

科学研究費助成事業 研究成果報告書

令和 4 年 6 月 13 日現在

機関番号：25406

研究種目：基盤研究(C)（一般）

研究期間：2018～2021

課題番号：18K02903

研究課題名（和文）プログラミング演習における学習者個別の演習状況の評価とその応用に関する研究

研究課題名（英文）Study on evaluation of individual learner's exercise situation in programming exercise

研究代表者

宇野 健（UNO, TAKESHI）

県立広島大学・地域創生学部・准教授

研究者番号：20305783

交付決定額（研究期間全体）：（直接経費） 2,500,000円

研究成果の概要（和文）：本研究では、プログラミング学習における学習者個別の学習状況を定量的に把握し、的確な指導を行うことを目的とした。まず、Web上で利用可能なC言語プログラミング学習環境の開発と学習データのサーバへの蓄積を行った。次に、蓄積した学習データを用い、演習過程の評価を行うフィードバックシステムの開発と運用を行った。これを応用し、授業中の学習者個別の演習の進捗を、リアルタイムで把握可能とし、指導の効率化を図るシステムを開発し、運用した。これらのデータを活用し、成績不振兆候者の早期発見を行う手順を考案し、授業進行中における成績不振兆候者の検出を可能とした。

研究成果の学術的意義や社会的意義

近年、小中学校で必修化されるなど、プログラミング教育の重要性が認識されている。プログラミング教育には実習が不可欠となるが、実習環境の構築や、指導側の進捗把握が難しいという問題があった。そこで本研究では、これらの問題を解決するため、Web上で利用可能なC言語プログラミング学習環境の開発と運用を行い、サーバに蓄積した学習データを用い、演習過程の評価を行うフィードバックシステムの開発と運用を行った。さらに、これらの学習データを活用し、定量的な手法で成績不振兆候者の早期発見を行う手順を考案し、授業進行中における成績不振兆候者の検出を可能とした。

研究成果の概要（英文）：The purpose of this study was to quantitatively grasp the learning situation of each learner in programming learning and to provide accurate guidance. First, we developed a C language programming learning environment that can be used on the Web and stored the learning data on the server. Next, using the accumulated learning data, we developed and operated a feedback system that evaluates the exercise process. By applying this, we have developed and operated a system that makes it possible to grasp the progress of individual learners' exercises in class in real time and improve the efficiency of teaching. Utilizing these data, we devised a procedure for early detection of learner with poor grades, and made it possible to detect learner with poor grades during class.

研究分野：プログラミング教育

キーワード：プログラミング演習 リアルタイム演習状況把握 成績不振兆候者抽出

1. 研究開始当初の背景

2016年に、小・中学校において2020年度からプログラミング教育が必修化されることが発表された。プログラミング教育を推進する背景として、WebエンジニアをはじめとするIT人材の不足があった。さらに、今後予想されるAIやIoT等の普及による産業構造の変化により、その需要はますます増える可能性が高いとされたためである。プログラミング学習は、単にコーディングを覚えさせるだけでなく、プログラミング的思考を身に付けさせることができ、従来重要視されている読解力・論理的思考・創造的思考・問題解決力等の習得にも適している。プログラミング学習には、理論と演習の両面からのアプローチが不可欠である。特に演習については、単にプログラムやアプリケーションの作成のスキルを修得するだけでなく、トライ&エラーを繰り返すことによる種々の能力の獲得も期待できる。

しかし、実際のプログラミング演習の現場では、初学者がプログラミング演習を行う際に、プログラミング環境づくりが困難などの理由や、モチベーションの維持等の問題から、自学習が進まないという問題がある。さらに、指導者による、学習者個々の演習の進捗の確認や、個々の演習の評価などが困難となる。特に演習中においても理解しにくいポイントなどを自己申告しない学習者も多く、それらの把握が困難となり、ドロップアウトする者も少なくない。これは、教員が課題の提出等を義務付けて、詳細に添削等を行えば、同等の評価は可能となる。しかし、そのためには多くの時間が必要となり、毎週実施される授業で行うことは現実的ではない。また、この手法では、ソース作成にどれぐらい時間がかかったか、どの部分で躓いていた等、学習者個人ごとの学習履歴を把握することは不可能である。

2. 研究の目的

本研究では、WebベースのC言語プログラミング学習支援システムを開発し、授業での運用によってサーバに蓄積される学習者の作成ソースや操作履歴等の学習データを用い、それぞれの目的を達成するためのシステムを開発・運用することによって、その効果の検証を行う。主に以下に挙げる(1)～(4)を到達目標とする。

(1) C言語プログラミング学習環境の開発と学習データのサーバへの蓄積

Web上で動作するC言語プログラミング演習システムを開発し、学習者に利用させる。一つの画面でコーディング、コンパイル、実行の全ての作業を行うことが可能なWebアプリケーションとして開発するため、スマートフォンやタブレットPCでも利用可能とする。これにより、学習者の自学習の時間を増やすことを目的とする。また、C言語の入力待ちを実現するために、PHP言語による実行エミュレータを開発し、コンパイルエラーだけでなく、条件文の誤り等によって発生する実行時エラーの検知も可能とする。さらに、その詳細な操作履歴等をサーバに蓄積し、以降開発するシステムで活用する。

(2) 演習過程の評価とそのフィードバックによる学習モチベーションの向上

蓄積した学習データを元に、学習の評価を試験やレポート等の結果だけではなく、演習の過程についても評価し、それを学習者にフィードバックするシステムを開発する。このシステムを用いることにより、授業中の演習や、自学習における演習などの過程の評価を自動的にを行い、それを学習者に提示する。これにより、学習者のモチベーションを高め、演習への取り組みを増やすことを目的とする。

(3) 授業における学習者個別の演習の進捗把握と指導の効率化

演習システムで蓄積した学習データを元に、演習授業中の進捗把握をリアルタイムに行うシステムを開発する。演習授業では、教員やTAが多くの学習者の対応を行う必要があるため、学習者個々の細かい進捗の把握や、理解できずに手を止めている学習者の発見が困難なことがある。そのため、持ち運びが容易なタブレットPC上に、各学習者の詳細な進捗を視覚的に一覧表示し、演習が停滞している者を発見する。これにより、授業中の効率的な演習指導を可能とすることを目的とする。

(4) 授業進行中における成績不振兆候者の早期発見

演習システムで蓄積したデータを元に、成績不振兆候者を授業期間中に発見する。プログラミングの授業では、ドロップアウトする学習者が少なくない。しかし、従来は、それらを把握するのが定期試験後やレポート課題提出後になることが多く、分かった時には既に手遅れになるケースが多かった。そこで、蓄積された過去の学習データから、ドロップアウトする学習者の特徴を抽出し、授業進行中に成績不振兆候者の推定と抽出等を行う。これにより、毎回の授業ごとにドロップアウト兆候者を検出し、早期に適切な指導を可能とすることを目的とする。

(5) 申請時からの変更点

本題目の申請時の研究目的は上述した4点であるが、期間中に申請時に予想できなかった大きな変化が2つ生じた。1つは、2020年からの新型コロナウイルスによる授業方法と内容の大幅な変更である。もう1つは同じく2020年度からの大学の組織改編により、運用実験予定の科目「プログラミング」の配当年次が1年次配当から2年次配当へ繰り下げられ、また、授業の実施形態が semester制からクォーター制へ変更となったことである。このため、2020年度は対象の授業が開講されず、運用実験ができなかった。また、クォーター制への変更による授業実施方法や内容の変化やオンライン授業の実施等により、条件が大幅に変更になってしまったため、2021年度の授業のデータと、2019年度のデータとの比較が困難となった。以上により、リアルタイムでの成績不振兆候者の検出実験が困難となった。

これについては、申請時の予定を変更し、オンライン授業におけるプログラミング演習の円滑化のためのプログラミング演習状況のリアルタイム可視化システムの機能強化や、それに伴う負荷の増加によるサーバの強化等を行うこととした。

3. 研究の方法

(1) C 言語プログラミング学習支援システムの開発

今回の研究において、学習者と教員のインターフェースとなるシステムである。Web ブラウザ上で C 言語のソースの作成、コンパイル、実行が可能となる。また、このシステム上で学習者が行った操作は、全てサーバのデータベースに蓄積し、これを後述する各システムにおいて活用する。ソースファイルは、Web 言語のひとつである PHP を用いて開発するエミュレータで実行する。これにより、入力待ちが必要なプログラムの実行が Web ブラウザ上で可能になる。また、制御文の条件違いによる無限ループや論理エラー等の実行時エラーの取得と、それらのデータベースへの蓄積が可能となる。

(2) 学習状況の評価のためのフィードバックシステムの開発

C 言語プログラミング学習支援システムを運用によってサーバに蓄積される学習者の各種データを用い、学習者個別の学習状況の評価し、学習者に提示するフィードバックシステムの開発を行う。演習や練習問題の達成率や、入力文字数やコンパイル、実行回数等を元に評価基準を作成し、相対的に評価を行う。評価基準については、前年度までの運用データ等を元に、各種パラメータやしきい値の設定を行う。

(3) 授業におけるプログラミング演習状況の可視化システムの開発

プログラミング演習中に、教員のタブレット端末上でクラスの全学習者の演習状況を、リアルタイムに把握するためのシステムを開発する。学習者の演習状態については、C 言語プログラミング演習システムから得られる学習データを、リアルタイムで取得・判定を行う。状態判定には、学習者の課題達成率、総入力行数、総タイプ数、コンパイルエラー、実行時エラーなどを用いる。これを元に、問題が発生していると思われる学習者を割り出し、その状況や、進捗状況に応じて、教員や TA の持つタブレット端末上に提示する。

(4) オンライン授業でのプログラミング演習のための機能拡張

新型コロナウイルスによる完全オンライン授業の実施のため、プログラミングの演習もオンラインでの実施が必要となった。座学と違い、学習者の演習状況の把握が困難になるプログラミング演習のために、C 言語演習システムの機能拡張を行い、プログラミング学習者のリアルタイムモニター機能の開発を行う。これは、上述したプログラミング演習システムのログデータと、演習状況の可視化システムをベースとして作成する。1 枚のウィンドウに複数の学習者のソースプログラムの作成状況をほぼリアルタイムに表示し、対面式授業と同等の演習環境を実現する。また、コピー & ペーストの検知などを新たに追加し、オンラインでの試験への対応も試みる。

(5) 学習データを用いた成績不振兆候者の検出の試み

演習システムで収集した定量的な学習データを用いて、試験成績を推定することによって成績不振兆候者の検出を試みる。授業期間全体の学習データの各項目に対して重回帰分析を行い、試験偏差値を推定する詳細なアルゴリズム等については、蓄積されたデータの分析を行ってから補正等を行う。

4. 研究成果

(1) C 言語プログラミング演習システムの開発

2018 年度は、主に C 言語プログラミング演習システムの開発を行った(図 1)。サーバ上の C 言語コンパイラによって正常にコンパイルされたソースファイルを、Web 言語のひとつである PHP を用いて開発した C 言語エミュレータで実行する。これにより、従来では不可能であった、標準入力を用いたデータ入力待ちが必要なプログラムの実行を Web ブラウザ上で可能とした。さらにこれまで取得が困難であった、制御文の条件違いによる無限ループや論理エラー等の C 言語の実行時エラーの取得と、それらのデータベースへの蓄積が可能となった。

開発した本システムを実際の授業「プログラミング」で、活用した(受講者数 47 名)。運用期間は第 10 回から第 14 回までの 4 回(39 日)で、大きなトラブルなく運用できた。ソースプログラムの

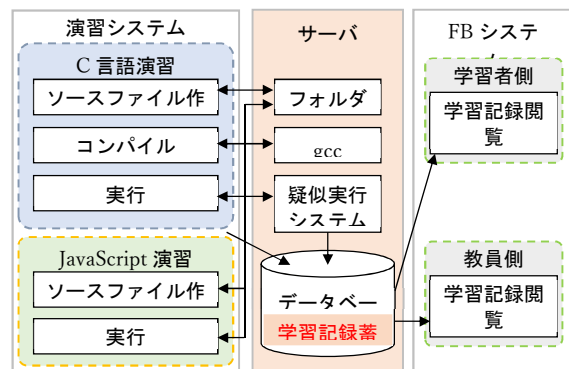


図 1 プログラミング演習システムの構成図

の保存回数 254 回、コンパイル回数 239 回、実行回数 138 回、例題プログラム達成率 91.0%、練習問題(任意課題)の達成率 74.9%など、定量的データが正しく蓄積されていたことも分かった。前年度の授業方式(大学の PC もしくは家の PC で演習、USB メモリでデータを持ち運び)では、練習問題の達成率が 33%程度だった。これは、受講者アンケートの結果、本システムの運用により、家の PC に開発環境を入れていなかった者、持ち運びが面倒だった者、スマートフォン等で通学中など、空き時間での演習が可能となったためであることが分かった。

(2) 学習状況の評価のためのフィードバックシステムの開発

2019 年度は、昨年度開発した C 言語プログラミング学習支援システムを授業で運用するとともに、その際に発生する学習ログデータを元にした、学習状況評価のためのフィードバックシステム(以下 FB システム)の開発を行った。演習の評価は、例題や練習問題の作成の達成率や、入力文字数やコンパイル、実行回数等を元に評価基準を作成し、相対的に行うようにした。特に本システムの特徴である、実

行時エラーの検知機能を用い、より詳細に演習の躓きなどを評価することを可能とした。

学習者側のインターフェースは、上述した評価の提示や、振り返り機能などを実装した。教員側のインターフェースは、教員向けのフィードバック機能として、全学習者の詳細な評価の閲覧、作成したソースプログラムの表示、演習の操作ログの集計、各種エラーの集計機能等を実装した。

開発したシステムを用いて、「プログラミング」(受講者 46 名)の授業で運用を行った。その結果、総ログインが約 1600 件あり、そのうち授業時間外が 1050 件となり、時間外での利用が非常に多くあることが分かった。授業終了後に行ったアンケートでは、システムの操作性やエラーの解決支援について高評価が得られた。学習状況の評価の他に、FB システムには各種演習操作記録を集計・表示する機能を実装した。学習者側の FB システムでは、單元ごとのコンパイル成功率や、ファイルごとのエラー履歴などを確認することができる。

「本日の成果」まとめ(全体)						
第4回 データの入出力						
2013年5月13日 13時00分 ~ 14時30分						
A+	A	B	C	D		
17人	22人	3人	1人	2人		
例題達成率の平均: 93.3%			練習問題達成率の平均: 52.8%			
ID	氏名	例題	練習問題	コンパイル	タイプ数	評価
		100% (1個中1個)	75% (4個中3個)	66.7% (8回中6回)	2350打	A
		100% (1個中1個)	100% (4個中4個)	68.8% (16回中11回)	2120打	A+
		100% (1個中1個)	50% (4個中2個)	63.6% (11回中7回)	1119打	A+
		100% (1個中1個)	25% (4個中1個)	100% (8回中8回)	837打	A
		100% (1個中1個)	50% (4個中2個)	50% (8回中4回)	1707打	A
		100% (1個中1個)	25% (4個中1個)	100% (8回中8回)	500打	A
		100% (1個中1個)	25% (4個中1個)	62.5% (8回中5回)	3152打	A
		100% (1個中1個)	25% (4個中1個)	66.7% (8回中5回)	1629打	A

図2 教員側のフィードバック確認画面

教員側のシステムでは、課題ファイルの作成状況やソースプログラムの編集履歴、コンパイルエラーの検出傾向など、さらに詳細に学習者の演習操作記録を確認する機能がある。教員側 FB システムの主な機能は、学習者別の基本情報、作成ファイルの一覧、ソースファイルの進捗状況、コンパイルエラーの集計、演習操作ログの集計、実行時エラーの記録がある(図2)。

FB システムとは別に、演習システムに蓄積された演習データを元に、ほぼリアルタイムに演習の進捗状況を判定し表示する、演習状況のリアルタイム可視化システムの開発を行った。演習ログデータから取得した学習者の課題達成率、総行数、タイプ数、コンパイルエラー、実行時エラーなどを10秒おきに取得する。これを元に遅れている学習者、手が止まっている学習者等をほぼリアルタイムで割出し、システムが自動で学習者の状態を判定する。そして、その状態や、進捗状況に応じてシートマップ上で対象の学習者の座席の色を変化させ、教員やTAに提示するようにした(図3)。

図3 演習状況のリアルタイム可視化システムの画面

実際に本システムを「プログラミング」(受講者数 46 名)の授業で運用したところ、問題なく動作することが分かった。しかし、状態判定については、じっくりと考える学習者と、単にさぼっている学習者の判定が同じように判定されるなど、改善の余地が大いにあることが分かった。利用したTAからは、手を挙げない学習者に対してコミュニケーションを取るきっかけになる、遅れている学習者を素早くサポートできる、作成するファイルを間違えている学習者を発見できる等の肯定的な意見が得られた。

(3) 授業におけるプログラミング演習状況の可視化システムの開発

2020年度は、学部学科再編の影響で、対象授業である「プログラミング」が、配当年次変更(1年次から2年次へ)により、授業が行われなかったため、運用実験が実施できなかった。また、新型コロナウイルスによって、授業が完全オンライン化となり、演習授業の実施が困難となった。そのため、オンラインでの演習授業への対応として、リアルタイムでの各個人の演習状況の把握を可能とするための作成中のソースプログラムのモニター機能の開発を行った。また、このプログラムの動作を確認するために、JavaScriptを用いた演習授業「プログラミング演習」(受講者数 42 名)で運用した。この際、演習システムにおいて、JavaScriptへ対応させるための機能拡張を実施した。

オンライン授業での演習では、対面式授業での演習と同様に、リアルタイムで演習状況を把握する必要がある。そのため、演習の際に蓄積される各種ログデータをリアルタイムに集計し、教員用画面に個々の演習状況を表示するシステムの開発を行った。当初は利用者がデータ保存を行わないと、データの更新が行われなかったが、Ajaxの非同期通信を用い、10秒間隔で学習者の編集画面の状況を取得するようにした。また、学習者のソースコード画面を1~16人のソースプログラムを同時表示することを可能とした。また、表示する授業者の自動切替えや、コピー&ペーストを検知する機能などを追加した。

開発したシステムを、実際の授業において、完全オンラインでの演習で運用を行い、有効性の検証を行った。授業はJavaScriptを用い、モニター上で提示したサンプルプログラムの実行や、応用問題として、自分でソースプログラムを作成するなどの演習を行った。6回にわたって運用を行った結果、今回開発、実装したソースプログラムのリアルタイムモニター機能に関しては問題なく動作し、リアルタイムで学習者の演習状況を把握することが可能となり、対面式と変わらない演習指導ができた。

(4) 学習データを用いた成績不振兆候者の検出の試み

2021年度は、過去の運用実験で収集したプログラミング学習ログデータを用い、成績不振兆候者の早期発見手法を提案し、実際の授業に適用することによって、リアルタイムで成績不振兆候者を抽出する。成績不振兆候者の早期発見では、開発したシステムで収集した定量的学習データを用いて、試験

成績の推定を行う手法を提案する。試験成績とは、中間試験の得点と期末試験の平均得点の偏差値（以下、試験偏差値）を指す。2019年度の授業期間全体の学習データに対して重回帰分析を行い、試験偏差値を推定する式を作成した。推定した結果が偏差値であるため、成績不振兆候者の判定を行うためのしきい値は推定時に任意に設定可能である。

本手法では、(a)授業の実施と演習、(b)推定試験偏差値の算出(中間試験以降は補正)、(c)しきい値の設定と下位判定、(d)授業開講期間中、①～③を繰り返す、という手順を用いた。

試験偏差値の推定は各授業の終了時に行う。推定に利用する式や成績不振兆候者の抽出手法については後述する。ここで、中間試験後の推定については、推定精度向上のため中間試験結果をもとに補正を行う。具体的には、2019年度のデータをもとに作成した補正式を適用し、得られた結果を新たな推定値とする。また、上記③において、下位判定者のうち実際の試験偏差値がしきい値を下回る学習者(成績下位者)の割合を「的中率」、実際の成績下位者のうち下位判定者の割合を「抽出率」として、本手法の推定精度を評価する。的中率、抽出率ともに高ければ高いほど推定精度が高いと判断する。

次に、試験成績の推定にあたり、2019年度の運用で蓄積された全学習データに対して、試験成績を目的変数とした重回帰分析を行った。ログイン、パスワードやファイル情報の変更操作などシステムに対する操作は除外し、具体的な演習操作項目を説明変数とした。その結果、各説明変数の係数の有意確率 p が5%未満で、且つ決定係数が50%を超える以下のような重回帰式を2式作成できた。

- ・重回帰式 A: 決定係数 51.7% : $\text{コンパイル成功率} \times 43.57 + \text{FB 評価値平均} \times 11.88 - 21.47$
- ・重回帰式 B: 決定係数 50.9% : $\text{コンパイル成功率} \times 37.94 + \text{課題達成率} \times 30.77 + 2.16$

これらの式を実際の学習データに適用し、試験成績の推定を行った。前述の重回帰式 A および B は、2019年度の全学習データに対して行った重回帰分析の結果得られた式である。この式を実際の学習データに適用すれば試験偏差値の推定値が得られるが、授業期間中の全学習データに対して適用するのでは結局授業期間が終わってからの推定となり、成績不振兆候者を早期に発見するという本来の目的に沿わない。そこで、各授業終了時までの累計データに対して重回帰式 A および B を適用した。ある単元が終了した時点までの、コンパイル回数および成功数の合計値から算出した「コンパイル成功率」、授業中のみでなく授業時間外の評価も含めた FB システムでの総合評価の平均値を「FB 評価値平均」、これまで出題された例題・練習問題のうち、正常なコンパイルと実行が完了している課題の割合を「課題達成率」として重回帰式に適用し、得られた推定値のうち低い方をその時点における「推定試験偏差値」とした。本手法では推定時に任意のしきい値を設定し、下位者を判定するが、下位判定には以下の2つの方式を用いた。推定試験偏差値を設定したしきい値を下回る学習者を「単純下位者」とする単純下位判定方式(以下、単純判定)と、それまでの推定で単純下位者と判定された回数が平均値を上回る学習者を「累積下位者」とする累積下位判定方式(以下、累積判定)である。この「単純下位者」および「累積下位者」を成績不振兆候者とする。単純判定では、その時点の推定値のみで判定を行うため、過去の判定結果は考慮しないが、累積判定では、その時点までの単純下位判定回数をもとに下位判定を行うため、過去の判定結果を考慮する。

提案手法を2019年度の学習データに適用し、有効性を検証した。FB システムの本格的な利用を開始した第4回の授業からのデータを用いた。また、第9回の授業は中間試験にあたり、演習を行っていないため、推定から除外した。

図4に2019年度の的中率および抽出率を示す。縦軸は、各回での任意のしきい値(50~35)における結果の平均値である。累積判定では、過去の推定結果が考慮されるため、最初の数回以降は下位判定される人数の変動が少なくなり、的中率、抽出率ともに回を追うごとに安定して推移しており、単純判定よりも比較的安定していることがわかった。的中率は累積判定よりも単純判定の方が優位である。初回の推定である第4回の授業から、中間試験直前の第8回の授業までは約40~60%とやや変動が大きいですが、中間試験結果をもとに補正を行った第10回以降は、80%前後で安定して推移している。一方、抽出率は単純判定よりも累積判定の方が優位であり、第5回の授業から中間試験実施前までという早期の段階でも、80%前後と高い値となった。中間試験後は補正によりさらに高い値となっている。

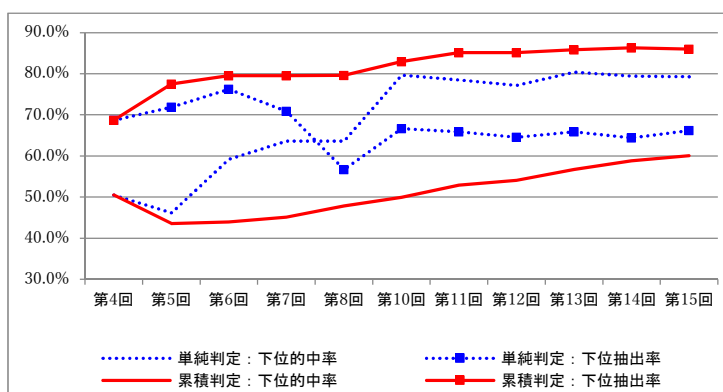


図4 2019年度の的中率および抽出率

以上より、本手法の適用により、成績不振兆候者を早期に発見するという目的に対して、一定の効果が得られたと考える。的中率のみでは単純判定が、抽出率のみでは累積判定が優位な結果となっており、実際の授業における運用中に成績不振兆候者の抽出を行う場合は、両方式の併用が望ましい。

5. 主な発表論文等

〔雑誌論文〕 計0件

〔学会発表〕 計1件（うち招待講演 0件 / うち国際学会 0件）

1. 発表者名 宇野健
2. 発表標題 Web上で利用可能なプログラミング演習システムのログデータを活用した演習状況の可視化システムの開発と運用
3. 学会等名 情報処理学会第82回全国大会
4. 発表年 2020年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

<p>プログラミング演習システムログイン画面（授業で使用中・ユーザ認証が必要） http://mis.pu-hiroshima.ac.jp/programming/ プログラミング演習システムログイン画面（授業で使用中・ユーザ認証が必要） http://mis.pu-hiroshima.ac.jp/programming/puh/ プログラミング演習システムログイン画面（授業で使用中・ユーザ認証が必要） http://mis.pu-hiroshima.ac.jp/programming/puh/</p>
--

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

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