

プログラミング上達を加速させる マスコットキャラクター付き統合開発環境の開発

浦山優樹^{†1} 齋藤雄人^{†1} 塩野駿^{†1} 竹林晃^{†1}
谷口翔一^{†1} 門地翔太^{†1} 江見圭司^{†2}

概要: 概要: ITエンジニアがプログラミングを始めて間もない頃、一人で統合開発環境のエラーメッセージと向き合ったつらい経験をする人が多い。ここでは初級レベルから中級レベルになるために、統合開発環境へのプラグインでプログラマーにアドバイスを行うことで上達を加速できるようにした

キーワード: 統合開発環境, 中級者, プログラミング, 学習

Development of Integrated Development Environment with Mascot Character to Accelerate Programming Progress

URAYAMA YUUKI^{†1} SAITO YUUTO^{†1} SHIONO TAKASHI^{†1}
TAKEBAYASHI AKIRA^{†1} TANIGUCHI SHOUICHI^{†1} MONCHI SHOUTA^{†1}
EMI KEIJI^{†2}

Abstract: When IT engineers have just started programming, they often have hard experience facing the integrated development environment error message by themselves. We have made it possible to accelerate their progress by giving advice to the program.

Keywords: Integrated development environment, intermediate-level programmer, learning.

1. はじめに

1.1 初心者向けのプログラミング支援

日本の情報教育は、森口繁一の「JIS FORTRAN 入門」[1]の圧倒的な成功により、Fortran のプログラミングができれば情報教育になるという時代が長く続いたようである。ところが、時代はプログラミング言語 C, そして Java へと変遷していき、次々と新しいプログラミング教育がでてきた。

手続き型のプログラミング言語であれば種々の実践報告はなされているが、最近では Java の事例もある[2]。一方、今世紀になって、オブジェクト指向も当たり前になりつつあり、ガニエの教授方略[3]をもとに分析して、オブジェクト指向技術者養成のためのカリキュラムを提案した報告がでてきた[4],[5]。これ[5]の報告によると、言語情報、知的技能などが主として手続き型のプログラミングの学習で、認定方略まで含むとオブジェクト指向技術の習得になるといっている。これらの分類をもとにオブジェクト指向プログラミング固有の難しさの学習に関して報告するものもかなりある[6][7][8][9][10][11][12]。

ここまで、述べた実践はプログラミング入門者が初心者

になることをめざしたものから、一步踏み込んで簡単なソフトウェア設計まで行うことを指向している実践もあり、ガニエの教授方略[3]でいえば、認知的方略のみならず、態度の育成まで考えている。

1.2 京都コンピュータ学院でのカリキュラム

専修学校京都コンピュータ学院のコンピュータサイエンス学系(school of Computer Science)では、一年次に Java の初級とアルゴリズムが必修であり、2 年次には Java アプリケーション開発とオブジェクト指向システム設計 (UML) が必修となっている。まさに文献[4],[5]を参考にしたカリキュラムとなっている。だから 1 年次終了時点では、文献[4],[5]でいう言語情報を終わっている。つまり、Java プログラミング初級レベルになっている。専門学校ではプログラミングの時間数が大学よりも多いのと、受講学生はコンピュータ専門学校へ入学するという時点でプログラミングを学習することは当たり前だと考えているため、入門者むけに「なぜプログラミングをするのか」みたいな態度[3]の育成はそれほど重要ではない。

それよりも 2 年次でプログラミングで開発していく上で、目的意識を失う学生が多いと分析を学生(江見以外の筆者)自らがおこなった。

初心者は脱したのに、その後何か自分で作るとなると壁があるということがある。まず、知っている世界が狭く自分が作るものが何を学べば、何を調べれば作ることができ

^{†1} 京都コンピュータ学院
Kyoto Computer Gakuin

^{†2} 京都情報大学院大学
The Kyoto College of Graduate Studies for Informatics

るのかわからない。中級者以降（個人製作）こそ孤独な戦いであり、そこをサポートできるシステムを制作したいという動機があった。以下のような点に集約される。

- (a) エラーメッセージがよくわからない。
- (b) リファレンスがよく読めない。
- (c) これが何に使えるかわからなかった
- (d) オブジェクト指向のメリットがわからない。

このうち、(d)は2年次以後、授業をうけて、開発経験が増えると自然に解消する。(c)はアンドロイドアプリ開発などを行えば、ようやく理解できるようになってきたということである。ところが難しいのは、(a)、(b)である。江見以外の筆者らは、これを友人経由でなんとかか身につけたという経緯がある。これを体系的に行いというのが本プロジェクトのモチベーションである。そこで本プロジェクトでは主なターゲットをプログラミング初心者ではなく中級者となった。

1.3 ターゲット分析

ここでターゲットとなるユーザを詳しく分析した。

例外メッセージの内容をデバッグに活用し適切な例外処理ができるようになることを目標に定めた。また、プログラミング中級者の定義を情報処理推進機構 IPA の実施する基本情報技術者試験、午後問題の Java プログラミングを解けるようになった程度の学習レベルと定義した。

Java の例外処理について基本情報技術者試験がどのような形で出題しているのかについて少し触れる。基本情報技術者試験の過去5年の午後試験のプログラミングの問題参照し扱われている Java の例外処理の内容を分析した結果、大きく分けて以下の4種に分類することができた。ここでの H30 春とは平成 30 年度春試験のことである。

- 不正な値が指定されたとして例外を送出する(H30 春,H29 春,H28 秋,H28 春,H27 秋,H25 秋)
- 検査例外の対処として try-catch を用いて例外処理を行う(H26 秋, H25 秋)
- try-catch の後、例外を握りつぶしている(H26 秋)
- throws を用い呼び出し元に処理を任せる(H30 秋)

いずれもそのソースコードを記述した者（本システムで当てはめると学習者）が予期している例外であり、予期せぬ例外に対する対処の方法を問う問題は存在しなかった。そこで、本システムに予期せぬ例外である実行時例外に関する情報をもとにアドバイスを行う機能があれば、学習者のさらなるプログラミングスキルの上達が見込めると考えた。

2. 設計・実装

2.1 設計の概要

プログラミング学習の補助を行うデスクトップマスコミ付き開発環境、「Ghost-Interface-Pairprogramming-System (以下、GIPS)」を開発した。GIPS は主に例外についてのアドバイスを提供し、プログラミング上達に不可欠な「例外メッセージの情報をデバッグに活かす」ことができるように学習者を誘導する。さらに、GIPS は既存のデスクトップマスコミソフトウェア、何か[13]の仕様に準じている。そのため他の開発者が拡張することも容易である。本項では主に GIPS の設計、実装について報告する。最終的にはペアプログラミング機能は実装できなかった。

先ほど述べたように、本システムの開発の動機として、初心者は脱したのに、その後何か自分で作るとなると壁があるということがある。まず、知っている世界が狭く自分が作るものが何を学べば、何を調べれば作ることができるのかわからない。中級者以降（個人製作）こそ孤独な戦いであり、そこをサポートできるシステムを制作したいという動機がある。これは、1.2 節の(a)～(d)で分析したように GIPS プロジェクトメンバーのプログラミング略歴をさかのぼっていく上で生まれたものである。

さて、デスクトップマスコミがユーザの補助として用いられていた例に、Microsoft エージェント[14]がある。その一例である Office アシスタントは、ユーザの入力した文字列でヘルプを検索し、表示するもので、ユーザは自分が何を知りたいのかを検索の文字列として考え、それをシステムに与える必要があり。ユーザは何がわからないのか、何をしたいのかを正しく理解し、検索の文字列を考えるスキルが必要とされた。

また、本システムと同じくデスクトップマスコミを用いて学習者の補助を行う研究の例として、汎用的な TA 代替エージェントの検討・開発もあり[10][15]、この場合では画像認識と知識データベースに基づく聞き取りにより利用者の状況をシステムが把握する形になっていた。

どちらの場合でも、学習者の補助を行うためにはまず学習者の状況をシステム側が把握する必要があることが分かる。

本プロジェクトではプログラミング上達のための一要素として例外メッセージの情報（図1参照）をデバッグに活用できるようになることに着目。ユーザがデバッグ中に遭遇した例外（図2参照）をシステム側が把握し、デスクトップマスコミからアドバイス(図2参照)を受けることができるシステム、GIPS を制作した。

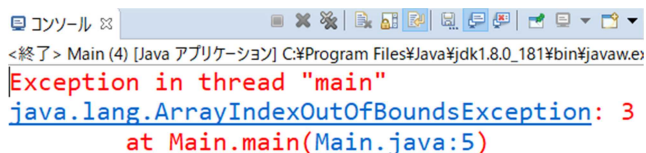


図1 実際に発生した例外の例

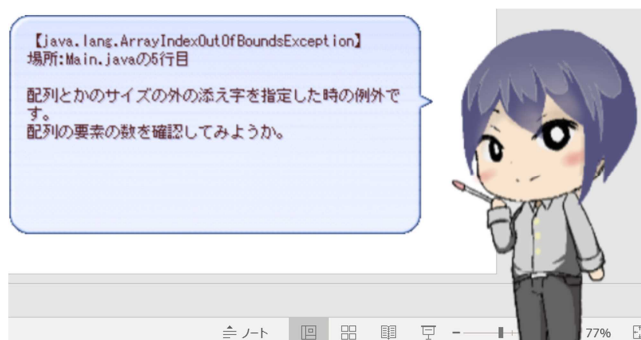


図2 “図1”の例外に対するアドバイス

GIPSは、学習者のコーディング中の情報を取得し、デスクトップマスケットに送信するIDEプラグインと、IDEプラグインから得た情報に基づいてアドバイスを行う何かの仕様に準拠したデスクトップマスケットの二つのソフトウェアがこれまた何かの仕様に定義されているSSTP[16]によって通信することで成り立つシステムである。(図3参照)

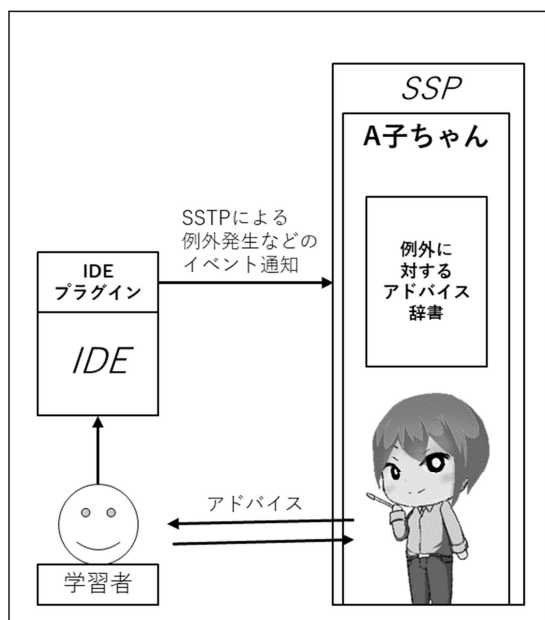


図3 システム構成図

なお、GIPSにおいて作成したデスクトップマスケットであるA子ちゃんとは何か用キャラデータの形で作成されており、何か互換ソフト「SSP」[17]で実行されることになる。

2.2 IDEプラグインの実装

IDEプラグインはVisual Studio, Eclipseに対応したものが、それぞれC#, Javaで実装されている。Visual Studioに関しては2005向けの実装が存在した[18]ため、それを参考にした。IDEプラグインはIDEの起動時、デバッグ実行時にユーザによる例外ハンドリングがなされなかった場合にデスクトップマスケットにイベントをSSTPで送信する。SSTPの送信内容の例としては以下のような内容がある。

```
// TCP ポート 9801 番で送信される
NOTIFY SSTP/1.1
Charset: UTF-8
Sender: GIPS-for-Eclipse
Event: OnGIPSExceptionOccurred
Reference0: Java
Reference1: java.lang.NumberFormatException
Reference2: Main.java
Reference3: 13
Reference4: 数値としての解釈に失敗
```

NOTIFYはデスクトップマスケットにイベントを通知する(図4)ことができるSSTPリクエストメソッドであり、通知するイベント名と共に付加情報をReference*(*)は0から始まる通し番号)として指定できる。GIPSで定義した、例外発生を知らせるOnGIPSExceptionOccurredイベントには、以下の情報を付加情報として送信するよう定義している(図4参照)。

- 例外が発生したプログラミング言語
- 例外名
- 例外が発生したソースコードのファイル
- 例外が発生した行数
- 例外の説明

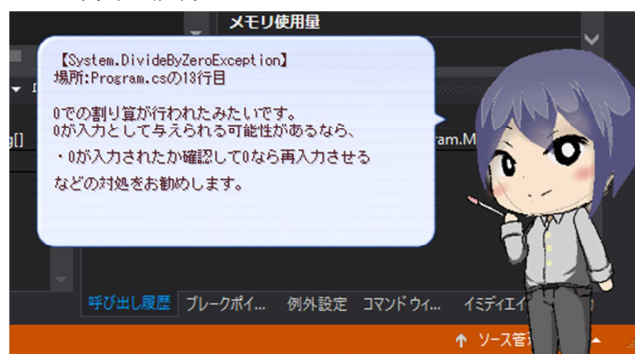


図4 イベントを受け取ったデスクトップマスケット

Eclipse, Visual Studioはそれぞれ異なるプラットフォームであるが、どちらも共通のイベントを通知するようになっており、かつそのイベントの仕様はWeb上に公開している。そのため、他の開発者が任意のIDE・エディタのプラグインを開発し、GIPSに対応させることも可能である。

2.3 デスクトップマスコットの実装

デスクトップマスコットは何か用キャラデータ記述言語, YAYA[19]で記述されており, A 子ちゃん, B 子ちゃんと名称がついている(図 5 参照).



図 5 A 子ちゃん(右) B 子ちゃん(左)

学習者は, このマスコットキャラクターを通じてアドバイスを受け取ることになる. システム内では, IDE プラグインから届く SSTP のイベント名および付加情報を取得, アドバイスを表示させる. アドバイス表示部の YAYA による実装の例は以下のとおりである.

```
OnGIPSExceptionOccurred
```

```
{  
    _language = reference[0]  
    _exceptionName = reference[1]  
    _fileName = reference[2]  
    _lineAt = reference[3]  
    _reason = reference[4]
```

```
    “%( _language) で書かれたプログラム  
    で%( _exceptionName)って例外が出たよ. ¥n この例外が  
    起こった理由は%( _reason)だって. ¥n%( _fileName)  
    の%( _lineAt)行目を確認してみるといいかも.”
```

```
}
```

この実装例では IDE プラグインより送信された情報をそのまま表示しているが, 送信された付加情報をもとにデスクトップマスコット側で定義したアドバイスを表示することもちろん可能であり, A 子ちゃんはそうように実装されている.

こちら, 何かと呼ばれるデスクトップマスコットのプラットフォームの仕様に基づいていることで, 他の開発者が自身の開発する何か用キャラデータに組み込むことも容易となっている. 具体的には YAYA を用いている何か用キ

ャラデータなら上記の実装を追加するだけで可能である.

2.4 未実装例外への対応

1.3 節で述べたように情報処理技術者試験では「実際に実装されたそのソースコードを記述した者 (本システムで当てはめると学習者) が予期している例外であり, 予期せぬ例外に対する対処の方法を問う問題」は存在しなかったのである.

本システムでカバーしている Java の例外数は 23 種であり, そもそもライブラリやフレームワーク等を用いて開発を行うであろう, 中級者が会えるライブラリやフレームワークに依存する例外は, 例外クラスが独自で定義できるという点からしても全てをカバーすることは事実上不可能だといえる. そのため, 本システムではアドバイスが実装されていない例外に対しては, 取得できる限りの情報とさらなる情報を得るための検索エンジンへのリンクを用意するという形で対応(図 1 参照)することにした.

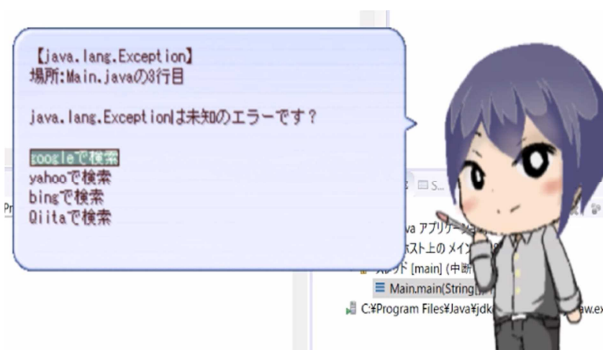


図 6 未実装例外への対応

3. 公開とその結果

3.1 インターネットへの公開

プロダクトページを作成, そこで紹介する形で GIPS 公式何かキャラデータとして A 子ちゃんを, GIPS 対応 IDE プラグインとして GIPS-for-Eclipse, GIPS.NET で公開した.

プロダクトページの公開後, SNS での広報用に Twitter アカウントを作成し公開した旨を報告, GIPS プロジェクトメンバーの Twitter アカウント(<https://twitter.com/GhInPaSy>)でもその報告を引用, 投稿した (図 6 参照).

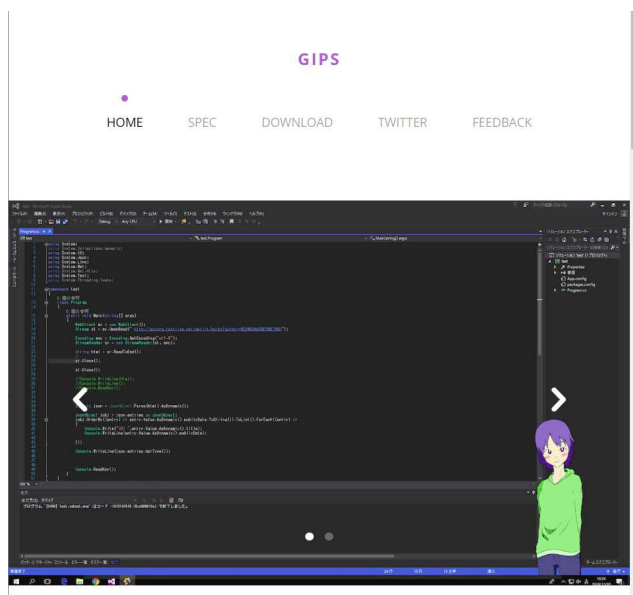


図6 GIPS 公開ページ

投稿後、得られた感想を以下にまとめる。

- 他の IDE (JetBrains IntelliJ Idea など) にも対応してほしい
- 自身で対応 IDE を増やすことができるのか
- プログラムを書く動機の一つになりそう
- 自身の何かキャラデータでも対応したい

このような感想が得られた理由として、プロジェクトメンバーの Twitter 上でのつながりが IT エンジニアなど、既にプログラミングが比較的にできる者が多かったためだと考えられる。

3.2 本学 1 年次学生への公開

前項での結果を受け、インターネット上への公開だけではプロジェクトメンバーの繋がり上、GIPS のターゲットであるプログラミング学習者へのアプローチとしては弱いと考え、本学、京都コンピュータ学院 コンピュータサイエンス学系の 1 年次学生 5 名 (全員が Java を学習し始めて約 10 ヶ月であり、授業で基本情報技術者の Java プログラミング問題を解いている) に試用してもらった。

その際、Eclipse/Java 版を自由に使ってもらい、現在の GIPS の例外アドバイス辞書でどこまでカバーできるか、知っている限りの例外を起こすプログラムを書いてもらうよう指示した。

試用してもらった後、得られた感想を以下にまとめる。

- 技術が (こんなことができるのかという点で) 面白い
- 自分たちの学校はアニメ好きの人が多いため、キャラクターものは受けがよさそう
- 今は読めるようになったが、例外メッセージを読める前はとても助かりそう

- 右下より左下の方が作業の妨げにならず使いやすいかもしれない
- キャラクター同士の掛け合いがあると面白いと思う

この感想及び、発生させてもらった例外を GIPS がすべてカバーできたことから、現在の GIPS で補助ができるのはプログラミング学習を始めて半年程度ではないかと推察した。

4. おわりに

4.1 今後の展望

公開後のフィードバックを受けて、初学者向け、開発者向けそれぞれの観点で以下のような展望を考えた。

初学者向け

- Java, C#, VB の実習系の授業の受講者に継続的に利用してもらい、より多くのフィードバックを得る

開発者向け

- GIPS で定義するイベントの拡充
- 対応 IDE の拡充

そして例外メッセージをデバッグに活用できるようになった学習者に対しての補助と開発者向けへのイベントの拡充を同時に進められる案として、A 子ちゃんにデバッグツールとしての機能を持たせることも検討したい。

4.2 まとめ

Ghost-Interface-Pairprogramming-System と題して、プログラミング学習者をマスコットキャラクターによって補助するシステムを構築し、公開することができた。1.2 節で述べたように、「(a)エラーメッセージがよくわからない。」という状態は解決することができたと考え。一方、「(b)リファレンスがよく読めない。」という状況については、きちんと調査ができていないが、エラーメッセージがわかれば自然に解決する問題であると判断している。

一方、現状ではシステム名に冠している「ペアプログラミング」が実現できているとは言えず、それを実現するためにはソースコードからクラス設計の状況を取得するなどの拡張が必要である。ペアプログラミングが実現できれば、プログラミングしながら、開発効率もあげつつ、学習効率も上げることができるからである。

最後に、本プロジェクトで制作したソースコードや通信使用のドキュメント等の成果物は GitHub [20]

にて公開しており、自由に閲覧できるようになっている。今後もプログラミング学習者が一人で辛い思いをせずとも学習できるよう、さらなる改良を目指し、更新していきたい。

謝辞 本システムの作成にご協力頂いた京都コンピュータ学院 藤戸俊行先生、水井英二先生、渡邊昭義先生に、

謹んで感謝の意を表する。

参考文献

- [1] (a)森口繁一, JIS FORTRAN 入門 (上). 東京大学出版会. 1968, p.282, (b)森口繁一, JIS FORTRAN 入門 (下). 東京大学出版会. 1969, p.250, (c)JIS FORTRAN 入門 (上) (下) -コンピュータ博物館, http://museum.ipsj.or.jp/heritage/JIS_Fortran.html (2019 年 2 月 1 日閲覧)
- [2] 吉田 典弘, 和田 裕一, 邑本 俊亮, 堀田 龍也, 篠澤 和久. 一般情報教育におけるプログラミングのスキルの習得度とプログラミングの考え方の理解度の関係に関する検討. 情報教育シンポジウム論文集 2018.vol.2018, pp.126-133.
- [3] (a)ロバート・M. ガニエ, キャサリン・C. ゴラス, ジョン・M. ケラー, ウォルター・W. ウェイジャー, 鈴木 克明 (訳), 岩崎 信 (訳). インストラクショナルデザインの原理. 北大路書房. 2007, p.462 ; (b)Robert M. Gagne, Walter W. Wager, Katharine C. Golas, John M. Keller. Principles Of Instructional Design 5th. Wadsworth Pub Co.2004, p.387 ; (c) 坂元・水越・西之園 (代表編集)『教育学事典』実教出版, 分担執筆 (項目:教授方略) ■教授方略 Instructional Strategy,<http://www.gsis.kumamoto-u.ac.jp/ksuzuki/resume/books/1998b.html> (2019 年 2 月 1 日閲覧).
- [4] 松澤 芳昭, 中鉢欣秀, 岡田 健, 大岩 元. オブジェクト指向技術者養成のためのカリキュラム.情報処理学会研究報告コンピュータと教育 (CE) .2002.(2002-CE-064),pp.1-8.
- [5] 江見圭司. 手続き型言語からオブジェクト指向言語までのカリキュラムの提案. 情報処理学会研究報告コンピュータと教育 (CE) .2007.vol.123(2007-CE-092), pp.1 - 4
- [6] 松澤 芳昭, 青山 希, 杉浦 学, 大岩 元. 「目的の表現」に注目した オブジェクト指向プログラミング教育とその評価. 2003.123(2003-CE-072),pp.77-84.
- [7] 平澤 智明, 高野 辰之, 宮川 治, 北澤 由貴, 古澤 資栄, 小澤 諒. プログラミングの入門教育を対象とした概念学習システムの開発.研究報告コンピュータと教育 (CE) .2010.2010-CE-107, pp.1-8.
- [8] 古澤 資栄, 高野 辰之, 小濱 隆司, 宮川 治. プログラミング初学者を対象とした概念学習システム.研究報告コンピュータと教育 (CE) .2012-CE-113, pp.1-9
- [9] 土肥 紳一. 参照に重点を置いたオブジェクト指向プログラミング入門教育の工夫. 情報教育シンポジウム 2015 論文集, 2015, vol.2015, pp.9-16.
- [10] 大城 正典, 永井 保夫. Eclipse 視覚化プラグインによる総合的なプログラミング教育支援システム. 情報教育シンポジウム 2015 論文集, 2015, vol.2015, pp.23-30.
- [11] 松澤 芳昭, 保井 元, 杉浦 学, 酒井 三四郎. ビジュアル-Java 相互変換によるシームレスな言語移行を指向したプログラミング学習環境の提案と評価. 情報処理学会論文誌. 2014, vol.55, No.1, pp.55-57.
- [12] 中山 陽平, 掛下 哲郎 . プログラミング穴埋め問題における穴抜きの難易度と学生の解答過程のクラス分析.. 情報教育シンポジウム 2018 論文集, 2018, vol.2018, pp.166-173.
- [13] “何か - 仕様書 - うさださくら”.
<http://usada.sakura.vg/contents/specification.html> (2019 年 2 月 1 日閲覧),
- [14] “Microsoft Agent -Windows applications | Microsoft Docs”.
<https://docs.microsoft.com/ja-jp/windows/desktop/lwef/microsoft-agent> (2019 年 2 月 1 日閲覧)
- [15] “コンピュータを用いた学習を支援する汎用 TA エージェントの検討”.
https://www.jstage.jst.go.jp/article/kitasatoclas/21/0/21_39/_article/-char/ja/ (2019 年 2 月 1 日閲覧)
- [16] “何か - SSTP プロトコル”.
<http://usada.sakura.vg/contents/sstp.html> (2019 年 2 月 1 日閲覧)
- [17] “独立何か研究施設 ばぐとら研究所”.
<http://ssp.shillest.net/> (2019 年 2 月 1 日閲覧)
- [18] “Visual Studio を何かに対応させるムダ技”.
<http://d.hatena.ne.jp/AtelierDeMuguet/20070806/1186371088>

(2019 年 2 月 1 日閲覧)

[19] “整備班 - The Maintenance Shop -”

<http://ms.shillest.net/yaya.xhtml> (2019 年 2 月 1 日閲覧)

[20]

GitHub(<https://github.com/Ghost-Interface-Pairprogramming-System>)