

令和 5 年 5 月 23 日現在

機関番号：57403

研究種目：基盤研究(C) (一般)

研究期間：2020～2022

課題番号：20K03265

研究課題名(和文) ラーニングアナリティクスを活用したオブジェクト指向プログラミング教育支援システム

研究課題名(英文) Object-oriented programming education support system utilizing learning analytics

研究代表者

村田 美友紀 (Murata, Miyuki)

熊本高等専門学校・拠点化プロジェクト系情報セキュリティグループ・教授

研究者番号：50290838

交付決定額(研究期間全体)：(直接経費) 2,900,000円

研究成果の概要(和文)：本研究の目的は、申請者らが先に開発したC言語用のプログラミング教育支援ツール「pgtracer」をJava用に拡張し、運用実験によって得られた学習ログの解析によりプログラミング教育に有用な知見を獲得することである。pgtracerは、プログラムと実行状況を表示するトレース表の穴埋め問題を出題する。Java用に拡張するため、与えられたJavaソースコードからプログラムとトレース表を自動生成するプログラムを開発した。トレース表については、メッセージ送信などオブジェクト指向プログラミング特有の機能に対応するための拡張も行った。運用実験を行い、収集したデータを用いた分析により知見を得た。

研究成果の学術的意義や社会的意義

大規模化・複雑化するソフトウェアの開発を効率化するためオブジェクト指向技術の重要性が増している。本研究で開発したツールは、大学・高専におけるオブジェクト指向プログラミング教育の問題となる演習時間とスタッフの不足を補い、オブジェクト指向プログラム特有の概念の習得を支援する。本ツールが出題する問題はプログラムと実行状況を表現するトレース表の穴埋め問題であり、学生にとっても取り組みやすい。運用実験によって収集したデータからは、作成した問題難易度の分析、学生が誤りやすい箇所の抽出など、オブジェクト指向プログラミング教育における学習効果や教育効果の最大化を図るのに有用な知見を得ることができた。

研究成果の概要(英文)：The purpose of this study is to extend pgtracer, a programming education support tool for C language developed by the applicants, to Java, and to obtain useful knowledge for programming education by analyzing learning logs obtained through operational experiments. To extend pgtracer for Java, we developed a program that automatically generates a program and a trace table from given Java source code. The trace table was also extended to support functions specific to object-oriented programming, such as message transmission. We conducted operational experiments and analyzed the collected data to obtain knowledge. These results were published in journals and international conferences.

研究分野：計算機工学

キーワード：ラーニングアナリティクス プログラミング e-learning 穴埋め問題

1. 研究開始当初の背景

IT を活用したサービスは、工業分野だけでなく、医療や農業といった社会のあらゆる分野で重要な位置を占めている。イノベーション創出の過程では IT サービスを改善するために仕様変更が頻繁に起こる。また、大規模化・複雑化するソフトウェアの開発を効率化するために、モジュールの部品化や再利用に適したオブジェクト指向技術の重要性が増しており、オブジェクト指向技術を習得した技術者の育成が急務である。政府も「最先端 IT 国家創造宣言」や「初等プログラミング教育必修化」などを通じて IT サービスの創出、開発、運用を担う高度 IT 人材の育成を推進している。大学や高専においても、Java 等を用いたオブジェクト指向プログラミング教育が行われている。

プログラミング技術を習得するためには、多くのプログラムを作成することが必要だが、授業だけでは十分な機会と支援スタッフを確保することが難しい。また、オブジェクト指向プログラムは動的概念（動的束縛や多相性等）や抽象概念（抽象クラス、インタフェース等）を含むため動作が複雑で理解や学習が難しい。

近年、Moodle や Blackboard などに代表される e-learning システムの導入が教育現場でも進み、大量の学習データを収集し、そのデータを学習支援に活用するため、ラーニングアナリティクス（LA）が研究されるようになった。LA では、収集されたデータを用いて学生の学習過程の比較や追跡を行う。LA によって得られた知見を活用することで、オブジェクト指向プログラミング教育でも学習効果の測定や教育効果の最大化を図ることが期待できる。

2. 研究の目的

申請者らは C 言語を用いたのプログラミング教育支援ツール pgtracer (図 1) を開発している [1]。本研究では pgtracer を Java 言語用に拡張する。pgtracer は、プログラムとトレース表の組で構成される穴埋め問題を提供する。トレース表はルーチン名、各変数の値、各ステップの出力を利用してステップの実行順序やプログラムの実行過程を表す。トレース表はプログラムの実行過程を可視化することから、特に初心者のためのプログラミング教育の有効な手段として、我々はトレース表を採用した。pgtracer は、プログラムとトレース表の中に様々な種類の穴抜きを用意しており、トレース表とプログラムの中の穴抜きを学生に埋めさせることによって、プログラムの理解度とプログラムの構成度の両方を確認できる。

また、実際の授業で pgtracer を運用することを通じて、データ収集や考察を行う。

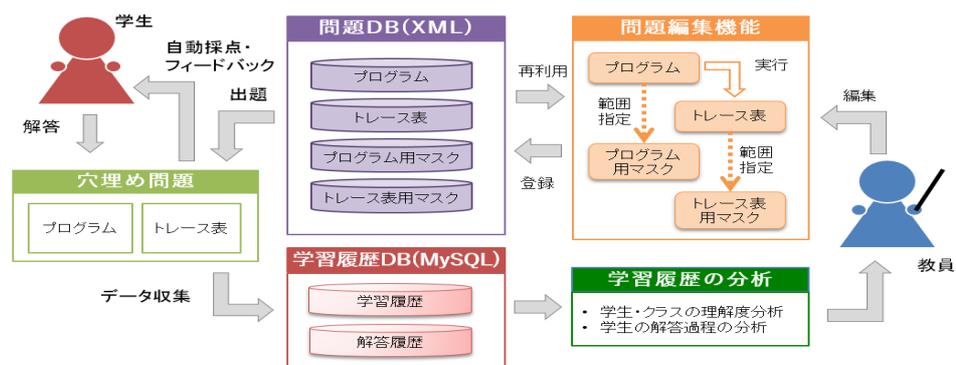


図 1 . pgtracer を用いた教育プロセス

3. 研究の方法

pgtracer は、プログラムとトレース表に対する穴埋め問題を出題し、学生が穴を埋めるたびに答案や解答時刻などを収集する。また、収集したデータの分析機能を持つ。これまでの研究において構造化言語である C 言語について、データ収集およびデータ分析を行い、様々な知見を得てきた。本研究では、Java プログラムの穴埋め問題を提供できるように pgtracer を拡張する。次に Java プログラミング教育を行う授業で利用し、学習データを収集する。C 言語プログラミングにおける LA の経験を活用し、収集したデータを用いて分析を行い、学生の解答行動の傾向を抽出する。分析によって得た知見をもとに考察する。

4. 研究成果

本研究の取り組みを大きく分類すると、(1)pgtracer の Java 言語への拡張、(2)Java 言語の穴埋め問題の作成、(3)運用実験の実施およびデータの分析、穴埋め問題の改良の 3 つとなる。

(1) pgtracer の Java 言語への拡張

pgtracer を Java 用への拡張のため、与えられた Java プログラムから XML ファイルの自動

生成プログラムの開発，Javaプログラムの実行状況を確認するためのトレース表について，XMLファイルの自動生成プログラムの開発を行った．また，pgtracerはMoodle上で動作するツールとして開発を行うため，Moodleのバージョンアップ等の盤環境の構築を行った．

与えられたJavaソースコードをXML形式に変換するプログラムの開発[2]

XML形式に変換するため，プログラムを表現するためのDTDを定義する．DTDでは，穴抜き箇所を適切に設定可能となるよう設計した．

XML生成プログラムは，Eclipse、Git、MavinとJavaCC、JavaParserを用いて開発した．JavaCCは字句解析ルーチンを生成するために利用した．XML形式への変換は，JavaParserを活用してJavaソースコードの構文解析を行い，構文解析木を走査することで実装した．実装の過程では，スタック等を活用してステップ番号の自動生成や，文との対応付けを自動化した(図2)．Javaプログラムは一般にクラスごとに複数のファイルで構成されるため，XMLファイルもそれぞれのファイルごとに作成した．

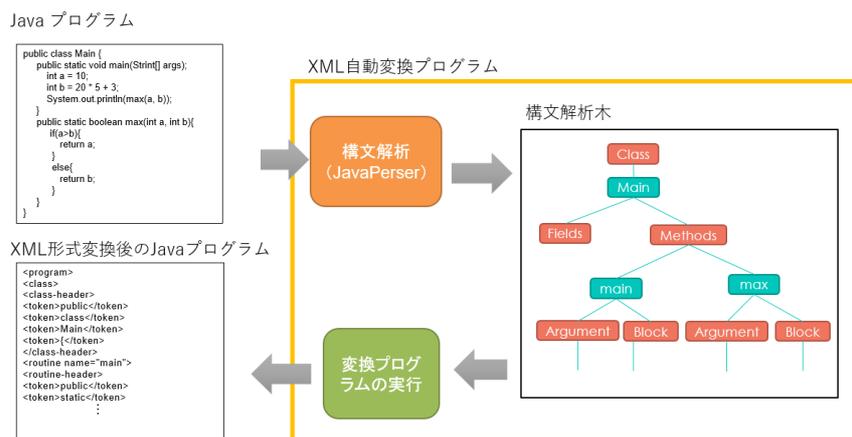


図2. JavaプログラムXML自動変換のフロー

与えられたJavaソースコードからトレース表を自動生成するプログラムの開発[3]

これまでのC言語用のpgtracerでは，構造化プログラムを対象としていたため，プログラムの実行から終了まで1つのトレース表で表現ができた．しかし，Javaプログラムのトレース表では，インスタンス間のメッセージ送受信やインスタンス変数などオブジェクト指向プログラミング特有の概念を表現する必要がある．そこで，トレース表の拡張を行った．これには，インスタンス毎にトレース表を分割する方法を採用した．与えられたJavaプログラムから，トレース表を自動生成するために，JavaParserを利用した．具体的には，与えられたJavaプログラムに，トレース表作成に必要なコードを自動挿入し，プログラムを実行することで，トレース表を自動的に生成する(図3)．

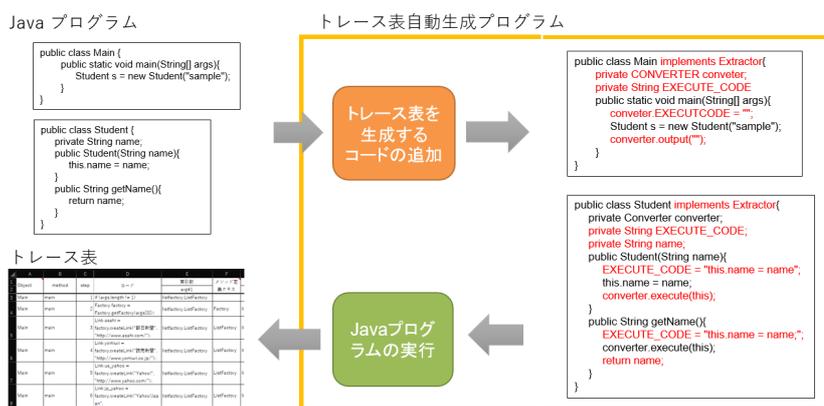


図3. トレース表作成フロー

(2) Java言語の穴埋め問題の作成[4]

本システムの運用実験を佐賀大学情報システム工学コース3年次に開講される「プログラミング演習」で実施するための穴埋め問題を作成した．対象授業を履修する学生は，1年次にC言語を学習済みであり，プログラミングの基礎は習得できている．そこでオブジェクト指向特有の概念やデザインパターンの学習を目的とした．問題は授業のテキスト[5]の中から，教授内容に合わせた12種類のデザインパターンについて，初級・中級・上級の穴埋め問題を作成した．問題の作成にあたっては，問題作成の意図を明確にするとともに，穴抜きの場所や個数について

は、C言語用 pgtracer による運用実験等で得られた知見を活用した。

ソースコードの中には、穴抜き箇所と同様のコードが前後に記載されている、トレース表では変数値の変化がない場合は前後の行に解答と同じ値が記載されているなど、ヒントとなる箇所が存在する。問題の難易度を保証するためには、ヒントとなる箇所を適切に隠す必要がある。これらを隠すために穴抜きとすると、学生が解答すべき穴抜きの数が多くなる。穴抜きの数が多くなると学生が解答を煩わしく感じ、学生のモチベーションが下がる、問題の難しさと混同するといったケースがあった。そこで、解答を要しない穴抜きを導入した。解答を要しない穴抜きを導入することにより、穴抜き数を増やすことなく、ヒントとなるコードや変数値を隠すことができ、問題難易度の調整をより柔軟に行えるようになった。作成した問題の例を図4、図5に示す。図において、番号が付されている濃い灰色の穴抜きは解答すべき穴抜き、番号が付されていない薄い灰色の穴抜きは解答を要しない穴抜きである。ステップ2の穴抜きは、ステップ1と同じコードであるため、解答を要しない穴抜きとしている。

```

public class Main {
    public static void main(String[] args) {
        // 'H'を持った CharDisplay のインスタンスを1個作る。
        AbstractDisplay d1 = new (1) ("H");

        // "Hello, world."を持った StringDisplay のインスタンスを1個作る。
        AbstractDisplay d2 = new (2) ("Hello, world.");

        // "こんにちは。"を持った StringDisplay のインスタンスを1個作る。
        AbstractDisplay d3 = new (2) ("こんにちは。");

        // d1,d2,d3とも、すべて同じ AbstractDisplay のサブクラスのインスタンスだから、
        // 継承した display メソッドを呼び出すことができる。
        // 実際の動作は個々のクラス CharDisplay や StringDisplay で定まる。
        d1 (3) ();
        (4) ();
        d3.display();
    }
}

```

Caller of the Method		Step of Called the Method			Argument of the Method		Instance Value	Local Variable of the Method
object	method	Class	method	step	String owner	Product product	List owners	Product p
Main	main	1	IDCardFactory	IDCardFactory				
Main	main	2	Factory	create	1	種族名		
			(1)	(2)	1	種族名		
			(3)	(4)	1	種族名		idCard#1
					2			idCard#1
			(5)	(6)	1	idCard#1		idCard#1
							{idCard#1}	idCard#1
			Factory	create	2		{idCard#1}	idCard#1
			Factory	create	3		{idCard#1}	idCard#1

図5 穴抜きをしたトレース表の一部

図4 穴抜きをしたプログラムの例 (Template Method)

(3) 運用実験の実施およびデータの分析、穴埋め問題の改良[6]

対象授業の演習問題として提供した問題のおよび解答数を表1に示す。ここで、穴抜き数は解答すべき穴抜きの数である。

表1. 問題と解答数

週	デザインパターン	レベル	問題形式	問題 ID	穴抜き数	解答数
1	穴埋め問題とトレース表を理解するための練習問題		プログラム	Ex01_PR	6	67
			トレース表	Ex01_TR	4	
6	Template Method	初級	トレース表	Ex06_PR	12	71
		初級	プログラム	Ex06_TR	12	
7	Factory Method	初級	トレース表	Ex07_PR	9	58
8	基本演習	初級	プログラム	Ex08_SP	2	67
	Iterator	初級	トレース表	Ex08_TR	10	
10	Factory Method	初級	トレース表	Ex10_TR	9	61
11	Composite	初級	トレース表	Ex11_PR	11	69
13	Strategy	中級	プログラム	Ex13_PR	6	66
14	Observer	中級	プログラム	Ex14_PR	10	65

表2にプログラムに設定した穴抜きについて、単一のトークンと複数のトークンからなるトークン率に分類して、正答率を示す。初級、中級のいずれでも単一トークンよりもトークン列の穴抜きの方の正答率が低いことから、トークン列の方が学生にとっては難しいことが分かる。また、初級よりも中級の方の正答率が低いことから、我々の穴抜きの設定レベルが適切であったことが分かる。

表2. プログラムに設定した穴抜きの正答率

レベル	問題数	単一のトークン		トークン列	
		穴抜き数	平均正答率 (%)	穴抜き数	平均正答率 (%)
初級	3	22	76.5	10	57.7
中級	2	11	67.2	9	41.9

表3にトレース表に設定した穴抜きについて、その種類ごとの平均正答率を示す。呼び出され

た関数名，インスタンスを回答する変数値についての正答率が低い．このことから，メソッド呼び出しやインスタンスなどオブジェクト指向プログラミングに特有の概念に関する問題は学生にとっては難しいことが分かる．

表 3. トレース表に設定した穴抜きの正答率

穴抜きの種類	穴抜き数	平均正答率(%)
コンストラクタまたは関数の呼び出し	16	72.5
呼び出した関数のステップ番号	3	46.0
呼び出された関数名	2	61.7
関数戻り値 (インスタンス)	5	64.0
変数値 (インスタンス)	1	38.8
変数値 (基本データ型)	4	71.1

学生へのアンケートの結果，63.8%の学生が問題をやや難しいと回答している．また，87.0%の学生が問題当たりの穴抜き数が適切であったと回答している．このことから，問題難易度が適切であったこと，解答を要しない穴抜きにより，問題難易度を制御できたことが分かる．また，自由記述においては，問題の表示方法に起因する解答のわずらわしさや変数名など大文字小文字の違いによる不正解の判定に対するコメントがあった．

学生のアンケートや学生の解答ログを精査し，以下の点について改善した．

- プログラムにおいて，インスタンス変数の定義にもステップ番号を付していたが，オブジェクト指向において，インスタンス変数はインスタンス生成時にすでに領域が確保された状態となる．このことから，インスタンス変数定義のコードには，ステップ番号を付さず，トレース表からも削除した．
- プログラムの穴抜きについて，想定した正答ではないが，プログラムは正常に動作する解答が存在した．このため，正答が唯一となるように穴抜きの場所やサイズを調整した．
- 学生の解答の煩わしさのが，問題のプログラムとトレース表は，1つの画像として提供し，それぞれ別のタブで表示していたことが学生の煩わしさ要因となっていた．このため，図6のようにユーザーインターフェースを改良した．プログラムは1つのタブで表示し，プログラム，トレース表へのナビゲーションを追加した．

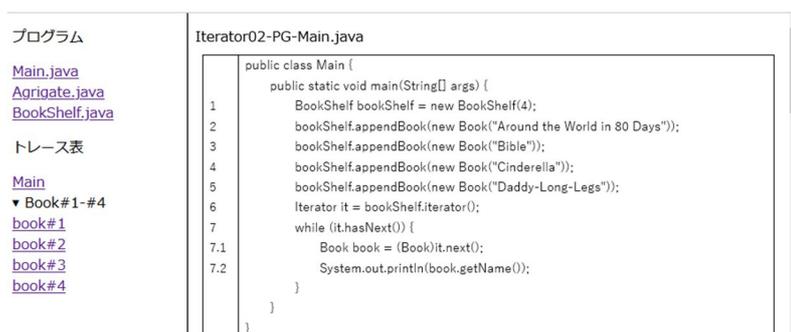


図 6. 改良後のユーザーインターフェース

< 引用文献 >

- [1] 村田美友紀，嘉藤直子，掛下哲郎，プログラミング学習支援ツール pgtracer を自学習に活用した授業実践と学習行動の分析，情報処理学会論文誌：教育とコンピュータ (TCE)，6，2，25-37，2020年6月．
- [2] Masateru Kishikawa, Tetsuro Kakeshita, Converting Java Source Code for Programming Education Support Tool pgtracer, Proc. 10th Int. Congress on Advanced Applied Informatics (IIAI-AAI), pp. 141-147, 2021年07月．
- [3] Masateru Kishikawa, Mika Ohtsuki, Tetsuro Kakeshita, Generating Trace Table for Java, 11th International Congress on Advanced Applied Informatics (IIAI-AAI), pp. 200-207, 2022年07月．
- [4] Miyuki Murata, Naoko Kato, Tetsuro Kakeshita, Fill-in-the-blank Questions for Object-Oriented Programming Education, 10th International Congress on Advanced Applied Informatics (IIAI-AAI), pp. 116-122, 2021年07月．
- [5] 結城浩：Java プログラミング言語によるデザインパターン入門．第3版．ソフトバンククリエイティブ，2021．
- [6] Miyuki Murata, Naoko Kato, Tetsuro Kakeshita, Improvement of Fill-in-the-blank Questions for Object-Oriented Programming Education, Proc. IFIP World Conference on Computers in Education (WCCE 2022), 1 page, 2022年08月．

5. 主な発表論文等

〔雑誌論文〕 計4件（うち査読付論文 4件/うち国際共著 0件/うちオープンアクセス 3件）

1. 著者名 Miyuki Murata, Naoko Kato, Mika Ohtsuki, Tetsuro Kakeshita	4. 巻 6
2. 論文標題 Fill-in-the-blank Questions for Object-Oriented Programming Education and Its Preliminary Evaluation	5. 発行年 2023年
3. 雑誌名 International Journal of Learning Technologies and Learning Environments	6. 最初と最後の頁 19
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 オープンアクセスとしている（また、その予定である）	国際共著 -
1. 著者名 Miyuki Murata, Naoko Kato, Tetsuro Kakeshita	4. 巻 1
2. 論文標題 Improvement of Fill-in-the-blank Questions for Object-Oriented Programming Education	5. 発行年 2023年
3. 雑誌名 Post Conference Book of IFIP World Conference on Computers in Education (WCCE 2022)	6. 最初と最後の頁 1-12
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 Kakeshita Tetsuro, Murata Miyuk, Kato Naoko, Nakayama Youhei	4. 巻 5
2. 論文標題 Analysis of Student 's Learning Log Data in Fill-in-the-Blank Programming Questions	5. 発行年 2022年
3. 雑誌名 International Journal of Learning Technologies and Learning Environments	6. 最初と最後の頁 1~17
掲載論文のDOI（デジタルオブジェクト識別子） 10.52731/ijltle.v5.i1.565	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -
1. 著者名 村田美友紀, 嘉藤直子, 掛下哲郎	4. 巻 6
2. 論文標題 プログラミング学習支援ツールpgtracerを自学習に活用した授業実践と学習行動の分析	5. 発行年 2020年
3. 雑誌名 情報処理学会論文誌：教育とコンピュータ（TCE）	6. 最初と最後の頁 25-37
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 オープンアクセスとしている（また、その予定である）	国際共著 -

〔学会発表〕 計7件（うち招待講演 0件 / うち国際学会 6件）

1. 発表者名 Miyuki Murata, Naoko Kato, Tetsuro Kakeshita
2. 発表標題 Improvement of Fill-in-the-blank Questions for Object-Oriented Programming Education
3. 学会等名 Proc. IFIP World Conference on Computers in Education (WCCE 2022) (国際学会)
4. 発表年 2022年

1. 発表者名 Masateru Kishikawa, Mika Ohtsuki, Tetsuro Kakeshita
2. 発表標題 Generating Trace Table for Java Programs
3. 学会等名 11th International Congress on Advanced Applied Informatics (IIAI-AAI) (国際学会)
4. 発表年 2022年

1. 発表者名 Masateru Kishikawa, Tetsuro Kakeshita
2. 発表標題 Converting Java Source Code for Programming Education Support Tool pgtracer
3. 学会等名 Proc. 10th Int. Congress on Advanced Applied Informatics (IIAI-AAI) (国際学会)
4. 発表年 2021年

1. 発表者名 Miyuki Murata, Naoko Kato, Tetsuro Kakeshita
2. 発表標題 Fill-in-the-blank Questions for Object-Oriented Programming Education
3. 学会等名 Proc. 10th Int. Congress on Advanced Applied Informatics (IIAI-AAI) (国際学会)
4. 発表年 2021年

1. 発表者名 Yuya Akinaga, Tetsuro Kakeshita
2. 発表標題 Development and Evaluation of a Search Application for Elementary School Programming Materials
3. 学会等名 Proc. 10th Int. Congress on Advanced Applied Informatics (IIAI-AAI) (国際学会)
4. 発表年 2021年

1. 発表者名 Tetsuro Kakeshita
2. 発表標題 Improved HyFlex Course Design Utilizing Live Online and On-demand Courses
3. 学会等名 In Proceedings of the 13th International Conference on Computer Supported Education (CSEDU 2021) (国際学会)
4. 発表年 2021年

1. 発表者名 掛下哲郎, 江島光代
2. 発表標題 小中学校の生徒と教員を対象とするオンライン・プログラミング講座
3. 学会等名 情報処理学会 情報教育シンポジウムSSS2020
4. 発表年 2020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	掛下 哲郎 (Kakeshita Tetsuro) (10214272)	佐賀大学・理工学部・准教授 (17201)	

6. 研究組織（つづき）

	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	大月 美佳 (Otsuki Mika) (20315138)	佐賀大学・理工学部・講師 (17201)	

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件

8. 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関