

プログラムの動作理解を表出する課題の実践と評価

山本 樹^{*1,2}, 華山 宣胤^{*1}, 國宗 永佳^{*3},

^{*1} 尚美学園大学 情表表現学科

^{*2} はこだて未来大学大学院システム情報科学研究科

^{*3} 信州大学学術研究院工学系

Implementation and Evaluation of Tasks to Express Learner's Understanding of Program Behavior

Tatsuki YAMAMOTO^{*1,2}, Nobutane HANAYAMA^{*1}, Hisayoshi KUNIMUNE^{*3},

^{*1} Faculty of Informatics for Art, Shobi University

^{*2} Graduate School of Systems Information Science, Future University Hakodate

^{*3} Institute of Engineering, Shinshu University

プログラミングを学習する過程では、学習者がプログラムの動作を思考し、思考したプロセスを自己の中で形成し、形成したプロセスを、実際のプログラムの動作と一致させることが重要である。この教育を行うため、完成したプログラムとその動作を学習者が記入する表を並べて提示した課題を用いてプログラムの動作理解を表出するための課題を用いた授業を実施した。本稿では、この課題を実施した後の試験結果から、誤答箇所を抽出し、学習者がプログラムの動作に関する誤答傾向を把握するための分析方法を提案し、分析結果から誤答傾向を見出すことを目的とする。

キーワード: プログラミング教育, プログラムの動作理解, 誤答傾向

1. はじめに

筆者らは、手続き型構造化プログラミングの教育を行っているが、教育する中で、分岐・反復などの制御構造化でつまづく学習者が多く存在することがこれまでの研究で明らかになっている⁽¹⁾。その結果として、プログラムの作成が困難な学習者が多数存在する。このことから、プログラミングを教育する上で教授者が特に留意する事項であることは明らかである。

本研究で対象とする手続き型構造化プログラミングの場合、プログラムを作成する前に、変数、代入・演算などの操作、順接・分岐・反復の動作順序などに関する概念について正しく理解する必要がある。これらの基本的な概念を習得した後にプログラムを作成する段階に進むことが学習のプロセスであると

考えている⁽²⁾。

この学習プロセスに習い、筆者らはプログラムの基本的な概念を理解させるために、プログラムの動作を追う、すなわちトレーシングを利用した教育を、ビジュアルプログラミング環境を用いて授業の早い段階で実施している。この結果、誤答した学習者に一定の誤答傾向が見られるのではないかと考えた。

そこで、本稿では、トレーシング課題を用いて、誤答傾向を把握するための分析方法を提案し、分析結果から誤答傾向を見出すことを目的とする。

誤答傾向を把握するために、この課題を実施した学習者の中間試験の結果を用いて、誤答箇所を抽出し、抽出結果から誤答パターンと考えられる7つの項目に分類し、因子分析を行った。さらに、プログラムの動作理解が不十分な学習者を早期に発見する

ため、1年次と再履修者を分けて分析した。

2. 関連研究

関谷ら⁽³⁾はプログラムの変数の値の変化を追う問題（トレーシング能力）に注目し、初心者が起こしやすい誤った解釈に基づいて、与えられたプログラムから誤った出力を得る仕組みを作成している。この仕組みを用いて、学生に与えたごく簡単トレーシング課題の解答から、プログラミング初学者の誤答パターンを見つけた。さらに、トレーシング能力と成績と期末試験の結果を比較・分析し、トレーシング試験の結果と期末試験の傾向に関係があり、特にfor文を解釈しない誤答のパターンに一致する学生の成績が良くない傾向があることを明らかにしている。先行研究は、繰り返しと分岐を様々な組み合わせたソースコードを用いた設問を様々な授業の学生に解答させ、似たような構造をもつソースコードで共通に見られる誤った解釈を誤答パターンとし、この誤答パターンとの学生の成績の関係性を確認したものである。

先行研究とは、誤答の傾向を把握する点において本研究と共通しているが、本研究では、トレーシングの誤答パターン抽出のための試験を実施せずに、過去に使用された課題や試験の結果から、誤答箇所の抽出することで、誤答パターンを類推し、因子分析を用いた誤答傾向の把握をしている点において先行研究は異なる。また、この手法を確立することで、トレーシング課題の解答から、即座に各授業やクラスでの誤答傾向を把握するためのシステム構築を目

指しているとともに、それに応じた指導方法を考案することを目指している点が先行研究とは異なる。

3. 課題の形式と利用環境⁽⁴⁾

本研究で使用した課題形式は、ビジュアルプログラミング環境 AT⁽⁵⁾を用いた。提示方法は、プログラムとその動作を学習者が記入するための表を並べて提示している。表の各行と各列は、プログラムの各行と各変数にそれぞれ対応しており、プログラムの各行の実行結果を、表の同じ行の対応する変数の列に記入する。また、同じ行が複数回実行される場合があるため、実行される回数に応じて右方向に同様の表を並べている。ある行が反復して実行された場合には、左隣の表へ移動し結果を記入することで、プログラムの全ての動作を記録することができる。実行結果として表に記入する項目は、以下の4つである。

- ・変数の値の変化
- ・条件判断
- ・入力、出力された値
- ・代入された値

変数が宣言されたときや変数の値に変化が生じたときには、変数の値を表の対応する列に記入する。変数の値が不定であるときには、不定である旨を記入する（後述する例では「-」で表す）。また、条件判断の結果についてはその真偽を、表の対応する列のセルの色を、真なら青、偽なら赤で表現する。セルの色を変更した同じセルに変数の値の変化も記入することができる。ただし、条件判断が $a < b$ の



図1 AT上で実装したトレース課題

ように複数の変数間で行われている場合には、左辺の変数に対応する列に記入する。変数の値を出力した場合にはその値を、表の対応する列に記入する。本研究では、この形式の課題で提示するプログラムについて、以下のような制約を課している。この制約によって、1行で複数の変数の値が変化することや複数の条件判断が行われることがなく、表の記入に対する混乱を小さくしている。

- 1行に複数の文を含めない
- 1行で複数の代入を行わない
- 計算や条件判断は単独の二項演算のみで行う

なお、授業ではこの課題形式を「トレース課題」と提示していることから、今後、この課題形式をトレース課題と記す。

4. 評価対象者と評価用データ

4.1 対象授業と対象者

対象授業は、尚美学園大学芸術情報学部情報表現学科の1年次秋学期配当の必修科目「プログラミング初級演習1（以下「初級演習1」）」である。この授業は1回90×2コマの演習形式の授業で、対象者はこの授業の2015年度の履修者（再履修者を含む）である。初級演習1では、授業の初段階でATを利用し、基本的なプログラムの動作（振る舞い）を学習し、中期以降、C言語を用いたプログラムを作成させる授業を行っている。

トレース課題は反復を学習する授業で使用した。授業では、反復の概念を説明し、トレース課題の記入方法の説明を行った後、課題を提示している。課題は授業全体で7問提示した。なお、学習者は、トレース課題での演習の前までに、ATを利用して操作、順接、分岐の講義と演習を行っている。

4.2 対象データ

ATを利用した授業終了後、中間試験を実施した。この試験の中でトレース課題を2問提示した。なお、試験は、履修者全員が一斉に試験を受験できる環境が整わなかったため、同日に、Aグループ、Bグループの2グループに分けて実施している。Aグループの受験者は119名、Bグループは83名である。また、別時間実施のため、A、Bグループに同じ問題を

を提示できないことから、両グループで別の問題を提示している。但し、プログラムの構成は同じとし、値を変更して提示している。提示した問題のプログラム構成は下記の通りである。

（問題1）入力、反復、算術演算（乗算）、出力

（問題2）入力、反復、分岐、算術演算（加算）、出力

両グループのプログラム違いは、算術演算式で用いた「即値」の値と、入力値のみである。

（問題1）のプログラムをC言語の表記で下記に示す。Aグループに提示した問題を「A-1」と呼び、

Bグループに提示した問題を「B-1」と呼ぶ。

（A-1）入力値は「5」とする。

```
int k;
int n;
int i;
k = 0;
scanf("%d",&n);
for(i = 1; i <= n; i++){
    k = i * 7;
    printf("%d¥n",k);
}
```

（B-1）入力値は「4」とする。

```
int k;
int n;
int i;
k = 0;
scanf("%d",&n);
for(i = 1; i <= n; i++){
    k = i * 5;
    printf("%d¥n",k);
}
```

（問題2）のプログラムを下記に示す。なお、C言語の表記で示す。Aグループに提示した問題を「A-2」と呼び、Bグループに提示した問題を「B-2」と呼ぶ。

（A-2）入力値は「5」とする。

```
int k;
int n;
int a;
int i;
k = 0;
scanf("%d",&n);
for(i = 1; i <= n; i++){
    a = i % 3;
```

```

    if(a != 0){
        k = k + 1
    }
}
printf("%d\n",k);

```

(B-2) 入力値は「4」とする。

```

int k;
int n;
int a;
int i;
k = 0;
scanf("%d",&n);
for(i = 1; i <= n; i++){
    a = i % 2;
    if(a != 0){
        k = k + 1
    }
}
printf("%d\n",k);

```

5. 評価手順

5.1 誤答箇所の確認

中間試験で提示したトレース課題で、誤答した箇所を、A-1、A-2、B-1、B-2 それぞれの問題に対し、下記の7項目で抽出した。

- (1) 解答に超過しているセル：正解では空だが、解答では値が書かれている
- (2) 解答に不足しているセル：正解では値が書かれているが、解答では空

- (3) 解答の値が誤っているセル：正解と同じ箇所だが、値が誤っている
- (4) 真の超過セル：正解では真になっていないが、解答では真になっている
- (5) 真の不足セル：正解では真になっているが、解答では真になっていない
- (6) 偽の超過セル：正解では偽になっていないが、解答では偽になっている
- (7) 偽の不足セル：正解では偽になっているが、解答では偽になっていない

上記の抽出結果の一部を表1に示す。

表の各列は誤答箇所の項目、各行は対象者となる。「[]」が記入されているセルは、その列の誤答箇所の項目に該当していないことを示している。No.3の項目(1)にある「[0][k][3]」は、「[]」の左から順に図1にある左表の繰り返しの回数-1の値([0])、変数名([k])、先頭行を0とした場合の行数([3])を表している。この例の場合、解答に超過しているセルが、繰り返し1回目の変数kの4行目にあることを示している。また、1つのセルに該当する誤りが複数箇所ある場合、誤答した箇所すべてが抽出される。取得した結果からは、一見すると一般的な分析手順が見当たらない。しかし、誤答パターンとして用意した項目に分け、因子分析を用いることで、誤答傾向を見ることが可能であるとともに、分析結果からいくつか興味深い因子を見つけることができた。

表1 誤答箇所を抽出した結果 (一部抜粋)

No.	(1)解答に超過している	(2)解答に不足している	(3)解答の値が誤っている	(4)真の超過	(5)真の不足	(6)偽の超過	(7)偽の不足
1	[]	[]	[]	[]	[]	[]	[]
2	['[4][k][6]']	[]	[]	[]	[]	[]	[]
3	['[0][k][3]']	['[0][n][3]']	[]	[]	[]	[]	[]
4	['[4][n][4]']	['[4][i][4]']	[]	[]	[]	['[4][n][4]']	['[4][i][4]']
5	['[4][k][6]', '[1][k][4]', '[0][k][4]', '[2][k][4]', '[4][k][4]']	['[2][k][6]', '[2][i][4]', '[0][k][6]', '[1][k][6]', '[0][i][4]']	['[1][k][5]', '[2][k][5]', '[3][k][5]']	['[0][k][4]']	['[0][i][4]', '[1][i][4]', '[2][i][4]', '[3][i][4]']	['[4][k][4]']	['[4][i][4]']

5.2 因子分析を用いた誤答傾向の確認

5.1 の抽出結果と、教授者の経験則から、誤答と考えられる項目を下記の7つに分類し、該当した場合1、そうでない場合0のダミー変数を用いて因子分析を行った。

- (a) Verify truth : 真偽について誤りがある
- (b) Understanding rule : 変数 k, n, i のすべてに対し超過がある
- (c) Understanding command : いずれかの変数について、誤った値が書かれている
- (d) Understanding constant value : 変数 n について、過不足・誤りがある
- (e) Understanding loop : 変数 i について、過不足・誤りがある
- (f) Careless for [i] : 繰り返し 1,2 回目の変数 i について過不足・誤りがなく、それ以外で変数 i について過不足・誤りがある
- (g-1) Careless for [k] : 繰り返し 1, 2 回目の変数 k について過不足・誤りがなく、それ以外で変数 k について過不足・誤りがある (問題 1 のみ)
- (g-2) Careless for [k] and [a] : 繰り返し 1, 2 回目の変数 k, a について過不足・誤りがなく、それ以外で変数 k, a について過不足・誤りがある (問題 2 のみ)

因子分析を行う際には、問題 1, 問題 2 とも、A グループ、B グループの提示問題に即値以外の違いがないことから、(A-1)と(B-1)を合わせて分析し、同じく(A-2)と(B-2)を合わせて分析している。

6. 結果と考察

6.1 誤答箇所に関する結果と考察

誤答箇所の分類では、(1)～(3)が値の誤りに

表 2 問題 1, 2 の誤答者数

	問題 1 (N=202)	問題 2 (N=195)
値の誤り	126	70
真偽の誤り	36	97
どちらも誤り	36	86
正解	76	61

関する項目、(4)～(7)が条件判断の誤りに関する項目であることから、それぞれの問題について、値の誤り者数と、条件判断の誤りの誤答者数を確認した()。なお、解答していない学習者は含まない。

6.2 因子分析の結果と考察

本研究では、履修者の中の1年次と再履修者のそれぞれの傾向を把握するため、問題 1, 2 それぞれの問題に対し、1年次と再履修者を分けて因子分析を行った。

6.2.1 問題1の結果の考察

6.2.1.1 1年次の結果と考察

1年次の因子分析の結果と、因子寄与率を表 3, 4 に示す。結果、第 1 因子は寄与率が 22.64%で、「Careless for [i]」と「Verify truth」の項目に高い負荷量が示されていることから、繰り返しの概念理解についての因子と考える。第 2 因子は寄与率が 19.44%で、「Understanding rule」と「Understanding constant value」の項目に高い負荷量が示されていることから、変数の変化する位置の理解についての因子と考えられる。第 3 因子の寄与率は 13.80%で「Understanding loop」の項目

表 3 問題 1 1年次の因子分析結果

(N=161)

	因子 1	因子 2	因子 3
Careless for [i]	0.921	0.013	-0.118
Verify truth	0.739	0.195	0.317
Understanding rule	0.198	0.878	0.155
Understanding constant value	-0.004	0.732	-0.001
Understanding loop	0.027	0.128	0.751
Understanding command	0.100	-0.012	0.462
Careless for [k]	-0.009	0.018	0.221

表 4 問題 1 1年次の因子寄与率

因子 No.	二乗和	寄与率	累積
1	1.44	20.64%	20.64%
2	1.36	19.44%	40.08%
3	0.97	13.80%	53.88%

に高い負荷量が示されていることから、繰り返しの理解についての因子と考えられる。しかし、3つの因子の累積寄与率は 53,88%となり、因子解釈の点からは妥当なもの判断しかねる。

結果、第1因子は寄与率が 22.64%で、「Careless for [i]」と「Verify truth」の項目に高い負荷量が示されていることから、繰り返しの概念理解についての因子と考える。第2因子は寄与率が 19.44%で、「Understanding rule」と「Understanding constant value」の項目に高い負荷量が示されていることから、変数の変化する位置の理解についての因子と考えられる。第3因子の寄与率は 13.80%で「Understanding loop」の項目に高い負荷量が示されていることから、繰り返しの理解についての因子と考えられる。しかし、3つの因子の累積寄与率は 53,88%となり、因子解釈の点からは妥当なもの判断しかねる。

6.2.1.2 再履修者の結果と考察

再履修者（2年次～4年次）の因子分析の結果と、因子寄与率を表5、6に示す。なお、「Understanding rule」と「Understanding constant value」の項目の変数の値が、すべて同じであったことから、この2つの項目を合わせ

表5 問題1 再履修者の因子分析結果

(N=41)

	因子1	因子2	因子3
Understanding loop	0.942	-0.230	-0.226
Understanding command	0.921	0.205	-0.044
Verify truth	0.797	-0.372	0.360
Understanding rule & constant value	-0.058	0.662	-0.166
Careless for [k]	0.058	-0.662	0.166
Careless for [i]	-0.055	-0.285	0.719

表6 問題1 再履修者の因子寄与率

因子 No.	二乗和	寄与率	累積
1	2.38	39.70%	39.70%
2	1.19	19.84%	59.54%
3	0.75	12.57%	72.11%

「Understanding rule & constant value」項目として分析している。結果、第1因子は寄与率が 39.70%で、「Understanding loop」、「Understanding command」、「Verify truth」の項目に高い負荷量が示されている。これらは、繰り返しの動作に関する項目、繰り返しの中にある算術演算に関する項目、繰り返し文にある条件判断に関する項目であることから、繰り返しの概念と動作の理解についての因子と考える。第2因子は寄与率が 19.84%で、「Understanding rule & constant value」項目に高い負荷量が、「Careless for [k]」項目にマイナスの負荷量が示されていることから、変数kを除いた変数の変化する位置の理解についての因子と考えられる。第3因子の寄与率は 12.57%で「Careless for [i]」との項目に高い負荷量が示されていることから、繰り返しの理解についての因子と考えられる。3つの因子の累積寄与率は 72,11%となり、因子解釈の点からは妥当なもの判断できる。

6.2.2 問題2の結果の考察

6.2.2.1 1年次の結果と考察

1年次の問題2の因子分析の結果と、因子寄与率を表7、8に示す。

表7 問題2 1年次の因子分析結果

(N=157)

	因子1	因子2	因子3
Careless for [i]	0.946	-0.078	0.174
Verify truth	0.444	0.664	0.099
Understanding rule	0.326	0.639	0.387
Careless for [k] and [a]	0.160	-0.081	0.728
Understanding command	0.010	0.353	0.381
Understanding constant value	-0.017	0.179	-0.017
Understanding loop	-0.078	0.364	0.017

表8 問題2 1年次の因子寄与率

因子 No.	二乗和	寄与率	累積
1	1.23	17.57%	17.57%
2	1.15	16.45%	34.02%
3	0.86	12.35%	46.37%

結果、第1因子は寄与率が17.15で、「Careless for [i]」項目に高い負荷量が示されていることから、繰り返しの概念理解についての因子と考える。第2因子は寄与率が16.45%で、「Verify truth」と

「Understanding rule」項目に高い負荷量が示されていることから、条件判断の変数の変化の位置の理解についての因子と考える。第3因子は寄与率が12.35%で、「Careless for [k] and [a]」項目に高い負荷量が示されていることから、分岐の中の算術演算、つまり分岐の真偽判定の結果で算術演算が実行されるか否かの理解についての因子と考えられる。しかし、3つの因子の累積寄与率は46.37%のため、因子解釈の点からは妥当なものとは判断しかねる。

6.2.2.2 再履修者の結果と考察

問題2の再履修者の因子分析の結果と、因子寄与率を表9、10に示す。なお、「Understanding constant value」項目の変数の値がすべて「0」であったため、この項目を除いて分析した。

結果、第1因子は寄与率が24.22で、「Verify truth」、「Understanding rule」、「Careless for [k] and [a]」項目に高い負荷量が示されている。これらは、繰り返し文の真偽判定と分岐の真偽判定に

表9 問題2 再履修者の因子分析結果

(N=38)

	因子1	因子2	因子3
Verify truth	0.757	0.123	0.292
Understanding rule	0.637	0.516	-0.055
Careless for [k] and [a]	0.521	0.103	0.001
Understanding command	0.218	0.865	0.146
Understanding loop	0.022	0.105	0.956
Careless for [i]	0.394	0.187	-0.171

表10 問題2 再履修者の因子寄与率

因子 No.	二乗和	寄与率	累積
1	1.45	24.22%	24.22%
2	1.09	18.10%	42.33%
3	1.05	17.55%	59.87%

関する項目、繰り返しの中にある算術演算の結果を代入する変数と算術演算の結果を用いた分岐の変数に関する項目、さらに、変数の変化する位置に関する項目が含まれていることから、繰り返し・分岐双方の概念と動作理解についての因子と考えられる。

第2因子は寄与率が18.10%で、「Understanding command」と「Understanding rule」項目に高い負荷量が示されていることから、算術演算に関係する変数の変化する位置の理解についての因子と考える。第3因子は寄与率が17.55%で、「Understanding loop」項目に比較的高い負荷量が示されていることから、繰り返しの動作理解についての因子と考えられる。3つの因子の累積寄与率は59.87%のため、因子解釈の点からは妥当なものとは判断できる。

6.3 誤答傾向に関する考察

6.3.1 1年次・再履修者の共通した誤答傾向

1年次、再履修者共通して、問題2の第2因子に「Understanding rule」項目が含まれていることから、繰り返しの中に分岐のあるプログラムでも、プログラムの基本動作を理解しており、トレース課題の解答位置の記述ミスや計算ミスといったケアレスミスの誤答傾向があると考えられる。しかし、問題の解答位置を間違える学習者は、変数の変化の位置が把握できていないという可能性も否定できない。このことから、繰り返し文の中にあるプログラムの構成が違い場合の分析を行う必要がある。

6.3.2 1年次の誤答傾向

1年次は、問題1の第1因子に「Careless for [i]」と「Verify truth」項目が分類されているのに対し、問題2では、第1因子に「Careless for [i]」、第2因子に「Verify truth」項目が分類されている。これは、分岐の条件判断と繰り返しの条件判断の誤答傾向は別であることを示していると考えられる。

一方、「Understanding command」項目が、どの因子にも属さない結果となった。この項目に該当する算術演算では、問題1、2とも繰り返し用に用意した変数*i*を計算内で使用している。このことから、変数*i*の値の変化の誤答傾向と相関があると考えられるが、分析結果では、その点について相関が確認できない。このことから、この項目に関しては

再検討が必要であると考え

6.3.3 再履修者の誤答傾向

再履修者については、問題 1、問題 2 とも第 1 因子に「Verify truth」と「Understanding rule」項目が含まれている。これは、繰り返しの真偽判定での誤答と、分岐での真偽判定での誤答の相関が強いということが考えられる。つまり、繰り返しと分岐の真偽判定の誤答が同時に起きる可能性が高いと考えられる。また、問題 1 では、第 2 因子で「Careless for [k]」項目がマイナスの負荷量であったのに対し、問題 2 では、「Careless for [k] and [a]」項目が第 1 因子でプラスの負荷量となっている。これは、繰り返し文と分岐の動作の理解ができていない傾向が示されていると考える。これらのことから、再履修者の特徴として、条件判断の誤答が、プログラムの動作の理解の妨げとなっていると考えられる。

7. まとめと今後の課題

本稿では、プログラミングを学習する際に、学習者がプログラムの動作を思考し、思考したプロセスを自己の中で形成し、形成したプロセスを、実際のプログラムの動作と一致させることが重要であることから、完成したプログラムとその動作を学習者が記入するための表を並べて提示した課題（トレーシング）を、ビジュアルプログラミング環境上で実装し授業で実施した。授業後に実施した中間試験から、トレーシング問題の誤答箇所を抽出し、抽出した誤答から、筆者らが誤答パターンと考える 7 つの項目に分類し、因子分析を用いて誤答傾向を確認した。分析結果の寄与率が、1 年次の問題 1 で 53.88%、問題 2 で 46.37%、再履修者の問題 2 で 59.87%と、高い寄与率とは判断できないものの、誤答傾向を知る上での一つの指針となり得ると考える。

再履修者の誤答傾向からは、関係演算子に関する誤答がプログラムの動作の理解の妨げとなっていると考えられることが確認できた。このことから、1 年次の授業直後にこの分析を実施し、条件判断に多くの誤答がある学習者に対して関係演算子の理解を促す課題を追加するなどといった対応をすることで、分岐だけでなく繰り返しの動作理解が促すことが可

能であると考えられる。

今後は、誤答パターンの項目で、因子分析の結果、因子に寄与しない項目があったことから、誤答パターンの項目を再検討する必要がある。

また、授業内の課題で学習者が提出した解答の中で、最初の提出で誤答があった学習者が正解するまでの過程を分析する予定である。

さらに、本結果を用いて演習時に誤答傾向を即座に確認できるシステムを追加することを検討している。

謝 辞

本研究は JSPS 科研費 26350284, 15K01023 の助成を受けたものです。

参 考 文 献

- (1) IT 総合戦略本部, 世界最先端 IT 国家創造宣言, June 2015, <https://www.kantei.go.jp/jp/singi/it2/kettei/pdf/20150630/siryou1.pdf> (2017 年 2 月 3 日アクセス)
- (2) 文部科学省, 中学校学習指導要領, March 2008. http://www.mext.go.jp/a_menu/shotou/new-cs/youryou/icsFiles/a_eld_le/2015/03/26/13562511.pdf (2017 年 2 月 3 日アクセス)
- (3) 関谷貴之, 山口和紀, 山本三雄: "初学者によるプログラムトレーシングにおける誤答に関する分析", 情報教育シンポジウム 2012 論文集 2012(4), 113-120 (2012)
- (4) 湯沢航太, 國宗永佳, 新村正明: "プログラムの動作理解を表出する課題形式の提案", 信学技法, ET2016-2, pp.49-52 (2016)
- (5) 小林慶, 國宗永佳, 香山瑞恵: "アルゴリズム的思考法教育を支援するビジュアルプログラミング環境の開発", 教育システム情報学会研究報告, vol.27, no.4, pp.3-8 (2012)