

令和 2 年 5 月 19 日現在

機関番号：16201

研究種目：基盤研究(C) (一般)

研究期間：2015～2019

課題番号：15K01075

研究課題名(和文) プログラミング学習環境のスマートデバイス対応の研究

研究課題名(英文) Research on extension of programming learning environments for smart devices

研究代表者

香川 考司 (Kagawa, Koji)

香川大学・創造工学部・准教授

研究者番号：50284344

交付決定額(研究期間全体)：(直接経費) 2,800,000円

研究成果の概要(和文)：ビジュアルプログラミングライブラリーであるBlocklyを使用し、ネストが深くなりがちな非手続き型言語向けの、スマートデバイスに対応するためのブロックを作成した。Webアプリケーションのプログラミング言語処理系に対するバックエンドとしてJava仮想マシンを用いるWappenLiteとDocker Engineを用いるWappenLiteDockerを実装し、サーバーで安全に実行できるように改良した。Haskellの構文解析ライブラリーを用いて、テキスト形式のコードからブロック形式のコードを生成するアプリケーションと、学習者の提出したソースコードにコメントを追加するアプリケーションを実装した。

研究成果の学術的意義や社会的意義

ブロックなどグラフィカルな方式を用いて、多肢選択や穴埋めだけではなく様々な形式のプログラミング問題を出題することが可能になる。演習に時間をかけることが難しい非手続き型言語に対して、文法だけではなく、意味の理解に焦点を当てることも可能になると考えられる。WappenLiteやWappenLiteDockerを用いて、マイナーな言語やライブラリーにカスタマイズした言語処理系用バックエンドを容易に提供できる。結果としてより多くの教育者が魅力的な学習用コンテンツを提供すること・学習者の行動と解答を解析することが可能になると考えられる。

研究成果の概要(英文)：Using Google Blockly, a block-based visual programming library, we created blocks for non-procedural languages that tend to be deeply nested, to accommodate the small screens of smart devices.

We implemented WappenLite using Java Virtual Machine and WappenLiteDocker using Docker Engine as backends of Web applications to the programming language processors, and improved them so that they can run securely on server machines.

Using a parser library in Haskell, we implemented an application to generate block code from text code and an application to add comments to the source code submitted by learners.

研究分野：計算機科学

キーワード：e-ラーニング Webアプリケーション プログラミング教育

様式 C - 19、F - 19 - 1、Z - 19 (共通)

1. 研究開始当初の背景

(1) ワープロ・表計算・メーラーなどさまざまなアプリケーションが Web アプリ化され、この流れはプログラム開発環境にも及んでいた。当時、CodePad (codepad.org)、ideone (ideone.com) などがよく知られており、Scratch (scratch.mit.edu) などの教育用プログラミング環境も Web アプリとして提供されていた。しかし、これらはメジャーな手続き型のプログラミング言語には対応しているものの、教師が個々にライブラリーなどをカスタマイズしたり、マイナーな関数型などのプログラミング言語に対応させたりするのは難しかった。そこで、主に Java 仮想機械ベースのプログラミング言語の処理系に対応可能な Web ベースのプログラミング学習支援システムのバックエンドを開発し、WappenLite として公開していた。また、スマートフォン・タブレットなどのスマートデバイスが普及し、高校生、大学生がスマートフォンを持つことが当たり前ようになっていた。スマートデバイスから WappenLite の一部機能を利用することができれば、講義形式の授業や自宅などでの予習・復習でも、さまざまな形式の問題を出題することができる考えた。

(2) 教師やティーチングアシスタント (TA) が、スマートデバイスを利用して、演習の授業中に学習者からの質問に答えることができる。例えば、学習者の提出したソースコードに対してコメントすることなどが考えられる。Ajax 上のホワイトボードアプリはいくつも存在したが、プログラミング言語の文法を理解しているわけではないので、意味のあるまとまりを手早く選択することができなかった。従来、JavaScript 上で構文解析器を作成し実用的な効率で動かすのは難しかったが、Emscripten (emscripten.org) 等を使うことより、C のコードから JavaScript に変換することで、ブラウザ上で構文解析器を動かすことができるようになっていた。

2. 研究の目的

(1) それまでに開発していた Web ベースのプログラミング学習支援環境に、スマートフォン、タブレットなどの新しいクライアント側のハードウェアに対応するための機能を加え、プログラミング学習コンテンツを提供できるプラットフォームへ進化させる。演習形式の授業だけでなく、講義形式の授業や自宅などでの予習・復習でも利用できるようにする。プログラミングの講義形式の授業の場合、多肢選択問題だけでなく、スマートデバイスでさまざまな形式の問題を出題・解答できるようにする。

(2) また、関連して、教師や TA が演習の授業中にスマートデバイスを利用して、学習者の提出したソースコードにコメントする作業を支援するためのシステムを開発する。

(3) このようなプラットフォーム上に、実際にプログラミング、コンパイラ、プログラミングパラダイムなどの授業で利用できるコンテンツを作成する。

3. 研究の方法

(1) 画面が小さく、物理的キーボードを持たないスマートデバイスで文字の入力を最小限に抑えるために、Scratch のようなブロックベースのユーザーインターフェイス (UI) を改良する。本研究の主な対象となる非手続き型の言語では一般に“式”が複雑になり、ブロックが深い入れ子になってしまう。そのため、狭い画面では有効ではないことが準備段階でわかっていた。そこで、深い入れ子をもった式を、ブロックベースの利点を保って編集するための UI を工夫する。

(2) ブロックベースのコードは一般的に教師がコードの見本を用意するのに大きな手間がかかるため、通常のテキスト形式のソースコードからブロックベースのコードを生成する必要がある。そのための構文解析器を Emscripten などの JavaScript 上で動作するプログラミング言語処理系を利用して作成する。また、このようなブラウザ上で動作する構文解析器を利用して、教師や TA がスマートデバイスから、学習者の提出したソースコードにコメントするシステムを構築する。

(3) もともと WappenLite はサーバー側でもクライアント側でもプログラムを実行できるように設計されていたが、大半の学習者が Java 仮想マシンの言語をクライアント側で実行することを想定していた。しかし、スマートデバイスで利用する場合はクライアント側で実行できず、サーバー側で実行する必要がある。このため、サンドボックスを強化し、長時間動作時の安定性を高める必要がある。また、サーバー側で実行できるソフトウェアに制約がある場合に備えて、JavaScript 上での処理系など WappenLite 以外のプログラミング言語処理系の実行方法を調査する。

4. 研究成果

(1) 【Blockly によるユーザーインターフェイス】当初 CodeMirror (codemirror.net) と jsTree (www.jstree.com) などのライブラリーを組み合わせて構造化エディターをプロトタイプ実装し動作を確認した。しかし iOS や Android では一部の機能が動作しないなどの問題があった。また当初期待したほど表示がコンパクトにならなかった。そのため、開始前に想定していた独自

のエディターの実装は断念し、ブロック方式のビジュアルプログラミングエディター構築用のライブラリーである Google Blockly (developers.google.com/blockly) を使用することにした。まず、C 言語用ブロックの作成と小さい画面に対応するためのカスタムブロックの開発を行った。また、C 言語のソースコードから、Blockly のブロックを生成するシステムを、当初は C 言語上の構文解析器生成系である Bison と、C 言語から JavaScript へのコンパイラーの Emscripten を利用して実装した [1]。このシステムのプラットフォームは、のちに構文木の処理を強力に記述できるように Haskell の language-c-quote ライブラリー (github.com/mainland/language-c-quote) によるものに切り替えた。

また Python と JavaScript のジェネレーター関数、関数型言語 Scheme のラムダ式、関数型言語 Haskell のパターンマッチングや do 記法・リスト内包表記、論理型言語 Prolog のユニフィケーション、字句解析器生成系 Flex の正規表現などさまざまなパラダイムのプログラミング言語や領域特化言語に対して、もともとの Blockly には用意されていなかった、それぞれの文法に適合したブロックを作成した。ブロックの入れ子が深くなり過ぎて画面が狭くならないよう、Blockly の mutator という機構を利用してブロックの変形を実装した [2]。また、Python を対象とする Blockly アプリケーションでは、インポートされた関数に対する汎用的なブロックを用意して、外部ライブラリーのインポートを可能にするブロックメニューの拡張方式を実装した。

(2) 【教師・TA 用のユーザーインターフェイス】学習者の提出したソースコードに対して構文解析をサーバー側で実行し、結果をブラウザに送って、教師によるコメントの入力を支援するシステムを実装した [3]。このシステムの構文解析部は当初は Eclipse CDT (www.eclipse.org/cdt) の構文解析器を利用していたが、拡張性を考慮して、Haskell の language-c-quote ライブラリーによるものに切り替えた。

ソースコード中の位置を素早く指定するために、CSS セレクターに類似の記法でコメント対象箇所を指定できるようにした。またスマートデバイスの特徴を活かすために SVG 形式を利用して手書きによるコメントも入力できるようにした。

(3) 【WappenLite の改良】研究開始当初 WappenLite は主に Java アプレットから起動されることを想定していた。しかし、主要なブラウザが Java アプレットのサポートを打ち切ったため、アプレットからの起動機能を廃止するとともに、ブラウザとの通信方法を見直す必要があった。このとき複雑になり過ぎていた部分があったので再設計した [4]。また、スマートデバイスで実行されるブラウザから利用できるように、WappenLite をサーバー側での実行に適合させるための改良を加えた。Java 仮想機械 (JVM) 上で実行されるプログラムがサーバーをダウンさせないように、無限ループにつながる可能性のあるコードの部分にチェック用のコードを挿入して、一定回数以上実行されると停止するような Java バイトコードの改変を行う機能を実装した。この目的に当初採用を予定していた Apache BCEL (commons.apache.org/proper/commons-bcel) は開発が止まっており、最新の JVM バージョンの中間コードに対応していないため、同様の機能を持つ OW2 ASM ライブラリー (asm.ow2.io) を利用した。またタイムアウト処理など、他のユーザー作成プログラムの暴走防止の方策も取り入れた。同じ JAR ファイルに対して二回目以降の起動を早くするためのキャッシュを導入した。

(4) 【WappenLiteDocker の実装】研究開始前に想定していなかった状況の変化として、アプレットの廃止など Java プラットフォームが勢いを失ったこと、Docker (www.docker.com) などの仮想化技術が進歩したことなどがある。そのため、学習者が作成したプログラムを実行するためのサンドボックス提供のために Docker をバックエンドとして用いる WappenLiteDocker というシステムを実装した [5]。このシステムは Docker コマンド起動のオーバーヘッドを避けるため、Docker API を利用しており、プログラミング言語処理系の実行に必要なと考えられる最小限のコマンドのみを Docker エンジンに渡すようになっている。ファイルの送受信や WebSocket による処理系との対話的な通信などに対応している。さらに WappenLiteDocker から起動できるイメージを事前に指定したものに制限する、WappenLiteDocker から起動した以外のコンテナに対してコマンドを送らない、などセキュリティー上の制限を実装した。これらにより、Java 仮想機械ベースの処理系を持たないプログラミング言語も利用可能となった。

(5) 【JavaScript 上の言語処理系の調査】サーバー側で WappenLite や WappenLiteDocker のようなプログラムが実行できない場合でも、ある程度のプログラミング学習コンテンツが用意できるように、DoppioJVM (doppiojvm.org), Emscripten (emscripten.org), Haste (haste-lang.org), ClojureScript (clojurescript.org), Kotlin (kotlinlang.org), TeaVM (teavm.org), BiwaScheme (www.biwascheme.org), jsProlog (ioctli.org/logic/prolog-latest) などの JavaScript 上の言語処理系を調査し、これらを利用する Web アプリケーションを実装した。大規模のプログラムを実行する場合には実行速度は期待できないが、例題としてよく使われる程度の初歩的で小規模なプログラムを実行するには大抵の場合十分であることが判明した。

(6) 【プログラミング学習用のコンテンツの作成】Haskell の language-c-quote ライブラリーを

利用し、学習者の提出した C 言語のソースコードにプログラム可視化のための改変を加えるプログラムの作成に利用した。これを WappenLiteDocker と組み合わせてプログラム可視化システムを実装した。また、JavaScript 上で動作するプログラミング言語処理系を利用して CPU スケジューリングアルゴリズムや再帰下降構文解析・有限オートマトンなど計算機科学のトピックスの可視化システムを開発した [6, 7, 8]。

<引用文献>

- [1] Yuto Yamagata and Koji Kagawa: “A Web-based Learning Support System for the C Language with Automatic Generation of Blockly Programs”, EDMedia + Innovate Learning 2018, Amsterdam, Netherlands, June 2018.
- [2] Yuya Sano and Koji Kagawa: “Design of a Programming Environment for Non-Procedural Programming Languages using Blockly”, The International Journal of E-Learning and Educational Technologies in the Digital Media (IJEETDM), Vol. 5, 2019, pp. 93–101.
- [3] 木村 光星・香川 考司: “構文解析を用いた C 言語指導コメント支援システムの構築”, 教育システム情報学会 (JSiSE) 2018 年度 第 4 回研究会, 2018.
- [4] Koji Kagawa: “Redesign of WappenLite: a Localhost Web Application Framework for Web-based Programming Environments”, Workshop on Computation: Theory and Practice (WCTP 2017), Osaka Univ. Nakanoshima Center, Japan, Sep. 2017, 8pages.
- [5] Koji Kagawa, Haruhiko Nishina, and Yoshiro Imai: “WappenLiteDocker – A Interface Program between a Web-Browser and a Docker Engine”, The Third International Conference on Electronics and Software Science (ICESS 2017), Takamatsu, Japan, July 2017, pp. 152–155.
- [6] Kyohei Nishiyama, Koji Kagawa, and Yoshiro Imai: “Development of a CPU Scheduling Algorithms Visualizer using JavaScript”, IEEJ Transactions on Electronics, Information and Systems, Vol. 137, 2017, pp. 1641–1648.
- [7] Kenta Ohashi and Koji Kagawa: “Web-based Learning Support System for Recursive Decent Parsing using Haste”, The Fourth International Conference on Electronics and Software Science (ICESS 2018), Takamatsu, Japan, Nov. 2018, pp 20–24.
- [8] Yuya Asano and Koji Kagawa: “Development of a Web-based Support System for Object Oriented Programming Exercises with Graphics Programming”, 18th International Conference on Information Technology Based Higher Education and Training, Magdeburg, Germany, Sep. 2019.

5. 主な発表論文等

〔雑誌論文〕 計2件（うち査読付論文 2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 Imai Yoshiro, Sauvion Lorkan, Messias Valentin, Kagawa Koji, Doi Shunsuke, Hara Shinya, Ando Kazuaki, Hattori Tetsuo	4. 巻 138
2. 論文標題 Improvement of e-Learning System for Network Study toward International Students	5. 発行年 2018年
3. 雑誌名 IEEJ Transactions on Electronics, Information and Systems	6. 最初と最後の頁 1444 ~ 1450
掲載論文のDOI（デジタルオブジェクト識別子） https://doi.org/10.1541/ieejeiss.138.1444	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 Nishiyama Kyohei, Kagawa Koji, Imai Yoshiro	4. 巻 137
2. 論文標題 Development of a CPU Scheduling Algorithms Visualizer using JavaScript	5. 発行年 2017年
3. 雑誌名 電気学会論文誌C（電子・情報・システム部門誌） IEEJ Transactions on Electronics, Information and Systems	6. 最初と最後の頁 1641 ~ 1648
掲載論文のDOI（デジタルオブジェクト識別子） 10.1541/ieejeiss.137.1641	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 -

〔学会発表〕 計19件（うち招待講演 0件 / うち国際学会 9件）

1. 発表者名 佐野 裕也, 香川 考司
2. 発表標題 Blockly を用いた多言語プログラミング学習支援環境の構築
3. 学会等名 教育システム情報学会（JSiSE） 2018年度 第4回研究会
4. 発表年 2018年

1. 発表者名 木村 光星, 香川 考司
2. 発表標題 構文解析を用いた C言語指導コメント支援システムの構築
3. 学会等名 教育システム情報学会（JSiSE） 2018年度 第4回研究会
4. 発表年 2018年

1. 発表者名 久保 俊貴, 香川 考司
2. 発表標題 演算子順位法に対するWebベース学習支援システムの開発
3. 学会等名 教育システム情報学会 (JSiSE) 2018年度 第4回研究会
4. 発表年 2018年

1. 発表者名 Yuto Yamagata and Koji Kagawa
2. 発表標題 A Web-based Learning Support System for the C Language with Automatic Generation of Blockly Programs
3. 学会等名 EdMedia + Innovate Learning 2018 (国際学会)
4. 発表年 2018年

1. 発表者名 Koji Kagawa
2. 発表標題 Comparison of Several Approaches to Provide Back Ends forWeb-based Programming Environments
3. 学会等名 Workshop on Computation: Theory and Practice (WCTP 2018) (国際学会)
4. 発表年 2018年

1. 発表者名 Toshiki Kubo and Koji Kagawa
2. 発表標題 Development of a Web-Based Learning Support System for Operator-Precedence Parsers
3. 学会等名 The Fourth International Conference on Electronics and Software Science (ICESS2018) (国際学会)
4. 発表年 2018年

1. 発表者名 Kenta Ohashi and Koji Kagawa
2. 発表標題 Web-based Learning Support System for Recursive Decent Parsing using Haste
3. 学会等名 The Fourth International Conference on Electronics and Software Science (ICESS2018) (国際学会)
4. 発表年 2018年

1. 発表者名 Koji Kagawa, Haruhiko Nishina and Yoshiro Imai
2. 発表標題 WappenLiteDocker - A Interface Program between a Web-Browser and a Docker Engine
3. 学会等名 The Third International Conference on Electronics and Software Science (ICESS2017) (国際学会)
4. 発表年 2017年

1. 発表者名 Koji Kagawa
2. 発表標題 Redesign of WappenLite: a Localhost Web Application Framework for Web-based Programming Environments
3. 学会等名 Workshop on Computation: Theory and Practice (WCTP 2017) (国際学会)
4. 発表年 2017年

1. 発表者名 大橋 健太・香川 考司
2. 発表標題 再帰下降構文解析に対するWebベース学習支援システムの開発
3. 学会等名 【C】平成29年電気学会電子・情報・システム部門大会
4. 発表年 2017年

1. 発表者名 仁科 陽彦・香川 考司
2. 発表標題 Dockerを用いたプログラム実行環境の開発
3. 学会等名 【C】平成29年電気学会電子・情報・システム部門大会
4. 発表年 2017年

1. 発表者名 山形 悠人・香川 考司
2. 発表標題 Haskellによるパーサーを使用したC言語用Blocklyプログラム生成の自動化
3. 学会等名 情報処理学会 コンピュータと教育研究会 142回研究発表会
4. 発表年 2017年

1. 発表者名 Kyohei Nishiyama, Koji Kagawa, Yoshiro Imai
2. 発表標題 A Scheduling Algorithms Visualizer using JavaScript
3. 学会等名 World Conference on Educational Media & Technology (ED-Media 2016) (国際学会)
4. 発表年 2016年

1. 発表者名 Kyohei Nishiyama, Koji Kagawa, Yoshiro Imai
2. 発表標題 E-Learning system for scheduling algorithms visualization in JavaScript
3. 学会等名 第6回香川大学・チェンマイ大学合同シンポジウム2016
4. 発表年 2016年

1. 発表者名 山形 悠人・香川 考司
2. 発表標題 Webベースグラフィカルプログラミングエディタを用いたC言語学習支援環境への問題提示機能の実装
3. 学会等名 教育システム情報学会第41回全国大会
4. 発表年 2016年

1. 発表者名 Kyohei Nishiyama, Koji Kagawa, Yoshiro Imai
2. 発表標題 Design and Implementation of a Scheduling Algorithms Visualizer using JavaScript
3. 学会等名 The Second International Conference on Electronics and Software Science (ICESS2016) (国際学会)
4. 発表年 2016年

1. 発表者名 島川 大輝・香川 考司
2. 発表標題 C-Helperを用いたWebベースのC言語開発環境の構築
3. 学会等名 教育システム情報学会第40回全国大会 (JSiSE2015)講演論文集
4. 発表年 2015年

1. 発表者名 西山 恭平・香川 考司
2. 発表標題 EmscriptenとD3.jsを用いたC言語ポインタのWebベース動的可視化システムの開発
3. 学会等名 教育システム情報学会第40回全国大会 (JSiSE2015)講演論文集
4. 発表年 2015年

1. 発表者名 Takahiro Suetomo, Koji Kagawa
2. 発表標題 A Program Visualization System using 3D Still Graphics
3. 学会等名 4th International Conference on Information Technology Based Higher Education and Training (ITHET 2015), (国際学会)
4. 発表年 2015年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

WappenLite http://guppy.eng.kagawa-u.ac.jp/WappenLite/ WappenLiteDocker http://guppy.eng.kagawa-u.ac.jp/~kagawa/WappenLiteDocker/
--

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