

プログラム実行時の変数変化に着目した

プログラミング学習支援ツール

橋浦研究室

1155216 工藤沢海

1. はじめに

文部科学省[1]は 2020 年に初等教育にプログラミングを必修にすることを決定した。このことから、プログラミング教育は今後より重要になると考えられる。

本研究では、プログラミング初学者が、つまり問題の一つである、変数の値の変化の処理が目に見えないという問題に着目した。

2. 研究目的

本研究は、学習者が直接コードから変数の値の変化を読み取ることが出来るようになることを目的とする。具体的には、ツールを用いて、記述したソースコードの変数の流れを実行課程に沿って学習者にトレース表として提示する。これにより、変数値の変化の認識が可能になる。

3. 提案手法

本研究は、対象となるプログラムに、ソートプログラムを用いる。その理由としては、基礎的な構文(変数宣言, if 文, for 文など)を含んでいること、内部の変数値の変化が多くみられることなどが挙げられる。

学習を支援する方法として、ツールの機能は以下の 2 つである。

- (1) ソースコードからトレース表を生成する機能

- (2) トレース表の正誤を判定する機能

(1)はプログラムを実際に行うことにより、トレース表を作成する機能である。この時、コードの行とトレース表の行を一致させ、変数の変化がより分かりやすいようにする。この可視化の例を図 1 に示す。(2)は(1)で作成されたトレース表をそのまま提示するだけでなく、一部を伏せることにより学習者に回答させるようにする。

4. 実験

本研究で作成したツールの有用性を検証するために実験を行った。その実験の手順を以下に示す。

1. 実験の概要を説明する(10 分)
2. ツール使用前のテスト(15 分)
3. ツールを用いての学習を行う(20 分)
4. ツール使用後のテスト(15 分)

テストで出題する問題の範囲は、「2016 年度プログラミング技術・演習」[2]の授業内容の範囲から、変数の変動が多くみられるソートプログラムを対象とする。具体的にはバブルソートと選択ソートの 2 つとし、変数の変化について問う問題を出題する。問題の具体例は図 2 に示した。

ツールによる学習方法は、変数の変化についての問題をトレース表で出題し回答させ

	#	i [0]	data [0]	data [1]	data [2]	data [3]	data [4]
1 #include <stdio.h>	1行						
2	2行						
3 int main(void){	3行						
4	4行						
5 int i = 0;	5行	0	1342...	0	1342...	0	-71856
6 int data[5] = {1,2,3,4,5};	6行	0	1	2	3	4	5
7	7行						
8 for(i = 0; i < 5; i++){	8行	0	1	2	3	4	5
9 data[i] = data[i] + 1;	9行	0	2	2	3	4	5
10 }	10行						
11	11行						
12 return 0;	12行	5	2	3	4	5	6
13 }	13行	5	2	3	4	5	6

図 1 トレース表の可視化例

る. 具体的には, 図 2 の問題があるとして, トレース表で出題する際には, 図 3 のように問題をトレース表上に穴埋め問題として出題できるようにする. その後, 学習者に回答してもらい, その後ツールによって正誤判定を行い, 変数に誤りがないトレース表を完成させる. これにより, 学習者が自ら変数について考える機会を得る.

問 2.6 行目の時の data[5]の値を教えてください
 data[0]=_, data[1]=_, data[2]=_
 data[3]=_, data[4]=_.

図 2 変数の変化を問う問題例

行	i	data[0]	data[1]	data[2]	data[3]	data[4]
6		1	2			
7						

図 3 トレース表上での出題例

5. 評価方法

評価方法は, ツール使用前のテストとツール使用後のテストの点数を比較し, その差異によって評価することとした. テストの評価方法については以下のとおりである.

- ① 記述無し(0点)
- ② 数値の記述があるが誤っている(1点)
- ③ 正解している(2点)

上記の①と②を区別した理由については, 何も分からない状態から少し理解ができたのか, 理解について細かく評価を行うためである.

6. 結果と考察

ツールで学習を行う前に行ったテストを事前テスト(①)とし, ツールで学習した後のテストを事後テスト(②)とする. テスト①, ②の採点結果を表 1 に示す. また, 満点は 142 点である. 表 1 より点数が上昇した被験者 C, F において, 詳細な分析を行った. その結果, 事前テストにおいて for 文内で行う配列の入れ替えにおいて間違いがあったが, 事後テストでは正解していた. このことからツールを用いた学習によって, 配列の代入についての理解ができるようになった可能性がある. 逆に, 点数が大きく下がった被験者 E

においてはコードの読み間違いによって配列の入れ替えが誤っていたが, 変数の入れ替え自体は理解していることが判明した.

表 1 テスト A と B の採点結果

被験者	テスト①	テスト②	テスト(②-①)
A	105	102	-3
B	142	142	0
C	109	142	33
D	142	140	-2
E	142	119	-23
F	92	102	10
平均	122	124.5	2.5

今回の 2 つのテストで変数の初期値を問うものについては, 誰一人として間違えることはなかった. これらの結果から, 能力の向上を測定する手段やツールの学習方法などを改善する必要があると考える. 能力の向上を測定する手段としては, 本学の授業で使われている Visual Studio を用いた学習との比較などがあげられる.

7. まとめと今後の課題

本研究で作成したツールを用いた学習により, 学習者が変数の変化を読み取ることができるようになるのか実験を行った. その結果, 全体としては大きな変化が見られなかった. ただし, 一部の被験者において実行過程の変数の変化を問う問題において点数の上昇がみられた. また, 実験により学習者が変数の値を読み取ることができなくなってしまう個所として if 文, for 文があった. これは, 繰り返しや条件分岐により変数の変化が追いにくくなったためと考えられた. この確認された箇所についてのサポートができるようなツールを用いた学習方法を新たに考える必要がある.

参考文献

- [1] 文部科学省, “「小学校プログラミング教育の手引き(第一版)」の公表について,” http://www.mext.go.jp/b_menu/houdou/30/03/1403192.htm, 2018/3/30, (accessed 2018/6/21).
- [2] 山地秀美, 勝間田仁, 石原次郎, 中村一博, 大橋裕太郎, 橋浦弘明, 松浦隆文, “2016 年度春学期 日本工業大学シラバス プログラミング技術・演習 I,” (accessed 2018/11/22).