

令和 5 年 6 月 21 日現在

機関番号：13801

研究種目：基盤研究(C) (一般)

研究期間：2018～2022

課題番号：18K11566

研究課題名(和文)品質を考慮したプログラミング能力早期育成支援システムの研究開発

研究課題名(英文)Practice of tutoring support environment for early development of programming skills with software quality

研究代表者

野口 靖浩 (Noguchi, Yasuhiro)

静岡大学・情報学部・准教授

研究者番号：50536919

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)：本研究は、現状の初期プログラミング教育が、プログラミング言語の文法や論理の構造化に注力している場合が多く、変更性、テスト容易性、セキュリティなどの品質特性を意識に向けたプログラミング学習が行えない点を課題とした。初期段階から品質に着目したプログラミングを学習する環境を実現するために、初期のプログラミング学習者が演習コードの品質を評価できるように可視化による支援を提案した。行き詰まり時に理解不足な学習項目(カリキュラム後半の複数の領域にまたがる)による学習者の混乱を抑制するため、プログラミング演習中の学習者の行き詰まり箇所に対応する学習項目をダイレクトに参照できる仕組みを構築した。

研究成果の学術的意義や社会的意義

本研究は、従来のプログラミング教育では行われてこなかった初期段階から品質に着目しながらプログラミングを学ぶための方法論を探究するものであり、初学者自身が自身の演習コードの品質を評価する方法、現状では後続の複数の科目で学ぶ学習項目を学習者の混乱なく取り入れる方法、は新たな取り組みである。現在、小中高校でのブロックプログラミング言語によるプログラミング教育が普及してきており、本研究の成果は、その状況を前提とした新たな実践的なプログラミング教育の改訂に資するものと考えられる。

研究成果の概要(英文)：This study identified a problem in that current initial programming education often focuses on the syntax and logic structuring methods of programming languages, which prevents students from learning programming with an awareness of quality characteristics such as modifiability, testability, and security. In order to realize an environment in which students can learn basic programming skills while focusing on quality from the initial stage, we proposed a visualization support system that enables early learners to evaluate the quality of their edited exercise code themselves. In order to reduce learner confusion caused by that insufficiently understood learning items are widely dispersed in multiple subsequent learning areas (e.g., operating system, computer architecture, computer network, object oriented design, automated testing, and others) when learners get stuck, we developed an support environment that enables learners to refer to learning items corresponding to their stuck.

研究分野：学習支援システム

キーワード：プログラミング教育 学習支援システム ソフトウェア品質 非機能要求 拡張性 テスト容易性 セキュリティ

科研費による研究は、研究者の自覚と責任において実施するものです。そのため、研究の実施や研究成果の公表等については、国の要請等に基づくものではなく、その研究成果に関する見解や責任は、研究者個人に帰属します。

1. 研究開始当初の背景

近年、プログラミング技術やプログラミングにおいて鍛えられる論理的思考力が、現代社会の一員としての一般教養として重要であるとの認識が世界的に広まっている。その一方で、実際にユーザによって利用されるアプリケーションの作成には、様々な品質特性が必要となり、その重要性和要する技術は年々大きくなってきている。その中でも変更性、効率性、テスト容易性、セキュリティ、理解性、適合性、移植性など[a]はプログラミング能力に関係するところが大きく、それらの品質特性とプログラミングの双方の連携した学習を促進することは重要である。

しかしながら、現状では初期のプログラミング教育は、プログラミング言語の文法や論理の構造化手法を中心とした学習になっていることがほとんどである。この原因のひとつは、これらの品質特性の学習が複数領域に分散していることがある。例えば、ACM Computing Curricula[b]においてセキュリティに関わる学習項目は複数の領域(例えば、OS、コンピュータアーキテクチャ、ネットワークなど)に分散しているし、ソフトウェア設計に関わる学習項目も同様であり(例えば、ソフトウェア工学、ソフトウェア方法論、ソフトウェアテストなど)、更にそれらはカリキュラムの後半に位置することが多い。対して、プログラミング教育は初年度に実施されることが多く、これらの領域の学習を前提として講義・演習を設計することは難しい。

その結果、これらの特性を考慮したコーディングを中心的に学習する機会は学部教育にはない場合も多く、改めて社会人向けの教育[c]によって補完されることが多い。しかし、品質を考慮したプログラミングと考慮しないプログラミングは、設計・実装のプロセスも作成するソースコードも大きく異なるため、既に学習し定着したプログラム設計・実装の方法・価値観を大きく修正させる必要がある。これは非効率であると共に、対応できず脱落する学習者も想定される。また、補完が行われるまでの間は、不適当なプログラム設計・実装の方法・価値観で経験を積むことになり、妥当な方法ではない。

2. 研究の目的

本研究の目的は、初期段階から品質に着目しながらプログラミングを学べる学習コンテンツ・学習支援システムを開発することである。前述の背景より「プログラミングの初期段階から品質に着目しながらプログラミングを学習すること」を困難にしている問題点を「初期のプログラミング教育の段階に導入することが可能な品質に関わる学習項目の整理の必要」、「品質に関わる学習項目は、先行する複数の学習項目を必要とする場合が想定されることから、学習者の混乱を避けるための学習マップの必要性」、「従来の”プログラムが動くか否か”の判断基準に比べて、初学者自身がプログラムの品質に関わる評価を行うことの困難」と整理した。以上から、本研究では「初期段階から品質に着目しながらプログラミングを学習することが、学習項目の再整理・学習マップ支援・品質の良し悪しの理解を支援する可視化支援によって実現し得るか」を明らかにする。

3. 研究の方法

本研究の方法としては、(A)初期段階のプログラミング学習に導入可能な品質の学習項目を抽出するために既存の講義・演習やカリキュラムを対象として、学部の初期のプログラミング学習に導入可能な項目を抽出し整理する。(B)既存のプログラミング学習項目との混乱を避ける学習マップ作成支援システムの開発、(C)プログラムの品質の評価、比較を容易にする可視化システムの開発、(D)開発したシステム・コンテンツを実授業へ導入し、PDCAサイクルを回して改善・評価を行う、を進め、提案手法の実践評価を行う。

4. 研究成果

本研究の研究成果は、ソフトウェアの品質特性の内、変更性、テスト容易性、セキュリティを題材として、初期のプログラミング教育の段階を想定した学習支援の仕組みを設計・構築し、授業実践の中で評価を得て改善するサイクルを継続する段階まで遂行できたことである。

本研究では、(A)初学者のプログラミング学習の段階で導入可能な品質に関する学習項目、として、変更性、テスト容易性、セキュリティを抽出した(効率性に関しては、大学教育では“アルゴリズムとデータ構造”等の科目が初期のプログラミング教育の直後に立てられている場合が多く、既に教育の仕組みが存在するため本研究では扱わない)。各学習項目の難易度や依存関係を考慮した上で、特にプログラミング学習の中で、その品質の差異を体感することができる可能性を意識している。例えば、テスト容易性に関しては、そのコンポーネントや関数に対して自動テストを書けるかどうか、に関係するため、初期のプログラミング教育においても、テスト容易性の良し悪しによって、ソースコード上に自動テストの有無という結果を得ることができる。

(B)品質に関わる学習項目が先行する複数の学習項目に渡る点に関しては、初期のプログラミング演習の演習コードに対して、当該箇所を解決するために必要な学習項目を対応付けることで、プログラミング演習と理解不足な学習項目を対応付ける枠組みを構築した。文献[d][e]では、プログラミング演習において、学習者の演習時に生じた行き詰まりを自動検出し、学習者自身が

行き詰まりを生じた箇所に対して、理解不足の知識を確認するために対応する学習項目を参照できる仕組みを報告した。概要を図 1 に示す。この仕組みでは品質に関する学習項目を图中的 Lv.3 の箇所に設定することができる。学習者に対して、個々の行き詰まり箇所に対して、それに対応する学習項目を提示することができるため、本研究の想定する複数の領域に分散した学習項目に対して、プログラミング演習の中で学習者に必要な学習項目を提示することができる。また、Lv.2 において、教室内の学習者の行き詰まり箇所を集約することができるため、個別の学習者に個別の学習項目を提示するだけでなく、教室の中で多くの学習者が行き詰まり、共通の学習項目についての理解が不足しているとの判断を行うことができる。その場合、教師は教室全体に対して再度解決して理解不足を補うなどの対応を意思決定することができる。

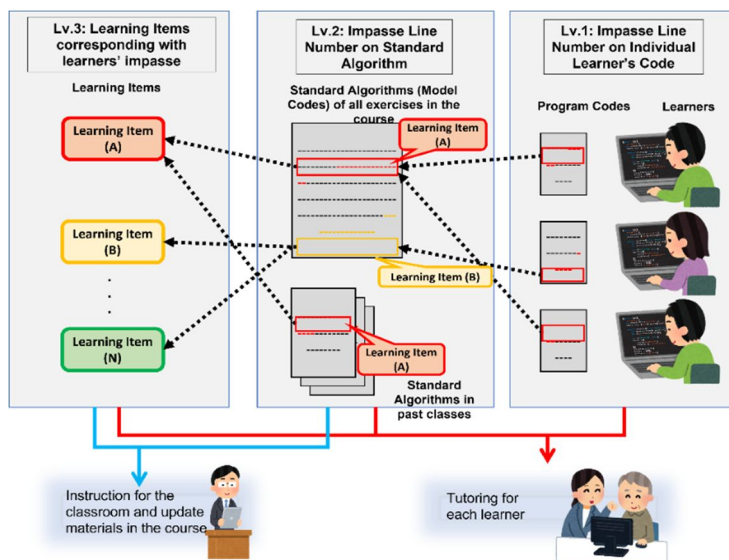


図 1 演習中の学習者の行き詰まりに基づき理解不足の学習項目を提案する仕組み

(C)プログラムの品質の評価、比較を容易にする可視化システムに関しては、変更性、テスト容易性、セキュリティを対象に可視化システムを提案した。セキュリティに関しては、バッファオーバーフロー・バッファアンダーラン等のプログラムスタックの扱いに起因するセキュリティリスクを対象として、これらのセキュリティリスクが生じる仕組みを可視化する仕組みを提案した。文献[f]では、プログラミング演習時にプログラム実行時のプロセス領域(スタック・ヒープ・グローバル・コード領域)を可視化して学習者に提示する仕組みを提案し、C言語のポインタの学習を対象に実践評価を行った。文献[g]では、更に拡張を進め、バッファオーバーフローやバッファアンダーラン等のバグを含むコード(即ち、正しい振る舞いをするプログラムではなく、実行するコードをコードリーディングしても振る舞いをシミュレーションできないプログラム)に対応し、セキュリティ上の問題を含むコードと、セキュアコーディングを行った場合のコードの違いの学習へと対応した。図 2 に提案システムを示す。図 2 右上の領域にはスタック領域が可視化され、学習者は図 2 左のコードの実行ステップに応じて、その状況の変化を観察することができる。演習の中で、学習者がコードを修正し実行し可視化することができるため、バッファオーバーフローやバッファアンダーランの振る舞いが、どのような仕組みで発生するのか(この理解のための学習項目は OS、コンピュータアーキテクチャに跨る)をコード演習ベースで理解することができ、セキュアコーディング規則の価値を、コードの振る舞いの変化から把握することができる。

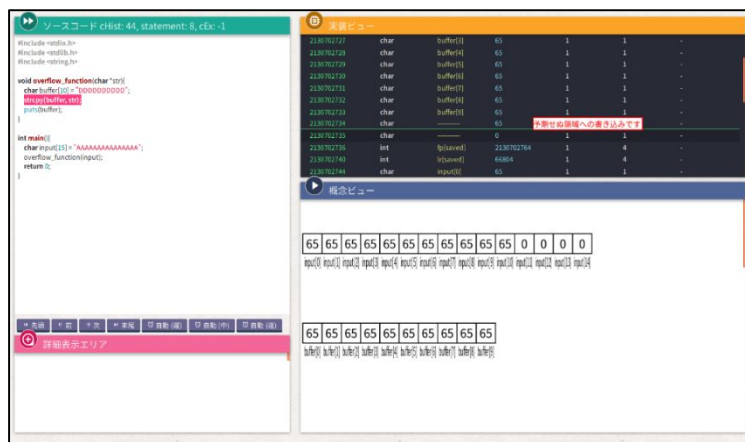


図 2 教師意図に基づくプロセス領域の可視化に対応した拡張 TEDViT

テスト容易性に関しては、コンポーネントあるいは関数に対して、自動テストを導入することで、当該コンポーネント・関数のテスト容易性を理解する演習を設計した。また、この演習に際して、学習者にテスト容易性の有無を評価しやすくするために、設計状態の可視化とウィザード形式で修正した設計に対するフィードバックを提供する支援システムを提案した。文献[h]では、テスト容易性指標の内、観測性・制御性・分解可能性に着目し、関数設計（関数分割並びに関数IF設計）を対象として試作システムを実装し、評価を行った。試作システムのインタフェースを図3に示す。学習者は図3左下のソースコードを選択し形式で変更し、現在の設計状態並びに設計変更の結果については図3左上に可視化される。学習者が選択した設計変更に関して、図3右のフィードバックを提供する。提供するフィードバックは、学習者の設計変更に応じて用意しており、その設計変更がどのテスト容易性指標に影響を及ぼすのか、また、その結果自動テストを記述する場合にどのような影響がでるのか（具体的には、なぜ自動テストが書けないのか、そしてその問題はこの設計変更で改善されるのか）を学習者に提示する。最終的に、学習者は改善した設計案のソースコードで自動テストを記述することができ、更にその自動テストを備えたコードでプログラミング演習を進めることができる。更に文献[i]では、更にオブジェクト指向設計の観点からテスト容易性改善を行える演習並びに支援システムへの拡張を提案した。表1にテスト容易性改善の評価結果を示す。提案システムを利用した場合と講義資料と演習課題を利用した場合で対照実験を実施し、提案システムを利用したケースの学習効果が高いことが示されている。

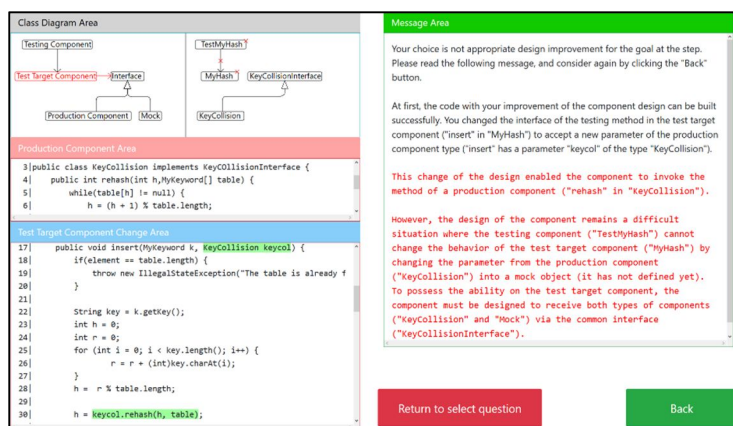


図 3 テスト容易性学習支援システム

表 1 テスト容易性設計改善の評価結果

	Pre-test	Post-test	Difference
Experimental	16.56	30.27	13.71
Control	15.62	19.82	4.20
Difference	0.94	10.45	9.51**

変更性に関しては、文献[j][k]において学習者のコンポーネント設計の自己レビューの支援の仕組みを構築した。この仕組みは図4のように、学習者の作成したコンポーネント設計を自然言語に変換して学習者にフィードバックすることで、課題の要求と、自身が考える設計案に反映した内容と、実際の自身の設計案の内容（変換された自然言語によるフィードバック）を見比べて、設計案の違いを学習者自身で評価できるように支援する仕組みを提案した。評価実験では、機能要件のみならず非機能要件についても対象とし、学習者はシステムの支援なしでは発見も改善もできなかった設計上の不具合を、システムの支援を受けることで発見・改善できることが示された。

(D)実授業でのPDCAサイクルを回して改善・評価の実施、に関しては、対象領域に合わせて学部・大学院・社会人向けの講座での実践を実施した。セキュリティに関しては、学部実施のenPiTSec「PBL演習-0 Basic SecCap演習（サイバー攻防基礎演習）」を実践の場としている。テスト容易性に関しては、社会人向けの講座を実践の場としており、C言語並びにJava言語の異なる講座で提案教材・支援システムを活用している。また、文献[l]において、当該講座での利用教材並びに支援システムを公開している。変更性に関しては大学院の講義を実践の場として実践している。各実践に関しては、上述の通り、その学習データ並びに参加者からのフィードバック等を元に改善を継続している。

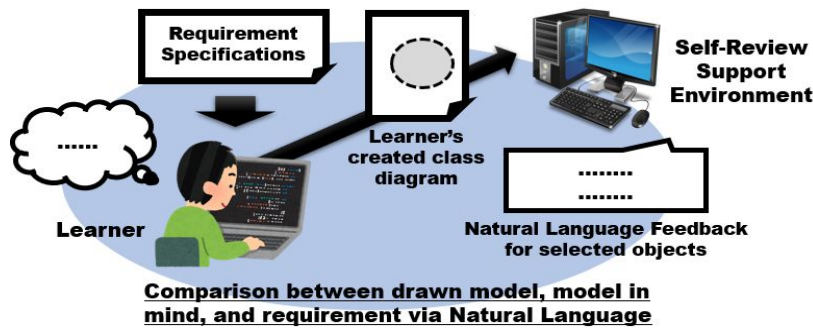


図 4 設計自己レビュー支援システムのアーキテクチャ

【引用文献】

- [a] IPA (2005). 組込みソフトウェア開発における品質向上の勧め（コーディング編），日経印刷株式会社.
- [b] ACM and IEEE-Computer Society Joint Task Force on Computing Curricula (2013). Computer Science Curricula 2013 Ironman Draft (Version 1.0), <http://ai.stanford.edu/users/sahami/CS2013/ironman-draft/cs2013-ironman-v1.0.pdf>, accessed at 2017-10-25.
- [c] IPA (2017). IPA セキュア・プログラミング講座, <https://www.ipa.go.jp/security/>, accessed at 2017-10-25.
- [d] Noguchi, Y., Ayabe, K., Yamashita, K., Kogure, S., Yamamoto, R., Konishi, T., Itoh, Y. (2020.11). Experimental Design of Automated Extraction for 3-Level Tutoring Support Information in Programming Exercises, Proceedings of the 28th International Conference on Computers in Education, 255-260
- [e] Ikegame, T., Noguchi, Y., Kogure, S., Yamashita, K., Yamamoto, R., Konishi, T. and Itoh, Y. (2021.12). Instruction Support System Using Impasse Detector and Major Failure Diagnoser for Programming Exercises, Proceedings of the 29th International Conference on Computers in Education, 700-702.
- [f] Yamashita, K., Fujioka, R., Kogure, S., Noguchi, Y., Konishi, T., and Itoh, Y. (2017) Classroom practice for understanding pointers using learning support system for visualizing memory image and target domain world, Research and Practice in Technology Enhanced Learning (RPTEL), 12:17, 1-16, doi:10.1186/s41039-017-0058-4.
- [g] 瀧嘉人, 野口靖浩, 小暮悟, 山下浩一, 小西達裕, 伊東幸宏. (2019) 視覚化に基づくプログラムの脆弱性学習支援システムの構築, 第 44 回教育システム情報学会全国大会, 45-56.
- [h] Noguchi, Y., Ihara, D., Kogure, S., Yamashita, K., Konishi, T. and Itoh, Y. (2019.12) Learning Support System for Software Component Design based on Testability, Proceedings of the 27th International Conference on Computers in Education, 306-311. (Kao Shuan, Taiwan). (Best Technical Design Paper Award Nominee)
- [i] Muramatsu, M., Noguchi, Y., Kogure, S., Yamashita, K., Konishi, T., Itoh, Y. (2020.11). Introducing a Mock Technique into a Learning Support System for Program Design Based on Testability, Proceedings of the 28th International Conference on Computers in Education, 205-214
- [j] Noguchi, Y., Nishihata, S., Kogure, S., Yamashita, K., Kondo, M., Konishi, T. (2021.3). What is the Meaning of My Model? - Self-Review Support Environment based on Natural Language Translation from Learners' Software Structural Models, In Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21) ACM, New York, NY, USA. 7 pages. <https://doi.org/10.1145/3408877.3432387>
- [k] 西畑瞬, 高倉将希, 岸本祐季, 野口靖浩, 小暮悟, 近藤真, 小西達裕. (2019.09) UML クラス図の自然言語化に基づくクラス図自己レビュー支援環境の構築, 第 44 回教育システム情報学会全国大会, 37-38.
- [l] Knowledge System Lab (2023). Testability Learning Support System (TLS), <https://wpp.shizuoka.ac.jp/ks/research/tls/contents/>, accessed at 2023/06/01.

5 . 主な発表論文等

〔雑誌論文〕 計0件

〔学会発表〕 計11件（うち招待講演 0件 / うち国際学会 6件）

1 . 発表者名 Ikegame, T., Noguchi, Y., Kogure, S., Yamashita, K., Yamamoto, R., Konishi, T. and Itoh, Y.
2 . 発表標題 Instruction Support System Using Impasse Detector and Major Failure Diagnoser for Programming Exercises
3 . 学会等名 Proceedings of the 29th International Conference on Computers in Education (国際学会)
4 . 発表年 2021年

1 . 発表者名 池亀 智紀, 野口 靖浩, 小暮 悟, 山下 浩一, 山本 頼弥, 小西 達裕, 伊東 幸宏
2 . 発表標題 プログラミング演習における行き詰まり検出に基づく個別指導支援システムのオンライン・オンデマンド型演習への対応
3 . 学会等名 人工知能学会研究会資料 先進的学習科学と工学研究会
4 . 発表年 2022年

1 . 発表者名 Soma, H., Kogure, S., Noguchi, Y., Yamashita, K., Yamamoto, R., Konishi, T. and Itoh, Y.
2 . 発表標題 Development of Mapping Function between Variable-Value and Object Properties for Program Behavior Visualization Tool TEDViT
3 . 学会等名 Proceedings of the 29th International Conference on Computers in Education (国際学会)
4 . 発表年 2021年

1 . 発表者名 Y. Noguchi, K. Ayabe, K. Yamashita, S. Kogure, R. Yamamoto, T. Konishi, Y. Itoh
2 . 発表標題 Experimental Design of Automated Extraction for 3-Level Tutoring Support Information in Programming Exercises
3 . 学会等名 Proceedings of the 28th International Conference on Computers in Education (国際学会)
4 . 発表年 2020年

1 . 発表者名 M. Muramatsu, Y. Noguchi, S. Kogure, K. Yamashita, T. Konishi, Y. Itoh
2 . 発表標題 Introducing a Mock Technique into a Learning Support System for Program Design Based on Testability
3 . 学会等名 Proceedings of the 28th International Conference on Computers in Education (国際学会)
4 . 発表年 2020年

1 . 発表者名 Y. Noguchi, S. Nishihata, S. Kogure, K. Yamashita, M. Kondo, T. Konishi
2 . 発表標題 What is the Meaning of My Model - Self-Review Support Environment based on Natural Language Translation from Learners ' Software Structural Models -
3 . 学会等名 Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE ' 21) (国際学会)
4 . 発表年 2021年

1 . 発表者名 Yasuhiro NOGUCHI, Daiki IHARA, Satoru KOGURE, Koichi YAMASHITA, Tatsuhiro KONISHI & Yukihiro ITOH
2 . 発表標題 Learning Support System for Software Component Design based on Testability
3 . 学会等名 27th International Conference on Computers in Education (国際学会)
4 . 発表年 2019年

1 . 発表者名 瀧 嘉人, 野口 靖浩, 小暮 悟, 山下 浩一, 小西 達裕, 伊東 幸宏
2 . 発表標題 視覚化に基づくプログラムの脆弱性学習支援システムの構築
3 . 学会等名 第44回教育システム情報学会全国大会
4 . 発表年 2019年

1. 発表者名 村松 昌哉, 野口 靖浩, 小暮 悟, 山下 浩一, 小西 達裕, 伊東 幸宏
2. 発表標題 Testabilityに基づくプログラム設計学習支援システムのモックへの対応
3. 学会等名 第88回 先進的学習科学と工学研究会
4. 発表年 2020年

1. 発表者名 伊原大貴, 野口靖浩, 小暮 悟, 山下 浩一, 小西 達裕, 伊東 幸宏
2. 発表標題 Testability に基づくプログラム設計学習支援システムの構築と教育実践
3. 学会等名 電子通信情報学会研究報告ET
4. 発表年 2018年

1. 発表者名 伊原大貴, 野口靖浩, 小暮 悟, 山下 浩一, 小西 達裕, 伊東 幸宏
2. 発表標題 Testabilityに基づくプログラム設計学習支援システムの構築
3. 学会等名 第16回情報学ワークショップ
4. 発表年 2018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	小西 達裕 (Konishi Tatsuhiko) (30234800)	静岡大学・情報学部・教授 (13801)	

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	山下 浩一 (Yamashita Koichi) (30340110)	常葉大学・経営学部・准教授 (33801)	
研究分担者	小暮 悟 (Kogure Satoru) (40359758)	静岡大学・情報学部・准教授 (13801)	
研究分担者	山本 頼弥 (Yamamoto Raiya) (70825116)	山陽小野田市立山口東京理科大学・工学部・助教 (33801)	

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