

令和 2 年 6 月 9 日現在

機関番号：53203

研究種目：基盤研究(C) (一般)

研究期間：2016～2019

課題番号：16K01153

研究課題名(和文) アルゴリズム設計能力育成のためのブレンド型導入教育に関する研究

研究課題名(英文) A Study of Blended Introductory Education for Fostering Algorithm Design Ability

研究代表者

新開 純子 (SHINKAI, Junko)

富山高等専門学校・電子情報工学科・嘱託教授

研究者番号：60179067

交付決定額(研究期間全体)：(直接経費) 3,400,000円

研究成果の概要(和文)：プログラミング教育の初期段階での戸惑いや難しさを感じさせないために、プログラミング教育の前段階として、日本語の手順書を書くプログラミング導入教育を3年間実施した。実施後のアンケート調査の結果、あくまでも学生の主観であるが、プログラミング導入教育は楽しく、あいまいさのない処理手順を書くことの必要性を感じていることがわかった。さらに、プログラミング教育の初期段階の試験の成績は、プログラミング導入教育を経験した学生の方が高いことがわかった。

研究成果の学術的意義や社会的意義

プログラミング教育の目的は、プログラム言語で記述可能なアルゴリズムを作成する能力とプログラム言語で表現する能力を育成することである。しかし、プログラミング教育の初期段階で戸惑いや、難しさを感じ学習意欲を失っている学生が少なくない。そこで、筆者らは、人に対して指示する処理手順の表現とコンピュータに対して指示する処理手順の表現の違いが戸惑いの原因の1つであると考え、日本語の手順書を作成するプログラミング導入教育を提案・実践した。

研究成果の概要(英文)： Introductory programming education to write procedure manuals in Japanese was practiced for three years for the students before programming learning so that they might not feel confusion and difficulty at an early stage of programming education. Results of a questionnaire survey administered after the introductory programming education, although assessing only subjectivity of the students, showed that introductory programming education is fun for them. Moreover, they feel the necessity of writing unambiguous procedure manuals. Moreover, regarding the results of examinations administered at an early stage of programming education, the students who had experienced the introductory programming education earned higher scores.

研究分野：教育工学

キーワード：プログラミング教育 導入教育 アルゴリズム構築 論理的思考

1. 研究開始当初の背景

情報系学科のプログラミング教育の目的は、プログラム言語で記述可能なアルゴリズムを作成する能力(アルゴリズム作成能力)とプログラム言語で表現する能力(プログラム作成能力)を育成することである。筆者らは特にアルゴリズム作成能力を育成するために、アルゴリズム作成までのプロセスを重視し、さらにeラーニング、アルゴリズム作成支援システム、相互評価活動などをブレンドしたプログラミング教育を実践した。その結果、eラーニングはプログラミングの知識の定着率を高めることがわかった。さらに、アルゴリズム作成までのプロセスを重視することはアルゴリズム作成能力育成に有効であることがわかった。

しかし、プログラミング教育の初期段階で戸惑いや難しさを感じて学習意欲を失う学生が少なくなかった。そのため、論理的に考え、アルゴリズムを作成するプログラミング教育へのハードルを低くして、初期段階のプログラミング教育への戸惑いや難しさを感じさせないようにするためのプログラミング導入教育が必要である。

2. 研究の目的

本研究の目的は、プログラミング教育の初期段階での戸惑いや難しさを感じさせないプログラミング導入教育を提案・実践して、その教育効果を評価することである。筆者らはコンピュータに対して変数や代入などを使って指示する処理手順(アルゴリズム)の表現と、人に対して指示する処理手順の表現の違いが、プログラミング教育の戸惑いや難しさの要因の1つであると考えた。

そこで、コンピュータに指示する処理手順の表現と人に指示する処理手順の表現の違いを学生に理解してもらうために、論理的に曖昧さのない日本語の処理手順を書くプログラミング導入教育を提案する。2つの処理手順の表現の違いを認識させることにより、プログラミングの初期で教える変数や代入という概念を理解させ、プログラム言語で記述可能なアルゴリズムを作成するプログラミング教育へのハードルを低くする。

3. 研究の方法

プログラミング導入教育では、学生に次の4つの経験をさせる必要があると考える。

1. システムを開発する立場で考える
2. 問題解決方法が1つではないことを知る
3. 曖昧さのない日本語の処理手順を書く
4. プログラム言語で表現・実行・確認する

経験1では、プログラミングはシステムを使う立場ではなく、システムを開発する立場で考える学習であることを学生に意識させる。学生には自動販売機システムを開発する立場で購入者の行動と自動販売機システムの処理を時系列に箇条書きする作業を行わせる。

経験2では、問題解決の方法は1つではないことを学生に経験させる。さらに、解決方法の効率や分かりやすさを意識させる。この経験では、8枚のコインの中から1枚だけ重さの軽い偽物のコインを、天秤を使ってみつけるパズルを教材にする。学生には、天秤の使った回数を少なくする方法を考えさせる。学生に偽物のコインをみつける方法と天秤の使用回数を発表させる。学生に問題解決の方法は1つではないことや、効率や分かりやすさも重要であることを説明する。

経験3では、わかりやすい処理手順を作成するために、0~9までの数字が書かれたカードを準備して、カードを使って問題解決のための手順を考えさせる。その後、人に対する処理手順を書かせる。学生が書いた処理手順にもとづき、人に対する処理手順とコンピュータに対する処理手順の違いを説明する。その後、人に対する処理手順をコンピュータで処理するための処理手順に変更する作業を学生に行わせる。このとき、処理手順は下記のルールで書くよう指示する。さらに、図1の処理手順の例を学生に示す。

- カードを置く場所に名前(英字で始める6文字以内の英数字の組合せ)をつける
- 処理手順を箇条書きにする
- 主語と述語を明確に書く
- 曖昧さのない明確な表現で書く

経験3の課題を表1に示す。日本語の処理手順作成の課題は、各年度の学生の要望や進捗状況により表1の中から選ぶ。

表1 日本語の処理手順作成課題

○ カードを置く場所をa1, b1, b2とする
・ 数値3のカードを場所a1に置く
▲ もし、場所a1のカードの数値が5以下ならば
・ 場所b1に数値0のカードを置く
そうでないならば
・ 場所b2に数値1のカードを置く

No.	課題の内容
1	8枚のコインを1枚ずつ天秤に置き、重さが軽い偽物のコインをみつける
2	分母が1桁の正整数で、分子が1の分数の値を小数点以下3桁まで筆算で求める
3	2桁の正整数の足し算を筆算で求める

図1 日本語の処理手順例

経験4では、コンピュータで問題を解決するためのプログラム言語表現を学生に体験させる。経験3で作成した処理手順に従って、Cプログラムを作成・実行をして、意図した結果が得られたかを確認させる。このとき、Cプログラムの文法は教えずに、サンプルプログラムを提示して、そのプログラムをまねて完成させるようする。

4. 研究成果

プログラミング導入教育の評価を行うために、アンケート調査や試験の得点の比較を行った。アンケートの回答は5段階(5.思う, 4.やや思う, 3.どちらとも言えない, 2.やや思わない, 1.思わない)で評価させた。各表の m , SD はそれぞれ平均, 標準偏差である。さらに, t 値と結果は「3.どちらとも言えない」に対して平均評定値が肯定側ないし否定側に偏っているかを t 検定した結果である。

(1) プログラミング導入教育の評価

プログラミング導入教育を3年間(2016年度40人, 2017年度43人, 2018年度44人)実施した。実施後のプログラミング導入教育の評価アンケート調査結果を表2に示す。3年間とも17項目中16項目が肯定的に有意に偏っていた。項目12の「プログラミング言語で表現することは簡単であった」は、3年間とも有意差が認められなかった。

あくまでも学生の主観によるものであるが、表2の結果より、次のことがわかった。

項目1, 2, 3は3年間とも肯定的に有意に偏っている。これにより、学生はプログラミング導入教育を良い経験であり、楽しいと思っていることがわかった。

項目4, 5, 6は3年間とも肯定的に有意に偏っている。これにより、学生が経験2で実施したパズルや経験3で他の人の処理手順を確認することは、論理的思考を鍛えると思っ

ていることがわかった。項目8, 9, 10, 11は3年間とも肯定的に有意に偏っている。これにより、学生がプログラミングにおけるアルゴリズムの重要性を認識し、論理的に処理手順を考え、曖昧さがないように表現することが重要であると思っ

ていることがわかった。項目14, 15, 16, 17は3年間とも肯定的に有意であった。これにより、学生がプログラムの完成時には達成感があり、プログラミングへの学習意欲が向上していることがわかった。

項目12だけが3年間とも有意差が認められなかった。Cプログラムのサンプルをまねる形でプログラム作成を行ったが、学生にとってはまだまだハードルが高かったといえる。

表2 プログラミング導入教育に関する評価アンケート

No.	評価項目	導入教育											
		2016年度				2017年度				2018年度			
		m	SD	有意差検定		m	SD	有意差検定		m	SD	有意差検定	
		t	結果			t	結果			t	結果		
1	プログラミング導入教育は良い経験になった	4.55	0.55	17.75	**	4.14	0.89	8.41	**	4.59	0.82	12.93	**
2	プログラミングの導入教育は楽しかった	4.25	0.74	10.65	**	3.91	0.89	6.65	**	4.07	1.07	6.65	**
3	問題解決のための処理手順を考えることは楽しい	4.13	0.79	9.00	**	4.00	0.82	8.03	**	3.84	1.1	5.08	**
4	パズルは論理的思考を鍛える	4.45	0.64	14.36	**	4.16	0.72	10.57	**	4.23	1.1	7.42	**
5	互いに教えあうことで論理的に考える力が向上する	4.38	0.67	13.03	**	3.98	0.96	6.65	**	4.34	0.86	10.33	**
6	他の人の手順書を確認することは論理的思考を鍛える	4.15	0.66	10.98	**	4.00	0.76	8.67	**	4.11	1.04	7.11	**
7	他の人の手順書を確認することは良い経験になった	4.23	0.62	12.50	**	4.16	0.72	10.57	**	4.39	0.89	10.28	**
8	プログラミングには論理的に手順を考えることが大切である	4.45	0.71	12.84	**	4.26	0.69	11.88	**	4.34	0.86	10.33	**
9	曖昧さがないように手順を書く必要があることがわかった	4.55	0.64	15.35	**	4.33	0.68	12.78	**	4.77	0.52	22.52	**
10	筋道をたてて考えることは大切である	4.35	0.74	11.61	**	4.28	0.77	10.95	**	4.48	0.66	14.75	**
11	アルゴリズム(処理手順)を考えることは重要である	4.40	0.78	11.38	**	4.33	0.68	12.78	**	4.52	0.73	13.82	**
12	プログラミング言語で表現することは簡単である	3.23	1.14	1.24		3.16	1.04	1.02		2.84	1.22	0.87	
13	プログラミング言語はルールに従えば書ける	3.93	1.02	5.72	**	3.79	0.91	5.67	**	3.86	1.03	5.59	**
14	プログラムが思い通りに動いたときは達成感がある	4.58	0.64	15.66	**	4.47	0.70	13.69	**	4.68	0.56	19.88	**
15	C言語の文法をもっと知りたい	4.30	0.79	10.39	**	4.14	0.86	8.67	**	4.27	0.9	9.39	**
16	プログラミングをもっと学びたい	4.38	0.67	13.03	**	4.26	0.79	10.43	**	4.5	0.79	12.56	**
17	プログラミングの授業で論理的考え方が身につく	4.20	0.85	8.89	**	4.26	0.79	10.43	**	4	1.01	6.56	**

** : $p < .01$, * : $p < .05$

(2) 処理手順作成課題の評価

経験3で行った日本語の処理手順作成課題の評価結果を表3に示す。2016年度に実施した課題3は、評価項目「処理手順を書くのは簡単である」に対して否定側に偏っている。学生にとっては2桁の正整数の足し算を手計算で求める処理手順の作成は、難しい課題であったといえる。

2017年度は、2016年度の結果をふまえ、課題2の分数の小数表現の処理手順の作成を先に実施し、その後、課題3を実施した。その結果、2つの課題とも、処理手順を書くのは簡単であると学生が思っていることがわかった。

また、2018年度は、課題1, 課題2, 課題3の順に実施したが、実施した順に簡単な課題であると学生が思っていることがわかった。これにより、学生にとって処理手順を考えやすい課題1から課題2, 課題3の順に処理手順作成を実施したといえる。

表 3 日本語の手順作成課題の評価

No.	課題	課題の評価項目	2016年度				2017年度				2018年度			
			m	SD	t値	結果	m	SD	t値	結果	m	SD	t値	結果
1	重さの違う コインを見 つける	良い課題である	-	-	-	-	-	-	-	-	4.43	0.82	11.61	**
		処理手順を書くのは簡単である	-	-	-	-	-	-	-	-	3.82	0.99	5.46	**
2	分数の 小数化	良い課題である	-	-	-	-	3.91	0.95	6.28	**	4.39	0.89	10.28	**
		処理手順を書くのは簡単である	-	-	-	-	3.79	0.91	5.67	**	3.75	1.06	4.70	**
3	2桁の 足し算	良い課題である	3.70	0.76	5.84	**	4.16	0.69	11.09	**	4.16	0.96	7.98	**
		処理手順を書くのは簡単である	2.65	0.83	2.66	**	3.79	0.97	5.37	**	3.39	1.20	2.13	*

** : $p < .01$, * : $p < .05$

(3) プログラミング導入教育の有無による試験の成績比較

2016年度入学生から1年次にプログラミング導入教育を実践した。2年次で実施するCプログラミング教育(前学期90分×15回,後学期90分×15回)の初期段階への効果を調べるために,前学期試験に同じ問題を出題して,試験の成績を比較した。表4に試験問題の内容を示す。2015年度入学生から2018年度入学生までの2年次で実施するCプログラミング前学期試験の各問の平均得点について,分散分析を行った。その結果,14問中9問が有意であった。有意であった問題についてScheffe法による多重比較を行った結果を表5に示す。

2015年度入学生(56人)は,プログラミング導入教育の未経験者である。2016年度入学生(40人),2017年度入学生(43人),2018年度入学生(47人)は,プログラミング導入教育を経験後,2年次のCプログラミング教育を受講した。

試験の合計得点の平均mは,プログラミング導入教育未経験の学生に比べ,プログラミング導入教育を経験した3年間の学生がすべて高いことがわかった。さらに,標準偏差SDは低く,得点のばらつきが小さいことがわかった。合計得点の平均の分散分析の結果,有意であった。多重比較の結果,プログラミング導入教育未経験の2015年度入学生とプログラミング導入教育経験ありの2016年度,2017年度入学生の間には有意差が認められなかった。しかし,プログラミング導入教育未経験の2015年度入学生とプログラミング導入教育経験ありの2018年度入学生の間,有意水準5%で有意差が認められた。また,プログラミング導入教育を経験している2017年度と2018年度の間にも有意差が認められた。これにより,2018年度にプログラミング導入教育を経験した学生は,プログラミングの授業の理解度が高いといえる。

さらに,多重比較の結果より,問1,問3,問4のC言語の知識を問う問題や問6,問7,問10,問11のプログラム作成やプログラムの流れを確認する問題について,プログラミング導入教育未経験の2015年度入学生とプログラミング導入教育経験ありの2018年度入学生の間,有意水準5%で有意差が認められた。これにより,プログラミング導入教育経験ありの2018年度入学生は,プログラミング教育の初期段階の変数や入出力,関数の書き方などの知識や分岐,反復の基本的プログラム作成の能力が高いといえる。

また,問6と問11のプログラムの流れを追跡する問題は,プログラミング導入教育未経験の2015年度入学生とプログラミング導入教育経験ありの2016年度,2018年度入学生の間,それぞれ有意水準5%で有意差が認められた。これにより,プログラミング導入教育は,プログラムの流れを追跡する能力を高めているといえる。

表 4 試験問題の内容

区分	問題	問題内容
C言語の知識	問1	変数の名前の付け方や型宣言など
	問2	算術演算子,インクリメント,デクリメントなど
	問3	入出力命令の書き方など
	問4	関数の引数,戻り値などの書き方など
プログラム	問5	2つの変数の値を交換するプログラムを作成せよ。
	問6	main関数で関数fun()を呼び出し,処理を行う。プログラム実行によりprintf関数で出力される変数の値を順に書きなさい。
	問7	「こんにちは」と出力する関数hello()と,関数を呼び出すmain関数のプログラムの空欄を埋めよ。
	問8	入力データが '+' ならば,2つの整数の和を出力し, '-' ならば差を出力する。その他の場合は, "Bad Character" と出力するプログラムの空欄を埋めよ。
	問9	整数nの絶対値を戻り値として返す関数abs_fun()のプログラムを作成せよ。ただし, nは仮引数とする。
	問10	整数nからmまでの奇数の積を戻り値として返す関数fun()のプログラムを作成せよ。ただし, n, mは仮引数とする。
	問11	プログラムのfor文の2重ループ処理中に出力される変数の値と,2重ループ終了後に出力される変数の値を順に書きなさい。
	問12	$1+(1+2)+\dots+(1+2+\dots+10)$ の値を求めるアルゴリズムの空欄を埋めよ。【応用】
	問13	2つの自然数a, bの最大公約数を戻り値として返す関数gcd_fun()のプログラムを作成せよ。ただし, a, bは仮引数とする。
	問14	正整数n (n>1)の素数判定を行うアルゴリズムの空欄を埋めよ【応用】

表5 プログラミング導入教育の有無による試験の成績比較

区分	問題	配点	プログラミング導入教育								4年間の分散分析	Scheffe法による多重比較						
			経験なし		経験あり							F値	2015 vs. 2016	2015 vs. 2017	2015 vs. 2018	2016 vs. 2017	2016 vs. 2018	2017 vs. 2018
			2015年度入学生		2016年度入学生		2017年度入学生		2018年度入学生									
			m	SD	m	SD	m	SD	m	SD								
C言語の知識	問1	20	16.14	3.34	16.80	3.32	16.60	3.28	18.72	2.06	6.76 **			*		*	*	
	問2	24	15.21	5.35	15.08	5.74	14.16	5.33	17.60	4.09	3.71 *						*	
	問3	16	7.13	5.18	9.45	5.02	9.72	4.42	12.47	3.44	11.59 **			*		*	*	
	問4	16	8.16	5.44	8.70	3.97	8.47	4.72	10.94	4.24	3.50 *			*				
	小計	76	46.64	15.21	50.03	13.79	48.95	13.76	59.72	9.88	9.03 **			*		*	*	
プログラム	問5	10	8.50	2.23	8.70	2.05	9.12	2.06	9.04	1.79	0.98							
	問6	5	0.98	2.00	2.50	2.53	1.51	2.32	3.19	2.43	9.08 **	*		*			*	
	問7	9	5.89	3.28	7.50	2.63	7.40	2.95	7.91	1.70	5.47 **	*		*				
	問8	12	8.55	4.02	8.78	3.42	8.51	3.70	9.38	3.10	0.59							
	問9	15	10.02	5.91	11.80	4.75	10.88	5.56	12.17	5.03	1.62							
	問10	15	7.05	6.24	7.45	5.87	7.88	5.92	10.64	5.21	3.71 *			*				
	問11	4	0.82	1.47	2.25	1.77	1.33	1.52	3.11	1.55	20.54 **	*		*			*	
	問12	21	13.93	5.96	13.50	7.29	14.79	6.91	16.66	5.93	2.16							
	問13	15	9.61	5.49	10.63	5.44	9.74	5.64	11.36	4.79	1.13							
	問14	18	14.23	6.29	11.63	5.77	12.63	5.98	15.00	4.76	3.11 *							
小計	124	79.59	31.88	84.73	30.60	83.79	30.64	98.47	25.98	3.67 *			*					
合計	200	126.23	44.67	134.75	41.67	132.74	41.37	158.19	34.02	5.71 **			*			*		

** : $p < .01$, * : $p < .05$

(4) プログラミング導入教育の有益度

2016年度, 2017年度に入学し, プログラミング導入教育を受講し, その後プログラミング教育を修了した学生75人(2016年度入学生35人, 2017年度入学生41人)を対象に, プログラミング導入教育がプログラミングの初期段階の難しさや戸惑いを解消するための導入教育として有効かどうかについてアンケート調査を行った結果を表6に示す. 表6はt値の高い順に並べた. あくまでも学生の主観によるものであるが, アンケートの6項目すべてが, 肯定的に有意であった. 特に日本語の処理手順作成は, プログラミングのアルゴリズム作成の導入になっていると学生が思っていることがわかった. また 曖昧さのない表現で書くこと, 簡条書きにすること, 主語と述語を明確にすることの処理手順を書くルールは, プログラミングへの導入になっていると学生が思っていることがわかった. さらに, カードを置く場所に名前を付け, 「カードを置く」などのように処理手順を書かせたことは, プログラミングの変数や代入の導入になっていると学生が思っていることがわかった.

表6 プログラミング導入教育の有益度

No.	評価項目	m	SD	t 値	結果
6	手順書作成はアルゴリズム作成の導入に有効	3.79	1.04	6.53	**
5	曖昧さのない表現で書くことはプログラミングへの導入に有効	3.68	1.08	5.45	**
3	簡条書きにすることは, プログラミングへの導入に有効	3.55	1.07	4.43	**
4	主語と述語を明確にすることは, プログラミングへの導入に有効	3.45	1.07	3.67	**
1	カードを置く場所は, プログラムの変数への導入に有効	3.41	1.01	3.47	**
2	カードを使うことは, プログラムの代入の導入に有効	3.33	1.04	2.76	**

** : $p < .01$

プログラミング教育の初期段階で, 学生に戸惑いや難しさを感じさせないために, 曖昧さのない日本語の処理手順を書くプログラミング導入教育を3年間実施した. 3年間のプログラミング導入教育へのアンケート調査の結果, あくまでも学生の主観であるが, プログラミング導入教育は楽しく, 曖昧さのない処理手順を書くことの必要性を感じていることがわかった. また, 2018年に実施した処理手順作成は, 学生にとって考えやすい課題から順に実施したことがわかった.

さらに, Cプログラミングの初期段階の試験におけるプログラミング導入教育経験の有無による試験の成績比較により, プログラミング導入教育を経験した学生の得点が高くなったことがわかった. また, あくまでも学生の主観であるが, プログラミング導入教育で行った日本語の処理手順作成は, プログラミングへの導入に有益であると肯定的に思っていることがわかった.

これらの結果より, 日本語の処理手順を書かせるプログラミング導入教育は, Cプログラミング教育の初期段階での戸惑いや難しさを軽減する有効な教育であるといえる.

今後は, 学生にとって考えやすく, 基本制御構造である順次, 選択, 反復を意識させる課題をさらに考えたい.

5. 主な発表論文等

〔雑誌論文〕 計5件（うち査読付論文 3件／うち国際共著 0件／うちオープンアクセス 0件）

1. 著者名 Shinkai, J., Hayase, Y.	4. 巻 -
2. 論文標題 THREE-YEAR PRACTICE AND EVALUATION OF INTRODUCTORY PROGRAMMING EDUCATION	5. 発行年 2019年
3. 雑誌名 Proceedings of 12th annual International Conference of Education, Research and Innovation, ICERI2019	6. 最初と最後の頁 pp.3517-3522
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 新開純子, 早勢欣和	4. 巻 JSET19-4
2. 論文標題 プログラミング導入教育の実践と評価	5. 発行年 2019年
3. 雑誌名 日本教育工学会研究報告集	6. 最初と最後の頁 pp.77-84
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 無
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 Shinkai, J., Hayase, Y.	4. 巻 -
2. 論文標題 IMPROVEMENT AND EVALUATION OF INTRODUCTORY PROGRAMMING EDUCATION	5. 発行年 2018年
3. 雑誌名 Proceedings of 11th annual International Conference of Education, Research and Innovation, ICERI2018	6. 最初と最後の頁 pp.5825-5830
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 Shinkai, J., Hayase, Y., Miyaji, I., Moananu, C	4. 巻 -
2. 論文標題 PROPOSAL AND TRIAL OF INTRODUCTORY PROGRAMMING EDUCATION	5. 発行年 2018年
3. 雑誌名 Proceedings of International Technology, Education and Development Conference, INTED2018,	6. 最初と最後の頁 pp.9668-9673
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 新開純子, 早勢欣和, 宮地功	4. 巻 vol.31, no.6
2. 論文標題 シナリオ作成を用いたプログラミング導入教育の試み	5. 発行年 2017年
3. 雑誌名 教育システム情報学会研究報告	6. 最初と最後の頁 pp.1-6
掲載論文のDOI (デジタルオブジェクト識別子) なし	査読の有無 無
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計3件 (うち招待講演 0件 / うち国際学会 0件)

1. 発表者名 新開純子, 早勢欣和
2. 発表標題 学生群によるプログラミングの難しさの違い
3. 学会等名 日本教育工学会2019年度秋季全国大会講演論文集
4. 発表年 2019年

1. 発表者名 新開純子, 早勢欣和, 宮地功
2. 発表標題 プログラミング導入教育の改善について
3. 学会等名 日本教育工学会第34回全国大会講演論文集
4. 発表年 2018年

1. 発表者名 新開純子, 早勢欣和, 宮地功
2. 発表標題 シナリオ作成を用いたプログラミング導入教育の試みとその評価
3. 学会等名 日本教育工学会第33回全国大会講演論文集
4. 発表年 2017年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

-

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
研究 分担者	早勢 欣和 (HAYASE Yoshikazu) (60238144)	富山高等専門学校・電子情報工学科・准教授 (53203)	
研究 分担者	宮地 功 (MIYAJI Isao) (30043722)	富山高等専門学校・その他部局等・特命フェロー（教育・研究支援） (53203)	