

プログラミング教育におけるアルゴリズム的思考法のための 教育支援ツール利用の実践と評価

A practical use of an educational support tool for algorithmic thinking in programming education and its evaluation

萱津 理佳 Rika KAYATSU¹⁾, 香山 瑞穂 Mizue KAYAMA²⁾
國宗 永佳 Hisayoshi KUNIMUNE²⁾, 不破 泰 Yasushi FUWA²⁾

あらまし

プログラミング教育においてアルゴリズム的思考法のための教育支援ツールの導入を試みた。本研究の目的は教育支援ツールの利用により、これまで課題であったアルゴリズムを考え記述する力、および、C言語の学習においてつまずきやすい配列の学習を効率的かつ効果的に行う方法を探求することである。本ツールでは、グラフィカルにアルゴリズムを組み立てることができ、また実行のトレース機能により学習者自身の検証も可能である。2010年度の「プログラミング言語I」の配列の授業においてツールを導入し、その効果について考察した。アルゴリズムを記述する実験では導入前の年度の受講生との差異は現れなかったが、配列に関する理解度については定期試験の結果の比較よりツール利用の効果があったことが言える。

キーワード

アルゴリズム的思考 教育支援ツール プログラミング 配列

1. はじめに

社会の情報化に伴い、プログラミング教育の重要性は高まってきている。しかしながら、プログラミング教育における現場では、プログラミング教育が開始された当初からほとんど変わらない形式で授業が行われることが多いのが現状で、まず講義形式で対象となるプログラミング言語の文法について解説を行い、実習としてプログラムを作成せよという形式が一般的である。

筆者は、国立長野高専電気電子工学科「プログラミング言語I」（3年次通年、必修科目）においてC言語の授業を担当しているが、この従来のプログラミング教育における授業方法では、文法知識については理解していても、実際のプログラムを作成する段階になるとなかなかプログラムが書けない受講生が多いと感じている。これは、文法の学習に重点が置かれ、言語仕様を習得させることが目的かようになってしまっており、アルゴリズム的な思考が身についていないことが原因ではないかと考えた。そこで、課題が与えられた際、いきなりプログラム作成に入るのではなく、まず課題に対するアルゴリズムを日本語箇条書きで表現させることが必要と考え、

実践した。しかしながら、アルゴリズムを自然言語で記述させた場合、自由度が大きすぎ、受講生によって処理の粒度がまちまちで、曖昧な表現も多くみられる。また、作業を基本的な操作に分解できていない受講生もみられた。そこで、いくつかの記述ルールを形式化し、グラフィカルなインタフェースによりアルゴリズムを組み立てるアルゴリズム的思考法のための教育支援ツール [1][2] の導入を試みた。本授業におけるもう一つの問題として、C言語の学習において配列でつまずく受講生が多いことが挙げられる。そこで、配列の学習の際にこの教育支援ツールを利用することで、配列の構造を意識することなく、アルゴリズムを思考することができ、これにより配列を利用する際の処理手順や考えた方がわかり、C言語による配列の理解がスムーズに進むのではないかと予測した。

本論文では、筆者が担当する「プログラミング言語I」の配列の授業において、アルゴリズム的思考法のための教育支援ツールを利用した実践について報告する。以下、2節でアルゴリズム的思考法のための教育支援ツールの概要を述べ、3節で授業での導入について説明する。さらにツールを導入した効果に関して、アルゴリズムを記述する力についてと配列理解についての二つの側面から、それぞれ4節および5節で考察する。最後に6節でまとめを行う。

1) 長野県短期大学
2) 信州大学

2. アルゴリズム的思考法のための教育支援ツール

アルゴリズム的思考法とは、作業を基本的な操作に分解しそれらの順序を意識する思考法である [3]。本研究ではアルゴリズムを、ある目的を達成するための処理（仕事）の手順である、とする。その上でアルゴリズム的思考法とは、与えられた目的を達成するためのアルゴリズムを思考し、それを他に伝えるために形式化することと定義する。

本研究で利用したアルゴリズム的思考法のための教育支援ツール [1][2] は、学習者自身によるアルゴリズムの記述の検証、解答の見直しや参照を可能とし、教員による採点の労力の軽減を図り、より曖昧さが少ない形式化を意識させるために開発されたツールである。学習者としてログインした際のインタフェース例を図 1 に示す。構成要素の組み合わせでアルゴリズムの構造を表現し、各処理を日本語および項目選択の組み合わせで記述する。学習者による課題のスタート画面の例を図 2 に、課題の記述および実行画面の例を図 3 に示す。

本ツールの特徴は以下の 4 点である。

- * 構造記述と内容記述の分離：**アイコンで示される構成要素を所定の場所にドラック＆ドロップすることでアルゴリズムの構造を表現する。さらに、個々の要素における処理内容を日本語等で記述する。
- * 構成要素の利用の可否を制御：**管理者権限のユーザは、授業で導入されるルールに合わせて、アルゴリズムを構成する要素の利用の可否を制御する。
- * アルゴリズムのトレース機能：**アルゴリズムの実行をトレースできる。トレース時に値を表示させる変数を指定できる。



図 1 学習者ログイン画面の例

- * 提出物の状態管理：**指導者権限のユーザは、提出物の作成中・提出済、未評価・評価済、再提出等の状態を学習者ごと、課題ごとに確認できる。

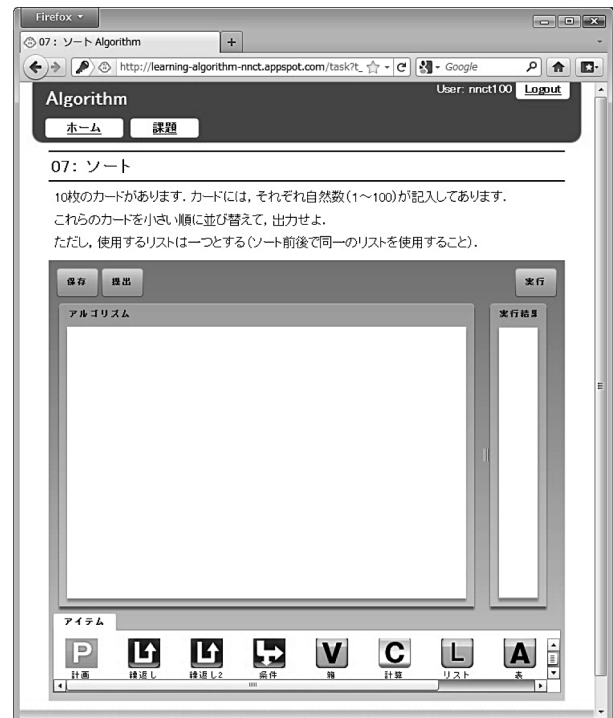


図 2 課題のスタート画面の例



図 3 課題の記述・実行画面の例

3. ツール利用の試み

「プログラミング言語 I」（通年，必修科目）の 2010 年度の受講生に対し，アルゴリズムを考え記述する力をつけること，C 言語の配列に関する理解を向上させることの 2 つを目的に，アルゴリズム的思考法のための教育支援ツールの導入を試みた。本年度は配列理解に焦点をあて，後期の配列学習の際にツールを利用した。前期より C 言語によるプログラミングの学習を始め，ツール導入時点で入出力，選択処理，反復処理まで学習済みであった。

2010 年 9 月 30 日の授業より，11 月 25 日までの合計 5 回の授業でツールを利用した。ツールを利用した課題は，ツールの操作に慣れてもらうための課題 2 題を含め全 9 題である。図 4 に課題の一覧を示す。

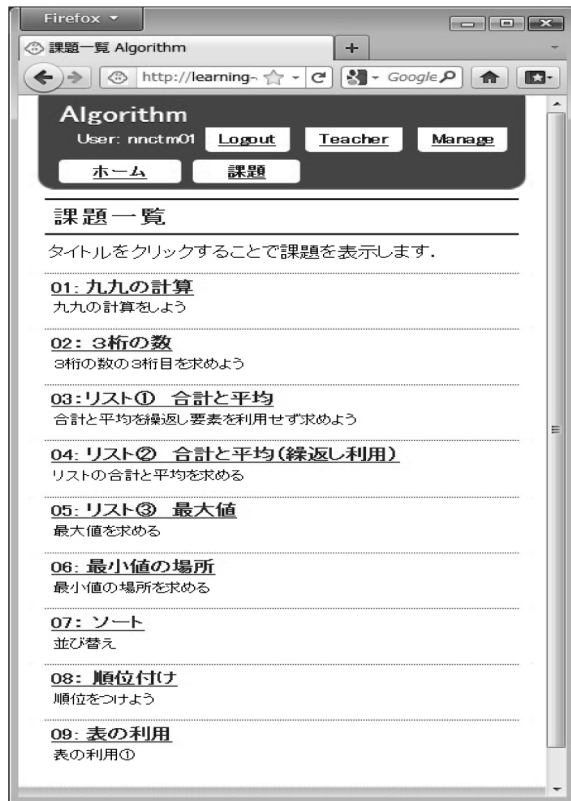


図 4 ツールで出題した課題一覧

アルゴリズムを考え，記述する力を評価するために，アルゴリズムの記述実験を実施した。ツールを導入する前年度（2009 年度）の受講生の結果と，ツールを導入した 2010 年度の受講生の結果を比較する。また，配列の理解度については，後期中間試験および後期期末試験の配列に関する問題の解答結果について，ツールを導入する前年度（2009 年度）

の受講生とツールを導入した 2010 年度の受講生を比較する。

4. アルゴリズム的思考に関する評価

アルゴリズムを記述する力が身についたか評価するために，アルゴリズムの記述実験を実施した。ツールを導入していない 2009 年度の受講生と授業でツールを利用した 2010 年度の受講生の結果を比較する。

以下に，記述実験の概要を示す。

- (1) 目的：アルゴリズムを記述する力を調査する。
- (2) 日付・被験者数：2009 年度生は 2009 年 12 月 11 日に実施。受講生 41 名。2010 年度生は，2010 年 12 月 9 日に実施。受講生 40 名。
- (3) 時間：30 分
- (4) 内容：二つの問題（問題 1，問題 2）のアルゴリズムを日本語で記述するというもの。

問題 1 から取りかかるものとしたため，問題 2 については時間がなく白紙や途中の解答が多かった。本論文では，問題 1 について報告する。問題 1 は，米 Vertigo Software の開発者 Jeff Atwood 氏が自分のブログで紹介して [4] 話題を呼んだ，まったくプログラムを書けないのにプログラムの採用試験に応募してくる人が後を絶たないことに業を煮やした開発者が，ふるい分けのために使っているという FizzBuzz 問題を利用した。FizzBuzz 問題は，「1 から 100 までの数を順に出力せよ。ただし，その数が 3 で割り切れるときは“Fizz”という文字列を，5 で割り切れるときは“Buzz”という文字列を，3 と 5 のどちらでも割り切れるときは“FizzBuzz”という文字列をそれぞれ数の代わりに出力せよ」というものである。

記述されたアルゴリズムを大まかに，正解，一部不備を含む，不正解の 3 種類に分類した。この結果を図 5 に示す。正解の割合は 2009 年度生の方が 18.5 ポイントほど高いという結果であった。しかしながら，一部不備を含むが基本的な処理の手順は正解であるものを含めると，2009 年度も 2010 年度も 8 割ほどとなっており，同程度の割合である。両年度とも不備なものとして多かったのが，(1) 無理に配列を宣言して使おうとするもの，(2) 変数の型の理解が不十分なもの（文字列と数値を同一変数に代入），(3) 問題文の順序で処理を記述し，3 と 5 両方の倍数の場合の処理に不備があるもの，の 3 種類であった。(1) に関しては，配列の学習を終えたところでの実験だったため，配列の問題だと思いこ

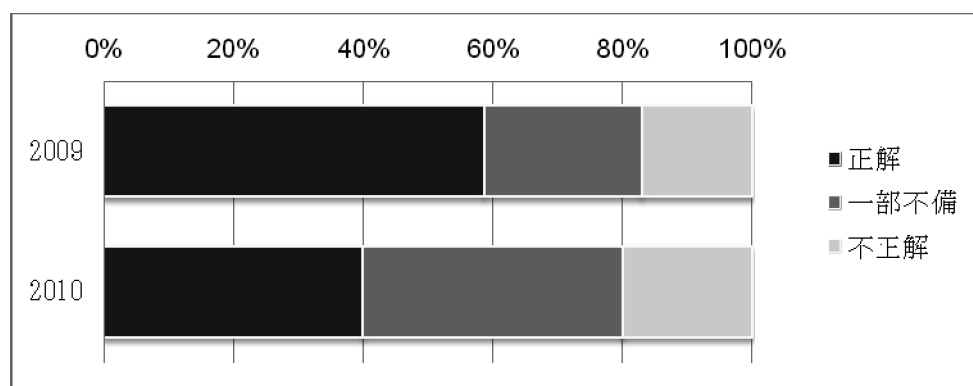


図5 FizzBuzz 問題のアルゴリズム記述実験結果

んで取り組む受講生が多かったのではないかとと思われる。

5. 配列の理解についての評価

配列の理解度について評価するため、ツールを導入する前の受講生とツールを導入した2010年度の受講生の後期中間試験および後期期末試験の試験結果について比較する。

5-1 ツール利用以前との試験結果の比較

「プログラミング言語I」では、2007年度より現在のシラバスで授業を実施している。2007年度から2010年度までの定期試験の平均点を表1に示す。また、ツールを利用した2010年度とツール導入前の3年間の平均点を表2に示す。

2010年度の後期でツールの利用を試みたが、ツールを利用する前の前期中間、前期期末では2010年度とそれ以前の3年間の平均点はほぼ同じである。学年末の試験では、ツールを利用した2010年度の

平均点が、過去3年の平均点より7ポイント程度高くなっていることがわかる。平均点の差は、受講生の理解度だけでなく、問題の難易度にも左右されることから、平均点のアップがツールの利用による効果であるとは断定できないが、ツールの利用はプログラミングおよび配列の理解向上に一定の効果があったのではないかと考えられる。

5-2 同一問題に対する2009年度、2010年度の試験結果の比較

2009年度の定期試験においては、配列に関する問題として後期中間試験で3問（配列問題01～配列問題03とする）、後期期末試験で2問（配列問題04～配列問題05とする）出題した。そこで、ツールを導入した2010年度の定期試験において2009年度と同一の配列問題を出題し、結果を比較した。2009年度、2010年度の配列に関する問題の試験結果をそれぞれ表3、表4に示す。また、2009年度と2010年度の得点率の比較を図6に示す。

表1 2007年度～2010年度までの定期試験の平均点

年度	前期中間	前期期末	後期中間	学年末	4回の平均
2007年度	82.6	58.6	83.6	72.6	74.4
2008年度	77.6	61.0	76.8	65.0	70.1
2009年度	72.2	60.0	79.2	62.2	68.4
2010年度	77.3	59.4	79.8	73.5	72.5

表2 2010年度とそれ以前の定期試験の平均点

年度	前期中間	前期期末	後期中間	学年末	4回の平均
2007～2009 平均	77.5	59.8	79.9	66.6	70.9
2010年度	77.3	59.4	79.8	73.5	72.5

表 3 2009 年度の配列に関する問題の試験結果

	配列問題 01	配列問題 02	配列問題 03	配列問題 04	配列問題 05
平均点	13.4	17.9	23.5	11.0	5.9
配点	16	20	30	17	16
得点率	83.8%	89.5%	78.3%	64.7%	36.9%
分散	12.0	8.9	35.0	31.6	36.2
標準偏差	3.5	3.0	5.9	5.6	6.0

表 4 2010 年度の配列に関する問題の試験結果

	配列問題 01	配列問題 02	配列問題 03	配列問題 04	配列問題 05
平均点	14.2	18.5	21.3	13.1	9.7
配点	16	20	28	20	16
得点率	88.8%	92.5%	76.1%	65.5%	60.6%
分散	7.6	3.4	57.2	34.2	35.6
標準偏差	2.8	1.8	7.6	5.8	6.0

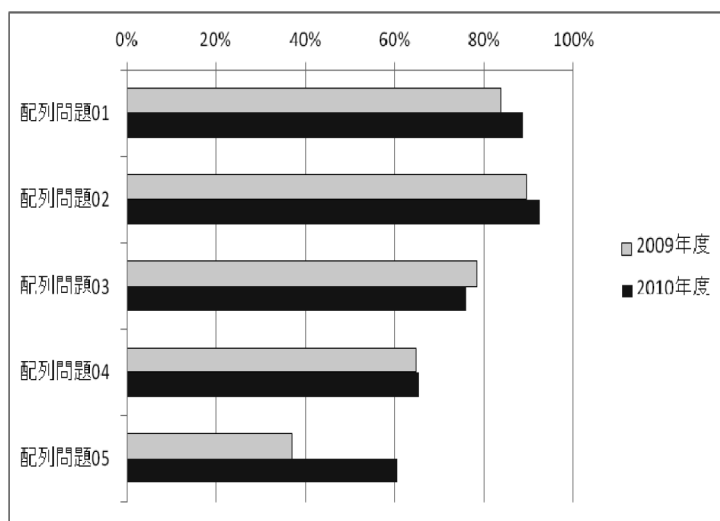


図 6 配列問題に関する 2009 年度と 2010 年度の得点率の比較

配列問題 03 をのぞいた 4 問に関して、ツールを導入した 2010 年度の受講生の方が、得点率が高い結果となった。また、後期中間試験に実施した配列問題 01 および配列問題 02 では、2010 年度の受講生の方が平均点が高く分散が小さくなっている。後期期末試験に実施した配列問題 05 では、2010 年度の受講生の方が 23.7 ポイント得点率が高い。これらの結果より、2009 年度の受講生より 2010 年度の受講生の方が配列に関する理解度が高いと言える。

5-3 配列の理解について

定期試験においては、授業中に扱った問題を出題することが多く、5-2 で示した配列問題 01～05 も、授業内で扱った問題である。配列を理解していなくても、試験勉強のためにプログラム等を丸覚えした受講生は、試験に正答することは可能であると思われる。そこで、2010 年度の後期中間試験において、授業では取り扱わない新規の配列に関する問題（配列問題 06 とする）を出題した。配列問題 06 の試験結果を表 5 に示す。

表5 2010年度の配列に関する新規問題の試験結果

	配列問題 06
平均点	12.1
配点	16
得点率	75.6%
分散	13.7
標準偏差	3.7

後期中間試験において実施した4問のうち、新規の問題である配列問題06と授業で扱った問題である配列問題01～03の解答結果の相関係数を表6に示す。また、相関図の一例として配列問題06と配列問題01の相関を図7に示す。

表6 新規の問題と既出の問題における結果の相関係数

対象	相関係数
配列問題06と配列問題01	0.67
配列問題06と配列問題02	0.50
配列問題06と配列問題03	0.64

2010年度の受講生に対し、授業では扱わなかった新規の配列に関する問題として出題した配列問題06は得点率が75.6%で、その他の配列問題と比較しても大きな差がなく、授業内での既出の問題との相関もみとめられる。これより、2010年度の受講

生が配列に関して一定の理解ができていることがいえる。

6. おわりに

プログラミングの授業で課題であったアルゴリズムを考え記述する力の不足を補うための学習、および、C言語の授業においてつまづきやすい配列の学習を効率的かつ効果的に教育することを目的に、C言語を学習する「プログラミング言語I」の授業においてアルゴリズム的思考法のための教育支援ツールの導入を試みた。グラフィカルにアルゴリズムを組み立てることができるツールを利用することで、配列の構造を意識することなく、アルゴリズムを思考することができ、これにより配列を利用する際の処理手順がわかり、C言語による配列の理解がスムーズに進むのではないかと予測した。

アルゴリズムを記述する実験では、ツールを導入した2010年度の受講生と導入前の年度の受講生の差異は認められず、ツール導入による効果はみられなかった。しかしながら、配列に関する理解度については、定期試験の結果の分析よりツール利用の効果があつたのではないかと考えられる。

アルゴリズム的思考法のための教育支援ツールは、大学等での運用成果を踏まえ、仕様および機能を更新中である。今後は、2010年度での実践をもとに、受講生のプログラミング能力を向上させるためには、どのような授業展開の中でツールを利用したらより効果的か検討していきたい。

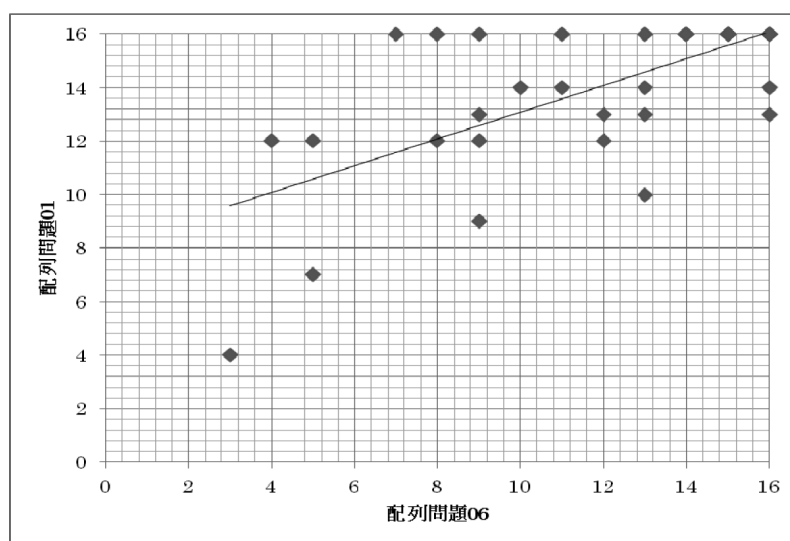


図7 新規の問題（配列問題06）と既出の問題（配列問題01）の相関

参考文献

- 1) 香山瑞恵, 永井孝, 國宗永佳他: “アルゴリズム的思考法のための教育支援ツールの開発”, 日本情報科学教育学会第3回全国大会, 27A2-4 (2010)
- 2) 香山瑞恵, 永井孝, 山本樹他: “グラフィカルなインタフェースによるアルゴリズム的思考法教育支援の試み”, 教育システム情報学会 (JSiSE) 第35回全国大会論文集, 26-D 1-04 (2010)
- 3) 飯田周作, 飯田千代他: “アルゴリズム的思考法の教育”, 情報処理学会コンピュータと教育研究会報告, 2008-CE-93 (2008)
- 4) raganwald2008, <http://weblog.raganwald.com/2007/01/dont-overthink-fizzbuzz.html>, (January 24, 2007)