

# 教師の負荷軽減のためのプログラム自動部分採点システムと 解答状況分析システムの開発

大館 裕紀<sup>\*1</sup>, 前田 新太郎<sup>\*2</sup>, 東本 崇仁<sup>\*1</sup>

<sup>\*1</sup> 東京工芸大学, <sup>\*2</sup> 東京工芸大学大学院

## Development of Program Automatic Partial Scoring and an Answer Status Analysis System to Reduce the Workload on Teachers

Yuki Oodate<sup>\*1</sup>, Shintaro Maeda<sup>\*2</sup>, Takahito Tomoto<sup>\*1</sup>

<sup>\*1</sup> Tokyo Polytechnic University, <sup>\*2</sup> Graduate School of Tokyo Polytechnic University

プログラミングの講義では、学習者にプログラミングを構築させる課題を与えることが多い。効果的な学習を実現するためには、各自の理解状況にあわせたフィードバックや、教室全体の傾向にあわせたフィードバックを教師が行うことが望ましい。しかし、教師が全学習者に対して採点と分析を行うことは負荷が高い。そこで本研究では、誤り個所の特定を指向した部分一致による自動部分採点システムと、教室全体の解答状況の確認と分析が行える解答状況確認システムの開発と評価を行った。手法として、正解として用意されたプログラムと学習者の解答の部分的な一致度を取り、それに基づいた部分採点を行う。さらに、各学生の誤り個所を統合し、全体の解答状況の分析を行う。評価実験では、学習者と教師にアンケートを行い、本システムが教師の負荷軽減と効率的な授業の推敲のために有効であることと、学習者にとって有効な指導がうけられることが確認された。

キーワード: 負荷軽減, 部分採点, 部分一致, 学習分析, 形成的評価

### 1. はじめに

プログラミングの講義では、教師が学習者に演習形式でプログラムを作らせることが多い。効果的な学習を実現するためには、個々の学習者の誤り個所を分析し、フィードバックを与えると同時に、教室全体の誤り傾向を分析し、皆が間違いやすい点などについてフィードバックを行うことが重要である。さらに、学習者の解答に対しては、正しいか正しくないかだけでなく、部分採点を行い、どこまでできているかの評価を与えることが有用である。しかし、これらを実現するためには、教師が個々の学習者のプログラムに対して、正誤判断だけでなく、誤り個所の特定や、部分採点を行う必要があり、さらに教室全体の傾向を分析する必要がある。プログラミングの授業で学習者に構築させ

るプログラムは、1回の講義においても複数個存在するため、そのすべてを分析、採点することは教師の負荷として現実的ではない。さらに、可能であれば学習者の解答に対して、即時集計、即時フィードバックを行うことが望まれる。

そこで本研究では、部分一致による自動部分採点システムと、講義内の学生の解答状況の分析が行える解答状況確認システムを開発した。この2つのシステムを用いることで、教師の採点の負荷を軽減し、効果的な授業を促進するとともに、学習者も自身の解答について振り返りながら、効果的な授業をうけることが可能になる。

## 2. 関連研究

小西ら<sup>(1)</sup>や鈴木ら<sup>(2)</sup>は、出題意図に基づいて、学習者のプログラムを「教師が記述したアルゴリズム」「大部分が一致するもの」「それ以外」に弁別し、教師の採点の負担を減らすシステムを開発している。このシステムは単にプログラムを採点するだけではなく、出題意図に基づいてアルゴリズムを採点する有用なシステムではあるが、教師は正解のプログラムだけではなく、正解のアルゴリズムの記述を別途行う負荷があり、弁別された後の学習者の個別の誤り個所の特定を行う負荷は残されている。

大和ら<sup>(3)</sup>は、プログラムにおいて基本ブロックと意味ブロックを定義し、部分採点を試みている。基本ブロックは、正解プログラムと学習者のプログラムにある入出力の変数と、入出力先の変数を比較して採点を行う。意味ブロックは、基本ブロックが集まり、意味的なまとまりとして構成されるブロックである。また、意味ブロックは基本ブロックの点数に重みづけを行い計算する。この重みや計算方式は、教師がそれぞれ設定をする。大和らの研究では、重みや意味ブロックというまとまりの定義は教師が行う必要があるものの、通常のプログラムの準備だけでよく、また個別の誤り個所の特定も行え、部分採点も行える。しかし、大和らのシステムでは、部分採点は教師のために行われるものであり学習者にフィードバックされる仕組みはない。また、教室全般の解答状況について分析する機能がないため、教師は教室全体の傾向や注意点を学習者にフィードバックすることができない。

平嶋ら<sup>(4)</sup>は、キットビルド概念マップという方法を提案し、教室内の理解状況を重畳表示するシステムを開発している。本システムはあらかじめ教授者が正解となるゴールマップを用意し、そのマップを分解して学習者に組み立てさせる方法である。本手法では、学習者は同じ部品を用いてマップを組み立てるため比較表示が可能となる。

## 3. 提案手法

本研究では、大和ら<sup>(3)</sup>の部分採点の手法と平嶋ら<sup>(4)</sup>のキットビルドの手法を組み合わせて、学習者に部品ベースでのプログラミング活動を行わせ、システムが

自動部分採点を行い、教室内の解答状況を集計する方法を提案する。

### 3.1 部分採点の手法

本研究では、大和らの木構造と基本ブロックや意味ブロックを用いて部分採点を行う。図1に木構造の例を示す。教師は通常の授業と同じように正解となるプログラムを作成する。その後、そのプログラムを授業の教育目標に応じて、機能ごとに意味ブロックに分割する。これを繰り返し最終的なステートメントを基本ブロックとする木構造を構築する。その後、各基本ブロックと意味ブロックに得点を付与する。例えば、図1の例でいうと、意味ブロック1が60点、意味ブロック2が40点という配点をつけるとする。このとき、ある意味ブロックから分岐した複数の子ブロックの合計点は100点となるように設定する。同様に意味ブロック1の子供である意味ブロック3が70点、基本ブロック1が30点という配点をつけ、意味ブロック3の子供である基本ブロック3に50点、基本ブロック4に50点などと配点したとする。この際、基本ブロック3が正解、基本ブロック4が間違いだとすると、意味ブロック3は100点中の50点となる。さらに、意味ブロック3は意味ブロック1に対して70点という配点を持つので、意味ブロック3が50点であることは、意味ブロック1に対して35点という得点をえることになる。これを繰り返し、部分採点を行う。

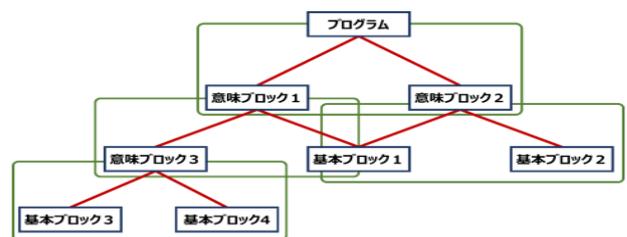


図1 木構造の例

### 3.2 部分一致による部分採点の手法

各意味ブロックは2個以上の基本ブロックまたは意味ブロックで構成される。そこで、本研究では順序関係や構造に基づく部分一致による部分採点を行う。例えば、図2の例では、正解は  $c = a;$  の後が、 $a = b;$  であり、そのあとが  $b = c;$  という順序を持つ。したがって、学習者の解答と順序関係に基づいて比較すると、 $a = b;$  と  $b = c;$  が正解と一致している（部分点 60 点）とみな

すか、 $c = a$ ;のみが正解と一致している（部分点 40 点）とみなすことができる。この際、システムはより部分点が高くなるように採点する。これは部分点が高い項目は、教授者がより教育的に意味があると考えているためである。同様に、階層関係についてもより点数が高くなるように採点する。

正解	得点	解答	得点
1. $c = a$ ;	40	1. $a = b$ ;	30
2. $a = b$ ;	30	2. $b = c$ ;	30
3. $b = c$ ;	60	3. $c = a$ ;	40

図 2 部分採点の例

### 3.3 解答状況の分析

3.1,3.2 の方法で学習者と正解の差の取得と各学習者の部分点を取得できる。本研究では、教室全体の理解状況の可視化として、「各問題の平均点の集計」と「各問題における意味ブロックごとの平均点の集計と誤り個所の特定」を提案する。

各問題の平均点の集計は、各学生の各問題で求めた部分点から、各問題における学生全体の平均点を集計する方法である。この方法により、一目でどの問題において学生全体が良い成績であるか、悪い成績であるかを確認することができる。

さらに、各問題における意味ブロックごとの平均点の集計と誤り個所の特定では、各問題の意味ブロック単位の点数について集計し、平均点を求める。この機能により、各問題で学習者が躓いている意味ブロックがどこであるかをより詳細に把握できる。さらに、点数ではなく、誤り個所の重畳表示により、学習者がどこで詰まっているか、どのような誤りを犯しがちであるかを瞬時に集計できる。なお、本稿では、「各問題における意味ブロックごとの平均点の集計と誤り個所の特定」の機能の実装までは至れなかったため、システムおよび評価については「各問題の平均点の集計」についてのみ行ったものについて述べる。

## 4. システム実装

### 4.1 システム概要

本システムの概要について説明する。本システムは、

プログラミング講義を受講する大学生と、受講生を指導する教師を対象とする。システムは 2 つ存在し、学習者が問題を解答し、その診断と採点を自動で行うシステム（自動部分採点システム）と教師が学習者の解答状況を確認するシステム（解答状況確認システム）となる。

本研究では前述したように部品ベースでのプログラミングを要求する。そこで、そのインタフェイスとしては古池らの開発したシステム<sup>(5)</sup>に準ずることとする（図 3）。教授者はあらかじめ正解となるプログラムを作成し、各プログラムにおいて図 1 のように意味ブロックと基本ブロックを設定し、配点を行っておく。

図 4 にシステム利用の流れを示す。学習者は、図 3 のインタフェイスにおけるブロックリストから適切と思うブロックを選択し、作業スペースに移し、パラメータを入力することで自分のプログラムを構築する。自動部分採点システムでは、3.1,3.2 で述べた方法に基づいて部分採点する。部分採点の結果は学習者にフィードバックされるため、学習者は自身のプログラムを即時に修正することができる。学習者の解答はインポートされ、解答状況確認システムですべての学習者の解答がエクスポートされる。教師は、受け取った解答データを解答状況確認システムで誤り個所の特定と分析を行う。各学習者の個別の誤り個所の特定や部分点を知ることができると同時に、教室全体の各問題における部分点に基づいた平均点を知ることができる。教師はこの教室全体の情報に基づいて、採点の負荷が一切かからず、皆の理解度が低い問題や高い問題を特定し、効果的な解説を行うことが可能となる。



図 3 入力インタフェイス

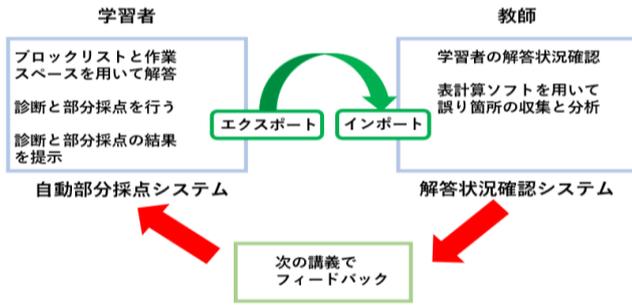


図 4 システム構成の流れ

#### 4.2 自動部分採点システム

自動部分採点システムが学習者に得点をフィードバックしている例を図 5 に示す。学習者がプログラムを構築すると、システムが自動で診断を行う。診断結果は画面右側にある「得点」に表示され、学習者は診断結果を確認することができる。学習者はこの得点を参考に試行錯誤を行うことができるが、正解に至れない場合でも「次の問題」をクリックすることで次の問題へ移行することができる。学習者はすべての問題を解答し終わったら、ファイル化された解答データを教師に送信する。図 5 の例では変数  $a$  と変数  $b$  を入れ替える問題である。この問題では、正解値を  $c = a$ ,  $a = b$ ,  $b = c$  と設定しており、得点を 100 点換算として 40 点、30 点、30 点と配分している。今回の例では、1 段目のブロックの入力  $c = a$  と 3 段目のブロックの入力  $b = c$  が正解であるため、得点としては、 $40 + 30 = 70$  点となる。

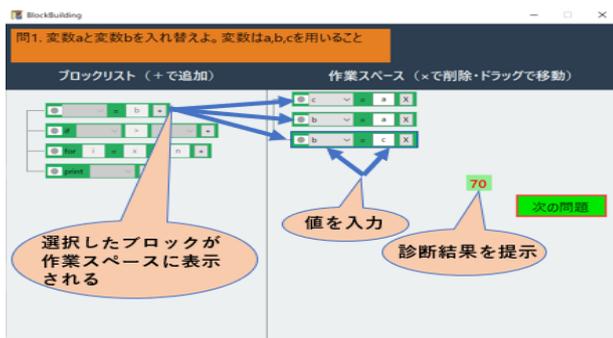


図 5 解答方法の例

#### 4.3 解答状況確認システム

解答状況確認システムを図 6 に示す。本システムは、自動部分採点システムで採点した結果を学習者ごとに表示する。教師は、画面左側に表示されている学習者のデータから解答状況を確認する。次に画面右側にある「全体の解答状況」ボタンをクリックし、問題ごと

に、ブロックのまとめりごとに集計した結果を表示する。教師は全学習者の解答状況を確認したら、「出力」ボタンをクリックすると、表計算ソフトで閲覧可能な csv ファイルが生成される。教師は、生成された csv ファイルを基に詳細な誤り個所の分析を行うことができる。図 6 の例では、問 1 の全学習者の解答状況を表示している。これにより、教師は個別の学習者の理解状況を知ることができるとともに、教室全体の平均点や標準偏差を知ることができ、全体の理解状況や理解状況のばらつきを知ることができる。

学習者名	問1	得点
A	正解したID: 代入 代入 代入 解答したID: 代入 代入 代入 解答: cf代入aaf代入bbf代入c	100
B	正解したID: 代入 代入 代入 解答したID: cf代入bfb代入abf代入c 解答: cf代入bfb代入abf代入c	30
C	正解したID: 代入 代入 代入 解答したID: 代入c a f代入 a c f代入 b 解答: b f代入 c a f代入 a c f代入 b	90
D	正解したID: 代入 代入 代入 解答したID: 代入 a a f代入 b b f代入 c 解答: cf代入 a a f代入 b b f代入 c	100
E	正解したID: 代入 代入 代入 解答したID: 代入 代入 代入 解答: c f代入 a a f代入 b b f代入 c	100
F	正解したID: 代入 代入 代入 解答したID: 代入 代入 代入 解答: a f代入 c c f代入 b b f代入 a	0
G	正解したID: 代入 代入 代入 解答したID: 代入 代入 代入 解答: 代入 代入 代入	0

図 6 解答状況の確認方法の例

### 5. 評価実験

#### 5.1 実験概要

開発したシステムの効果を検証するために、工学部大学生 14 名と教師 1 名を被験者として評価実験を行った。図 7 に実験の流れを示す。まず学習者には、1 回目のシステム利用を行ってもらい、解答データを収集する。なお、1 回目のシステム利用では学習者の理解状況のみを取得するために、診断結果である部分採点の得点は表示させないようにした。次に、収集した学習者の解答に基づいて、解答状況確認システムを教師に利用してもらい、解説動画を作成していただいた。さらに、その感想について回答してもらった。最後に、教師が作成した 20 分間の解説動画を学習者に視聴してもらい、学習者に 2 回目のシステム利用とアンケートを行ってもらった。

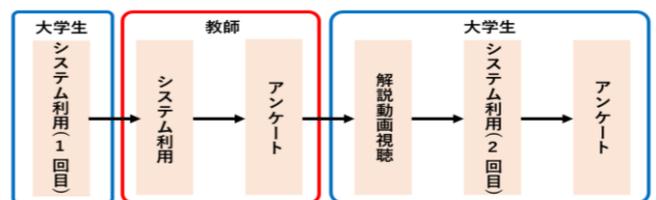


図 7 実験の流れ

## 5.2 解答状況確認システムのアンケート結果

教師の回答結果を表1に示す。誤り分析における部分採点の有効性では、「全体の部分点の平均がすぐわかるため、解説が必要な問題の有無が瞬時にわかる」という評価や、「受講者全員分、部分点込みで誤り個所をメモしながら採点するのはあり得ない労力であるため、教員の負荷軽減としては実に効果的」という評価がえられた。解答状況確認システムの評価では、「個別の解答状況も見られるため、よくある間違いを発見し、全体に対する注意ができる」という評価や、「ある学習者の各問題の解答状況が見られるので、問題の関係性を瞬時にえることができ、問題の関係性や注意点を学生にフィードバックできる」という問題間の関係の分析も行えるという評価もえられた。

このことから、本研究で目的とする教師の作業負担削減と効果的な授業の支援が行えることが示唆された。さらに、想定していなかった問題間の関係性の分析においても有用である可能性が示唆された。

表 1 教師によるアンケート結果

No	質問内容	回答内容
1	部分一致の採点は誤り個所の分析において有効か	全体の部分点の平均点などがすぐわかるので、解説が特に必要な問題や、まったく必要がない問題が瞬時にわかる。
2	誤り分析を行うための支援ができているか	個別の解答状況も見られるため、よくある間違いを発見し、全体に対する注意ができる。

## 5.3 学習者による部分採点の評価

表2に自動部分採点システムの1回目と2回目の利用による解答結果を示す。1回目の平均点より、学習者の理解状況について各問題にばらつきがあることがわかる。また、解説動画を閲覧後に行った2回目の学習における平均点について、問題1,3,4,5で向上が見られた。この結果と前節による教師のアンケート結果から、システム利用により、学習者の誤り個所に対し

て適切に教師が補足し、解説を行わせていることが示唆された。

表3に6件法を用いた学習者によるアンケート結果を示す(6:とてもそう思う-1:思わない)。アンケート結果から、学習者の理解していない個所について適切に解説動画で解説されており、理解に繋がるようになったことが示唆された。

表 2 学習者の解答結果

No	問題内容	1回目の平均点 (正答率)	2回目の平均点 (正答率)
1	値を入れ替える	52.9(36%)	100.0(100%)
2	条件を満たすとき値を入力	100.0(100%)	96.4(93%)
3	値を昇順に入れ替える	62.1(29%)	97.1(93%)
4	条件を満たすとき配列を出力	15.0(0%)	89.3(79%)
5	九九の表を出力する	23.6(0%)	76.8(0%)

表 3 解説動画と部分採点の評価

No	質問内容	平均評価
1	解説動画を視聴して理解が深まったか	5.21
2	解説動画と採点結果により誤り個所に気付けたか	5.50
3	解答した採点結果を確認できることは有効か	5.14
4	解答を部分ごとに採点することは有効か	5.29

## 6. システムについての検討

教師が学習者の解答から誤り分析する負荷をかけずに行うためには、学習者の解答から誤り個所を明示的にする部分一致が必要である。正確な部分一致を行う

ためには、プログラムを意味のあるまとまりとして認識することが必要であると考えた。そこで著者らは、古池ら<sup>6)</sup>の部品の段階的拡張手法を用いることで、より正確な部分一致ができるのではないかと考えた。

現状のシステムでは、教師が問題と解答を作成後に、手動で得点配分することが要求されるため、一定の負荷が教師にかかっているといえる。そこで、教師が作成した問題・解答に対して、システムが自動で得点配分できればより有用である。現時点では、各基本ブロックの重みを均一にすることで、教師が採点を入力しなくても自動で配点を決めることが可能となると考える。一方で、教師の教育目標を反映するために手動採点もまた重要である。そのため、自動での配点を与えたのちに、教師が必要に応じて配点を操作できる機能の開発が重要であると考えた。

解答状況確認システムでは、現状では部分点の平均点の集計しか実装できていない。そこで、3.3で述べた「各問題における意味ブロックごとの平均点の集計と誤り個所の特定」を行い、重畳表示をすることができれば、全体の学習者の理解できていない個所についてブロック単位で分析し表示したり、同じ誤りをしている学習者を集計して表示したりすることが可能となる。また、場合によってはカンニングの検知なども期待できる。

## 7. おわりに

プログラミングの講義では、教師が学習者にプログラム作成をさせる課題が多い。効果的な学習を実現させるためには、個々の学習者の誤り個所を分析し、理解状況にあわせたフィードバックや、教室全体の傾向にあわせたフィードバックを教師が行うことが重要である。しかし、教師が全学習者に採点と分析を行うことは負荷が高い。そこで本研究では、部分一致による自動部分採点システムと、講義内の学生の解答状況の分析を行える解答状況確認システムの開発と評価を行った。

本研究では、大和ら<sup>3)</sup>の部分採点と平嶋ら<sup>4)</sup>のキットビルド手法を組み合わせて、学習者にプログラミング活動を行わせ、システムが自動で部分採点を行う手法を提案した。さらに、教室内の学習者の解答状況を

解答状況確認システムに集計する方法を提案した。

評価実験による教師のアンケート結果から、採点と分析作業の負荷削減と効果的な授業の支援が行えることが示唆された。さらに、問題間での関係性の分析が行える可能性が示唆された。学習者の解答結果からは、解説動画の閲覧後に行った学習において、ほとんどの問題において平均点の向上が見られた。このことから、システム利用により、学習者の誤り個所を教師が適切に補足し、解説を行わせたことが示唆された。また、学習者のアンケート結果から、学習者の理解していない個所を適切に解説されており、理解に繋がるようになったと示唆された。

## 謝辞

本研究の一部は JSPS 科研費 JP21H03565, JP19H04227 の助成による。

## 参 考 文 献

- (1) 小西達裕, 鈴木浩之, 伊東幸宏: “プログラミング教育における教師支援のためのプログラム評価機構”, 電子情報通信学会論文誌, D-I, Vol.J83-D-I, No.6, pp.682-692 (2000)
- (2) 鈴木浩之, 小西達裕, 伊東幸宏: “抽象的データ構造を含むアルゴリズム表現に基づくプログラム評価支援システムの構築”, 教育システム情報学会誌, Vol.24, No.3, pp.167-186 (2007)
- (3) 大和雄一郎, 吉永泰輔, 竹内章: “C プログラムの診断と重み付け採点を行う学習支援システム”, 情報処理学会研究報告, pp.1-7 (2009)
- (4) 平嶋宗, 長田卓哉, 杉原康太, 中田晋介, 舟生日出男: “キットビルド概念マップの小学校理科での授業内利用の試み”, 教育システム情報学会誌, Vol.33, No.4, pp.164-175 (2016)
- (5) 古池兼人, 東本崇仁, 堀口知也, 平嶋宗: “プログラミングの構造的理解を指向した部品の段階的拡張手法の提案と支援システムの開発・評価”, 教育システム情報学会誌, Vol.36, No.3, pp.190-202 (2019)
- (6) 古池兼人, 東本崇仁: “プログラミングにおける構造的理解のための部品の段階的拡張手法の提案とそのシステムの開発”, 教育システム情報学会誌, Vol.35, No.2, pp.215-220 (2009)