

高校生が学ぶアルゴリズム

—情報の教科書から—

山口健二 | お茶の水女子大学

高校生とアルゴリズム

高校の教科「情報」

本稿では、アルゴリズムの基本構造について述べる。筆者は高校の教員でもあるため、高校の教科「情報」の科目についての説明と、主に普通科で使用されている教科書において、アルゴリズムがどのように位置付けられ、どのような説明がされているのかを交えながら解説する。最後に将来のアルゴリズム教育の展望を述べたい。

2020年現在、高校の教科「情報」は、まず大きく2つに分類される。1つは共通教科情報科、もう1つは、専門教科情報科である(表-1)。前者は主に普通科で、後者は主に専門学科で履修する教科である。今回は前者の共通教科情報科の科目を見ていく。

「社会と情報」と「情報の科学」

共通教科情報科では、「社会と情報」、「情報の科学」の2科目が存在し、どちらかを週2単位で選択必修することになっている。たとえば、平成31(2019)年度の東京都立高校については、それぞれの教科書の採択

表-1 情報科の科目

共通教科情報科の科目	専門教科情報科の科目
社会と情報	情報産業と社会
情報の科学	情報の表現と管理
	情報と問題解決
	情報テクノロジー
	課題研究
	アルゴリズムとプログラム
	ネットワークシステム
	データベース
	情報システム実習
	情報メディア
	情報デザイン
	表現メディアの編集と表現
	情報コンテンツ実習

数が、「社会と情報」は156、「情報の科学」は106となっており、東京都立高校の3/5程度は、「社会と情報」の教科書で学んでいることになる¹⁾。

「社会と情報」の目標は、情報機器や情報通信ネットワークなどを適切に活用して情報を収集、処理、表現し、コミュニケーション力を養い、情報社会に積極的に参画する態度を育てることである²⁾。情報を処理・表現する過程で、2進法や文字・音・画像のデジタル化は学ぶが、学習指導要領にアルゴリズムの扱いに関して記載がないため、アルゴリズムに関する内容がない教科書が大半である。したがって、アルゴリズムという言葉を知らずに高校を卒業する生徒も一定数存在することになる。

一方、「情報の科学」の目標は、情報と情報技術を問題の発見と解決に効果的に活用するための科学的な考え方を習得させ、情報社会の発展に主体的に寄与する能力と態度を育てることである。こちらは、「社会と情報」のデジタル化の内容に加えて、学習指導要領にアルゴリズムやモデル化とシミュレーションといった記載があり、流れ図(フローチャート)やプログラミング言語を使って、アルゴリズムが説明されている。

アルゴリズムの基本構造

「情報の科学」の教科書においてアルゴリズムの定義は、「問題を解くための処理手順」³⁾、「手順が有限個の処理からなり、各処理にあいまいさがなく、必ず結果を出して終了するときの手順」⁴⁾、「問題を解決するための効率的な手順を整理し、定式化したもの」⁵⁾と説明されている。



アルゴリズムにおける各処理を1つのブロックと考え、このブロックを繋げて手順を可視化したものをフローチャートという。

まず、アルゴリズムの3つの基本構造を、フローチャートを用いて説明する。どのアルゴリズムもこの3つの基本構造の組合せからなる。

順次構造

各処理が直線で繋がっている構造である。上から順番に処理が実行される。図-1では、処理1を実行した後に、処理2を実行する。処理は四角形の記号で描かれる。

選択構造

菱形の記号に条件(条件式)があり、条件が真か偽によって処理が分岐する。図-2では、条件が真であれば処理1を実行し、偽であれば処理2を実行する。処理後の手順を同じにしたい場合は、どちらかの直線に矢印を付けて、もう1つの直線に合流させる。

反復構造

上の角が切り取られた四角形の記号①と、下の角が

切り取られた四角形の記号②との間に処理を挟んだものである。

図-3は、①の条件が真である間は、処理を実行し、②まで進んだら①に戻る。また、①の条件が偽であれば、処理は実行せずに②まで飛び、そのまま進む。

教科書で扱う内容

次に、教科書では、どのような題材でアルゴリズムを学んでいるか、いくつか紹介する。

• 2つの整数の合計を求めるアルゴリズム(図-4)

フローチャートの開始部分と終了部分は、角丸長方形の記号が描かれる。また、上辺が斜めになった四角形の記号①は、キーボード入力といった手入力操作を表す。そして、記号②は、出力装置への表示を表す。aやb、cは変数と呼ばれ、数値や文字を格納する箱と考えてほしい(図-4A)。

• 偶奇判定アルゴリズム(図-5)

入力された整数を2で割ったときの余りが0か0でないかで、偶数か奇数かを判定するアルゴリズムである。選択構造の題材としてよく使われる。

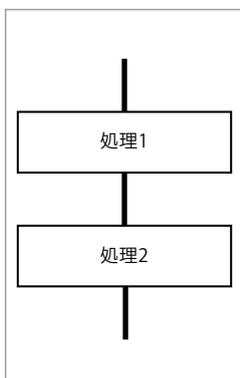


図-1 順次構造

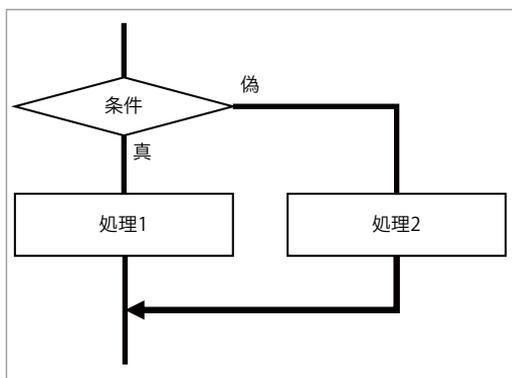


図-2 選択構造

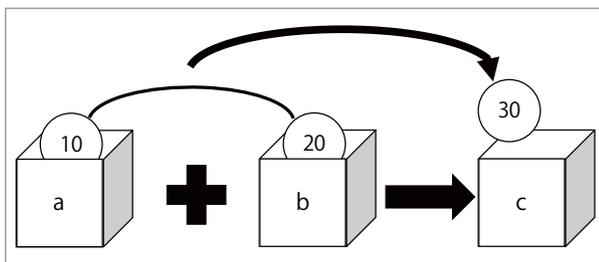


図-4A
変数のイメージ

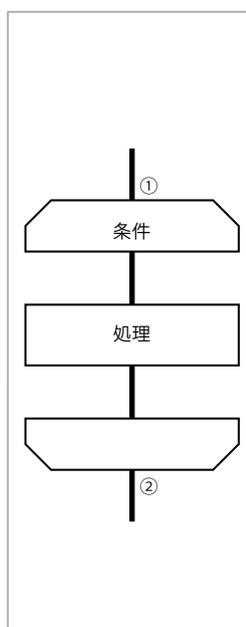


図-3 反復構造

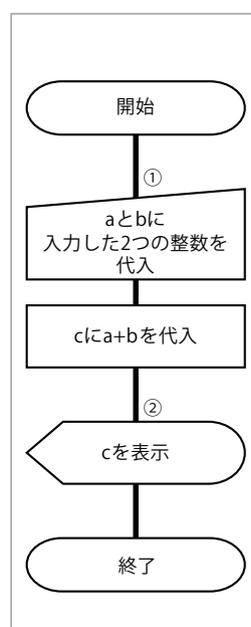


図-4 2つの整数の合計を
求めるアルゴリズム

•1からnまでの和を求めるアルゴリズム (図-6)

入力した整数 n に対し、1 から n までの和を表示するプログラムである (n が 0 以下の場合は 0 を表示)。ここで面白いのは、1 から n を足す順番によって、処理を1つ少なくできることである。条件を $i \leq n$ とし、1, 2, ..., n の順番で足す方法 (図-6の左) と、条件を $n > 0$ とし、n, n-1, ..., 1 の順番で足す方法 (図-6の右) とでは、後者の方が、i を定義しない分、フローチャートが短いことが分かる。

また、i に i+1 を代入といった、プログラミング独特の処理も登場し、生徒によっては興味を惹かれるかもしれない。

•交換法 (バブルソート) のアルゴリズム (図-7)

複数の数値を昇順 (小さい数から大きい数の順にする) や降順 (大きい数から小さい数の順にする) に並べ替える。フローチャートに従って、4つの数値 34, 92, 11, 75 を昇順に並べ替えた場合の実行の様子を表-2に示す。表-2の行が下がる (アルゴリズムが進む) に従い、大きい数 (たとえば 92) が右方向に移動して

いることが分かる。このように数値が右方向に移動することを泡が水面に浮かび上がることに例えて、バブルソートと名付けられている。

また、扱う数の個数が増えるため、変数ではなく配列 (a[]) を使用している。配列は、変数の箱が順番に並んでいて、中に数値や文字が書かれている球が入っている様子をイメージするとよい (図-7A)。

表-2の赤字の部分、図-7の選択構造の条件式の部分 ($a[j] > a[j+1]$) で比較対象となっている数値である。最初は、 $a[1] > a[2]$ が条件式となる。a[1]の中の数値は 34 で、a[2]の中の数値は 92 のため、条件式は偽となる。よって、a[1]の中の数値と a[2]の中

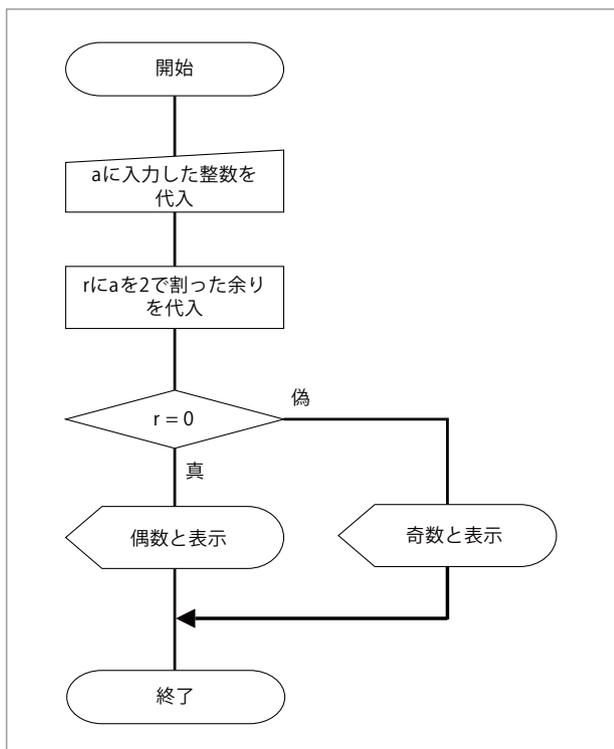


図-5 偶奇判定アルゴリズム

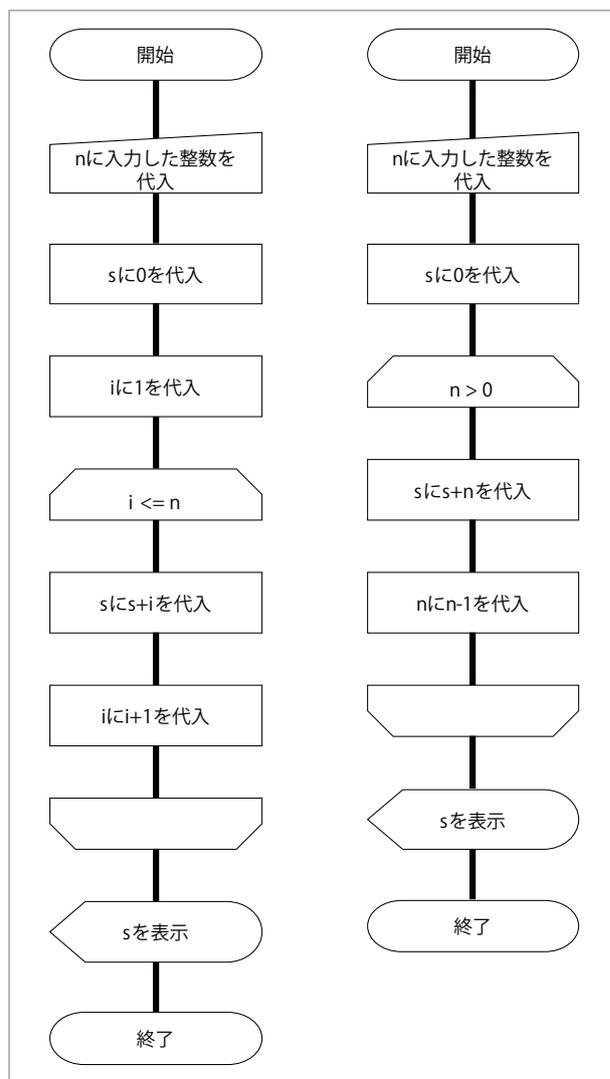


図-6 1からnまでの和を求めるアルゴリズム

の数値の入れ替えはしない。次に、 $a[2] > a[3]$ が条件式となる。 $a[2]$ の中の数値は 92 で、 $a[3]$ の中の数値は 11 のため、条件式は真となる。よって、 $a[2]$ の中の数値と $a[3]$ の中の数値の入れ替えをする。入れ替えの

際には、temp という変数を用いる。temp に $a[2]$ の中の数値を代入した後、 $a[2]$ に $a[3]$ の中の数値を代入する。そして、 $a[3]$ に temp の中の数値を代入する。コンピュータは、変数や配列に入っている数値を入れ替える際、このように一時的に使用する変数を用意する必要がある。これは、オレンジジュースが入ったコップの中身と、リンゴジュースが入ったコップの中身を入れ替えるとき、もう 1 つコップがないと入れ替えられないことをイメージするとよいだろう。

ソーティングのアルゴリズムまで来ると、かなりフローチャートが長くなるので、教科書によっては、図-7の右のようにして、反復構造の条件を表記することもある。

また、これ以外にも、逐次探索や二分探索、エラトステネスのふるいをはじめとした素数を求めるアルゴリズムなども掲載している教科書もある。どのような内容や題材が扱われているのかを表-3 にまとめた。

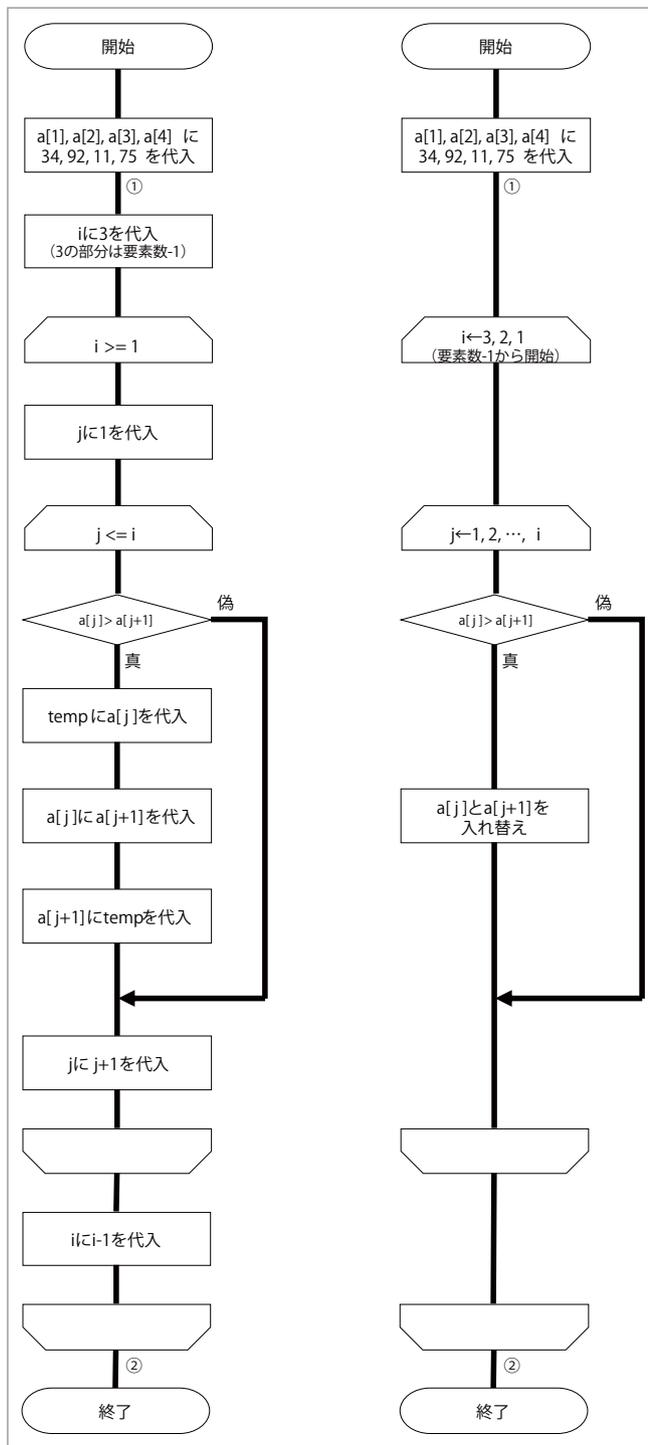


図-7 交換法（バブルソート）のアルゴリズム

教科書でのプログラミング言語

「情報の科学」の教科書では、これまで紹介したフローチャートと合わせて、プログラミング言語により、コンピュータ上でアルゴリズムを実際に動かせるようにしている。手元にある教科書を確認したところ、JavaScript または Microsoft Excel のマクロ機能

表-2 図-7のバブルソートの実行の様子

a[1]	a[2]	a[3]	a[4]	i	j	条件式 $a[j] > a[j+1]$	$a[j]$ と $a[j+1]$ を 入れ替え
34	92	11	75	←①の時点での状態			
34	92	11	75	3	1	$34 > 92 \dots$ 偽	しない
34	92	11	75	3	2	$92 > 11 \dots$ 真	する
34	11	92	75	3	3	$92 > 75 \dots$ 真	する
34	11	75	92	2	1	$34 > 11 \dots$ 真	する
11	34	75	92	2	2	$34 > 75 \dots$ 偽	しない
11	34	75	92	1	1	$11 > 34 \dots$ 偽	しない
11	34	75	92	←②の時点での状態			

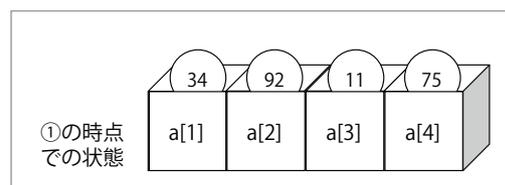


図-7 A 配列のイメージ

(VBA) のいずれかがプログラミング言語として使用されていた(図-8)³⁾。これらは、ブラウザやExcelがインストールされていれば使用可能というメリットがあるが、JavaScriptは、デバッグの際にブラウザの開発ツールを使用する必要があり、少しハードルが高い。また、情報系学部への進学を目指す生徒、国際情報オリンピック(使用できる言語はC/C++、Pascalのみ)出場を目指す生徒は、JavaScript、VBA以外の言語も学ぶ必要がある。

学習指導要領の改訂と これからのアルゴリズム教育

ところで、2020年度から、小学校では新学習指導要領に基づいた授業を実施している。中学校は2021年度から、そして高等学校は2022年度から実施される予定である。その際、共通教科情報科の科目は、現在の「社会と情報」、「情報の科学」から「情報Ⅰ」、「情報Ⅱ」に変更となる。「情報Ⅰ」は必修科目で、これまでの「社会と情報」、「情報の科学」の両方の内容を取り入れたものとなり、「情報Ⅱ」は選択科目で、「情報Ⅰ」の発展的な内容となる予定である。

また、新学習指導要領から、小中高でプログラミング教育が導入される。といっても、小学校1年生からプログラミング言語を用いたコーディングを行うのではなく、まずは、コンピュータに意図した処理を指示するという体験や、既存の教科と連携して、プログラミング的思考を深めることを行う。小学校段階におけるプ

ログラミング的思考とは、「自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、1つ1つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力」⁶⁾とされており、まさにアルゴリズム構築力といえるであろう。このような教育を受けた児童・生徒がどのようなアルゴリズムを創造するのか、将来に期待したい。

参考文献

- 1) 東京都教育庁:平成31年度使用都立高等学校(都立中等教育学校後期課程及び都立特別支援学校高等部を含む。)用教科書の採択結果について、<https://www.metro.tokyo.lg.jp/tosei/hodohappyo/press/2018/08/24/05.html> (2020年7月30日閲覧)
- 2) 文部科学省:高等学校学習指導要領解説情報編,pp.16-34 (2010).
- 3) 水越敏行, 村井 純, 生田孝至 他:新・情報の科学, 日本文教出版, pp.109-134 (2016).
- 4) 山口和紀 他:高等学校情報の科学, 第一学習社, pp.34-41, pp.114-117, pp.132-135 (2016).
- 5) 岡本敏雄, 山極 隆 他:最新情報の科学新訂版, 実教出版, pp.78-91 (2016).
- 6) 文部科学省:小学校段階におけるプログラミング教育の在り方について(議論の取りまとめ), https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/053/siryo/_icsFiles/afieldfile/2016/07/08/1373901_12.pdf (2020年7月30日閲覧)

(2020年7月30日受付)

山口健二 (正会員) yamaguchi.kenji@ocha.ac.jp

お茶の水女子大学サイエンス&エデュケーションセンター特任講師。同附属高等学校教諭(情報科)を兼務。2010年千葉大学大学院自然科学研究科単位取得退学。2017年お茶の水女子大学附属高等学校教諭(情報科)。2019年から現職。情報教育教材の開発や機械学習による企業情報データサイエンスに興味を持つ。修士(理学)。「どうぶつ並べ替えアプリでアルゴリズムを学ぼう」
<https://ocha-algo.com/>

表-3 教科書で扱われているアルゴリズム関連

探索	逐次探索, 二分探索
ソート	バブルソート, コームソート
数値	複数の数値の合計を表示, 複数の数値の中で最も大きい数値を表示, 偶奇判定
素数	入力した数より小さい素数を表示, 双子素数の表示, エラトステネスのふるい
配列	シーザー暗号で暗号化・復号
その他	巡回セールスマン問題の解説, 入力金額を満たす紙幣と硬貨の枚数を表示
	パズル(4x4の数独)を解く, 製造数を上手く調整して売上を最大化(線形計画法)

```

1 <html>
2 <body>
3 <script>
4 var a, r;
5 a = prompt("整数を入力してください");
6 r = a%2;
7 if(r==0){
8     alert("偶数");
9 }else{
10    alert("奇数");
11 }
12 </script>
13 </body>
14 </html>
15
Sub evenodd()
Dim a, r As Integer
a = InputBox("整数を入力してください")
r = a Mod 2
If r = 0 Then
    MsgBox ("偶数")
Else
    MsgBox ("奇数")
End If
End Sub
    
```

図-8 偶奇判定アルゴリズムのプログラム