりは輪講向きっぽい [要出典]

## ▶ GitHub もある

- ▶ <https://github.com/heineman/algorithms-nutshell-2ed>

## ▶ PDF 版も売られている



# 内容

- ▶ 1章：アルゴリズムで考える (8p)
  - ▶ 導入、内容がない
- ▶ 2章：アルゴリズムの数学 (28p)
  - ▶ 計算量とかの話
- ▶ 3章：アルゴリズムの構成要素 (20p)
  - ▶ ほぼいらない話、最後に  
「一般的なアプローチ」って言って  
分割統治と DP が出てくるので  
そこちゃんとやろう
- ▶ 4章：整列アルゴリズム (44p)
  - ▶ 色々なソート
- ▶ 5章：探索 (48p)
  - ▶ 線形探索、二分探索
  - ▶ 二分探索木を平衡に保つのは  
むずいのでお気持ちだけの予定
- ▶ 6章：グラフアルゴリズム (42p)
  - ▶ DFS, BFS, 最短路問題、最小全域木
- ▶ 7章：AI における経路探索 (56p)
  - ▶ minimax,  $\alpha\beta$ , A\*
- ▶ 8章：ネットワークフロー (32p)
  - ▶ 最大流、二部マッチング、最小費用流
- ▶ 9章：計算幾何学 (40p)
  - ▶ 凸包走査、ボロノイ図
- ▶ 10章：空間木構造 (44p)
- ▶ 11章：新たな分類のアルゴリズム (22p)
- ▶ 12章：結び (10p)

# 進め方

- ▶ 基本的には、「アルゴリズムの**お気持ち**」を理解する
  - ▶ 何が求まるのか
  - ▶ どういう時に使えるのか
  - ▶ どうやって求めていくのか
  - ▶ なんでそれで上手くいくのか
  - ▶ 何が嬉しい、どこがすごいのか
- ▶ 余裕がある人は実装をして（またはコードを拾ってきて）実際に問題を通してみる
- ▶ 適当な問題を見繕ったり、ちょっと補足的な内容を書いたりみたいな短い補足スライドを一応毎回用意する**予定**です
  - ▶ [https://github.com/riantkb/algorithm\\_quick\\_reference](https://github.com/riantkb/algorithm_quick_reference)

# 進め方（草案）

#	日付	担当者	内容
1	4/19		輪講計画／1章
2	4/26		2章
	5/3		祝日
3	5/10		3章
4	5/17		4章
5	5/24		5章
6	5/31		6章
7	6/7		7章 7.1 – 7.5 (ゲーム木、minimax, αβ)
8			7章 7.6 – (探索木、DFS, BFS, A*)
9			8章
10			9章
11			10章

# 進め方

- ▶ 正直 9 章以降はやってもやらなくてもいいかなという感じなので（それよか前半部分のお気持ちをちゃんと理解することの方が重要そう）、各章を無理に 90 分に収めずに次回に持ち越すのも手かなあという気持ちがあります
- ▶ 各回の担当を決めます
  - ▶ 発表者は 6 人いるはず
- ▶ 2Q のコマも決めたい

- ▶ 次のページから第1章の内容ですが、教科書の題材を取り上げずに好き勝手やってるのであまり参考にしないでください

# 1章：アルゴリズムで考える

9

## ▶ この章の内容

- ▶ 問題にあった**アルゴリズム**を上手に選べるようになろう！



- ▶ 実際にはこの章では**凸包**を題材にして選択したアルゴリズムによる計算時間の差、また近似解の導出などをしていますが、凸包なんていきなり言われてもアレだしどうせ9章でも出てくるので、ここでは他の題材でやります

# 例題

- ▶ あなたはある女性と会話をしています
- ▶ よくわからないですが、あなたはその女性の年齢を当てなければならぬことになりました
- ▶ そのために、あなたは以下の 2 つの質問を好きな回数行うことができ、女性はそれに対して “Yes” または “No” で回答します
  - ▶ 「あなたは x 歳以上ですか」
  - ▶ 「あなたは x 歳以下ですか」
- ▶ しかし、女性は歳を聞かれるのを嫌がるので、一回質問をされるたびにどんどん不機嫌になります
- ▶ よって、なるべく少ない回数で女性の年齢を特定するには、どのような**アルゴリズム**で質問をすればよいでしょうか



# 愚直な解法

- ▶ 「x歳以下ですか」と「x歳以上ですか」の両方に“Yes”と答えてきたら、x歳だとわかる
- ▶ よって、
  - 「0歳以下ですか」「0歳以上ですか」
  - 「1歳以下ですか」「1歳以上ですか」
  - 「2歳以下ですか」「2歳以上ですか」
  - 「3歳以下ですか」「3歳以上ですか」
  - ⋮

と聞いていけばよい
- ▶ 女性の年齢が0歳以上120歳以下だとすると、質問回数はだいたい240回くらい

# 問題を理解する

12

## ▶ 少し考察をする

- ▶ 実は、「 $x$ 歳以下ですか」と「 $x + 1$ 歳以上ですか」は得られる情報が全く変わらない
- ▶ つまりどちらか片方の質問だけで年齢を特定できるはず
- ▶ 「0歳以下ですか」「1歳以下ですか」「2歳以下ですか」……と聞いていって、初めて“Yes”になった質問の年齢が答え  
▶ それはそう
- ▶ 質問回数は120回くらい  
▶ 半分になった！

# もっと上手くできる？

13

- ▶ 実はこの問題、質問の回数を 7 回まで減らせる（はず）
  - ▶ 二分探索という**アルゴリズム**を用いる
    - ▶ 二分探索については 5 章で詳しく触れられるはず
- ▶ 計算量が  $O(N)$  から  $O(\log N)$  になったので質問回数が劇的に減った
  - ▶ 計算量については 2 章で詳しく触れられるはず



# 結局アルゴリズムってなんなの？

14

- ▶ 二分探索、みたいになんかカッコいい名前がついているものだけが**アルゴリズム**なわけではない
  - ▶ 最初の愚直な解法も十分アルゴリズムと言える
- ▶ アルゴリズムというのはただの手法にすぎず、問題を理解し、その問題の特徴にあった手法を選ぶ、というのが**アルゴリズムを上手く選ぶ**、ということになる
  - ▶ 二分探索に関しても、 $f(x) := (ans \leq x)$  という関数が単調である、という問題の特徴にあった手法を選んだ結果にすぎない

## 例題②

- ▶ [https://beta.atcoder.jp/contests/arc095/tasks/arc095\\_b](https://beta.atcoder.jp/contests/arc095/tasks/arc095_b)
- ▶  $N$  個の数字の中から 2 つ数字を選んで、 $pCq$  が最大になるようにする
  - ▶ 例えば  $6, 9, 4, 2, 11$  という 5 つの数字から選ぶ場合は、 $11$  と  $6$  を選ぶと最大になる
- ▶ 愚直解
  - ▶  $N$  個の数字の中から 2 つ数字を選ぶ選ぶ方を全通り試して、組合せを全通り計算して、一番大きいものを出力する
  - ▶ 計算量： $O(N^2 M)$  ( $M := N$  個の数字の最大値)

# 問題を理解する

- ▶ 少し考察する
  - ▶  $p$  を固定したとき、 $q$  の値と  $pCq$  の大きさの関係は？
  - ▶ 逆に  $q$  を固定したとき、 $p$  の値と  $pCq$  の大きさの関係は？
- ▶ 実はこの問題は組合せを全く計算せずに、 $O(N)$  で解ける

- ▶ **アルゴリズム**はすごく大事
- ▶ でもそれ以上に、適切なアルゴリズムを選択するための**問題を理解する**ということがとても大事！



- ▶ この輪講は、色々なアルゴリズムを実装できるようになる！というところよりも、色々なアルゴリズムとそれを選択するに至った問題の特徴を知ることで、この処理、もうちょい高速化、または高精度化できそう！という感性を磨く、というのが大きなメリットであり達成すべき目標かなあと思います