

すご@ぷろ：双六をモチーフとしたビジュアルプログラミング言語

蛭田 雄一† 桑原 悟†,†† 柴原 一友†
但馬 康宏††† 小谷 善行†††

日本の情報教育・情報処理教育に関する提言 2005 では、国民全体に対して何らかのプログラミング教育が必要であるとしている。プログラミングはとても知的な活動であるが故に、その面白さを感じるまでに多くの時間を要する。多くの人がつまづく原因としては、キータイピングが億劫であることや、構文規則等を習得が難しいことがあげられる。そこで、本稿ではより考える過程に重点をおき、ソースプログラムを作成する段階から楽しめる学習環境「すご@ぷろ」を提案し実装した。本稿ではすご@ぷろの提案と、それらを約 80 名程度の学生に対し講義で実験を行った。その結果、半数以上の学生が一般的なプログラミング言語ではみられない双六のマスと並べて形を作ることを楽しんでいった。結論として、短い講義時間の中で約 80% 以上の学生に対し、プログラミングに対し興味を持たせることに成功した。

sugo@pro : A Visual Programming Language based on the dice game Sugoroku

YUICHI HIRUTA,† SATORU KUWAHARA,†,††
KAZUTOMO SHIBAHARA,† YASUHIRO TAJIMA†††
and YOSHIYUKI KOTANI†††

In the proposal 2005 about the Japanese information education / computer science education, it is assumed that some kind of programming education is necessary for the all Japanese. It need much tie because programming is very intellectual activity untill the human feels the fun. Reasons why many people fail in are probably troublesome key typing and the difficulty of learning sentence structure rules. Therefore we put an important point by this report in a process to think about more and we suggested learning environment "sugo@pro" to be able to enjoy when coding the source program and implemented it. In this paper, we tested them by a lecture for about 80 students. As a result, it enjoyed that many students arranged and sat square of the sugoroku and made something shape. In conclouision, we succeeded in having them be interested for programming more than about 80% students in short lecture time.

1. はじめに

日本の情報教育・情報処理教育に関する提言 2005¹⁾では、国民全体に対して何らかのプログラミング教育が必要であるとしている。一般的にはプログラミングは、専門的であり、高度で知的であるものと考えられている。

実際に理系の大学や高等専門学校などの高等教育機関であっても、講義についていけず挫折感を味わって

いる学生を多々見る。特に講師に対する学生の比率が多い大学では、解らないという学生が占める割合が著者の経験からも多く感じる。「好きこそ物の上手なれ」という”ことわざ”にもあるように、まず、好き・興味を持たせる状態にしなければ、プログラミング教育の未来はない。

プログラミング教育を行う研究としては、初等中等教育でのプログラミング教育を目的とした言語や環境について、既存の実用言語へ注目したものよりはプログラミングの概念の習得を目的にされている報告がある³⁾。また、昨今では高等学校においても「情報」という科目の中で、プログラミングが教えられており、それだけではなく大学入試においても「情報」の科目が試されている⁴⁾。

高等教育機関や中等教育程度のレベル向けの入門用プログラミング言語としては「若葉」がある⁵⁾。若葉の特徴としてはCライクな文法構造とプリミティブ型として数値データ型が一つに統一されている。また、

† 東京農工大学大学院 工学府 情報工学専攻
Dept. of Computer and Information Sciences, Graduate School of Technology, Tokyo University of Agriculture and Technology

†† 新潟国際情報大学 情報文化学部 情報システム学科
Dept. of Computer System, Niigata University of International and Information Studies

††† 東京農工大学 共生科学技術研究院 システム情報科学部門
Institute of Symbiotic Science and Technology, University of Electronic-Technology

初心者には意味を説明したいヘッダーファイルの概念の処理をブラックボックスにしてある。似た試みとして「Kid's Programming Language(KPL)」がある⁶⁾。KPLは特にCライクな文法構造だけではなく、ゲームプログラミングを行いやすくすることで、より学習者に興味を持ってもらおうとしている。それに加えて、KPLでは学習者だけではなく、市販のレベルのゲームまで実装を行うことが可能である。

学習支援の環境としては、プログラミング入門教育用学習環境PEN(Programming Environment for Novices)がある⁷⁾。PENはソースプログラミング入力フィールドの下にプログラムの入力支援ボタンを設けて、条件分岐、繰り返しなどの定型構文をソースプログラミング入力フィールド部分に自動生成し、キーボード操作を減らすことができる仕組みを持つ。PENはあくまでもプログラミング入門教育用学習環境であるため、言語ではない。PENではプログラミング言語としては、特別な説明なしに入試でも用いられている手順記述言語DNCLおよび本学の入試用手続き記述言語TUATLEに準拠している。

このように様々な試みがされているが、プログラミングは、テレビゲームのエンタテインメント性とは異なり、すぐに楽しいと感じるものではない。プログラミングのような知のエンタテインメントは一般的に習得するのに時間を要するため、その持つ本来の楽しさを経験する前に挫折してしまう人も多くいる。将棋や囲碁などに代表される思考ゲームも、プログラミングの特性に似ているものがあり、取得するのに時間を要する^{8),9)}。

したがって、ハードルを下げて様々な角度から面白さを提供することにより、プログラミングを多くの人に楽しいと思わせることが可能ではないか^{10),11)}。そこで、我々はすご@ぶる(sugo@pro)を提案する。本稿では、すご@ぶるの概要と、実際の講義で約80名程度の学生たちに対して実験を行った結果を述べる。その結果、半数以上の学生が一般的なプログラミング言語ではみられない双六のマスを並べて形を作ることを楽しんでた。結論として、短い講義時間の中で約80%以上の学生に対し、プログラミングに対し興味を持たせることができ、今後プログラミングを学習してみたいというポジティブな回答を得ることに成功した。

2. すご@ぶるのシステム概要

2.1 システム

一般的なプログラミング言語には、標準的に利用できるライブラリ関数はいくつか存在する(C言語でprintfやscanfにあたるもの)。しかし、すご@ぶるには、一般的なプログラミング言語であるようなライブラリ関数はいっさい用意されていない。すご@ぶるでライブラリ関数を利用するためには、教育者が教材

ライブラリを用意する必要がある(すご@ぶるのシステムは、起動時に教材ライブラリを読み込む仕様になっている)。つまり、すご@ぶる上から利用したいライブラリ関数はあらかじめ、教材で定義しておく必要がある。

すご@ぶるは、文法構造のみを持っていて、関数、条件文、繰り返し文などの概念を扱える(詳しくは4節で述べる)。教材ライブラリにはそれらの具体的な内容:

- 関数呼び出し
- 条件文の分岐条件
- 繰り返し文の繰り返し条件

を記述することで、すご@ぶる上で利用できる。今回、この3種類に絞った理由としては、プログラミングを行う上でプリミティブになるものだと考えたからである。

2.2 教材ライブラリ

教材ライブラリは教材を作成する教育者がすご@ぶる上から利用できる関数呼び出し、条件文の分岐条件、繰り返し文の繰り返し条件などを教材テンプレートクラスを継承し、実装を行う。画面構成も自由に教育者が決めることができる。教材ライブラリの記述には、テンプレートクラスがC#で書かれているので、それを呼び出す.NET Framework 2.0からなら、記述することができる。教育者は実装した教材ライブラリをDLL(Dynamic Link Library)にする。すご@ぶるのシステムは、そのDLLを起動時に呼び出し、環境の実行を行う。つまり、教材ライブラリは課題設定そのものと、それを解くための道具から構成されていると言える。

今回の報告においては、次の3節で述べる教材ライブラリを利用した。

3. 教材ライブラリ”コイン集めゲーム”

今回の報告においては、教材ライブラリ”コイン集めゲーム”を利用した。コイン集めゲームの仕様を以下に示した。

3.1 ゲーム説明

キャラクターがある部屋の中にいる。プレイヤー(学習者)は、キャラクターを動かして、部屋から脱出する(出口に向かう)。部屋にはコインが落ちており、キャラクターがコインの上を通過するとコインを拾うことが出来る。単に脱出するだけでは駄目で、問題ごとに設定されたスコアを取得しなければ、クリアできない。また、キャラクターの方向は上下左右の向きを持っておる。

3.2 コインの種類

部屋に落ちているコインは:

- 銀コイン: 1ポイント
- 金コイン: 5ポイント
- 黒コイン: -2ポイント

の3種類である。問題ごとに予め、落ちている場所が決まっている。

3.3 クリア条件

ゲームのクリアは脱出（出口に着く）するだけでなく、脱出したときにスコアが一定以上ある必要がある。つまり、クリアするためには、部屋に落ちているコインを取得しなければならない（部屋に落ちているコインは無造作では無く、所定の繰り返し処理を行うと高い得点が得られるように設計されている）。通常、脱出が出来れば、脱出時のスコアが2倍される。さらに、スケッチのマス（4節で説明するが、一般的なプログラミング言語でのソースプログラムの長さに当たるもの）の数を指定された数以下にすることで、脱出時のスコアが3倍される。

3.4 ゲーム画面

コイン集めゲームのゲーム画面の例を図1に示した。問題はチュートリアル3題を含めて、全部で8題ある（ゲーム画面中のキャラクタ画像やマップ画像は¹²⁾より利用させて頂いている）。

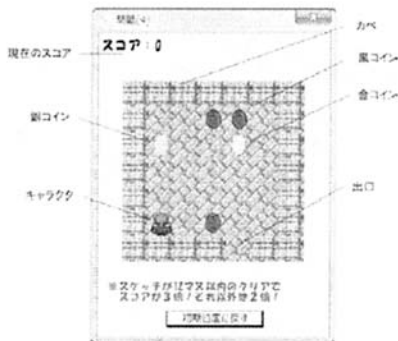


図1 ゲーム画面

3.5 定義した関数や分岐/繰り返し条件

この教材ライブラリには以下の関数や分岐/繰り返し条件：

- 関数呼び出し
 - － 前へ進む
 - － 右を向く
 - － 左を向く
- 条件文の分岐条件
 - － 前がカベ
 - － 左がカベ
 - － 右がカベ
 - － 後ろがカベ
- 繰り返し文の繰り返し条件
 - － 出口につくまで

を定義した。それぞれは日本語の意味のままである。学習者は上記の関数や分岐条件、繰り返し条件を用いて、プログラミングを行っていく。

4. すご@ぶろの設計

4.1 概要

すご@ぶろは「双六（の盤を作る過程）をモチーフにしたプログラミング」の略である。すご@ぶろは双六のマスと並べることで、プログラミングを行う環境であり、そのマスが意味を持ち計算機のように振る舞う機構である（サイコロは用いない）。すご@ぶろでは、一般的なプログラミング言語で頻発するキーボードでの誤入力を防止するためだけでなく、さらにマスを制約をなく自由に並べることで、絵画を行うような楽しさをプログラミングに付与したものである。

4.2 用語定義

ここでは、すご@ぶろで用いる用語を以下に定義する。

- マス (square)
 - － 何かの行動を行う最小単位である。一般的なプログラミング言語においては、「行」に当たるもの。
- スケッチ (sketch)
 - － マスを並べる領域、またその行為およびその成果物（ソースプログラムに当たるもの）をスケッチと呼ぶ。

4.3 マスの説明

すご@ぶろでは、スケッチを行うために、マスを用いる。マスには種類は、大きくわけて次のような種類がある。

4.3.1 はじまりマス

「はじまりマス」はプログラミングのエントリーポイントである。スケッチ上に1つ以上、配置する必要がある。C言語ではmain関数の始まり部分に相当する。スケッチ上には「はじまりマス」と表示される。図2に示す。双六においては、同様に「はじまり（スタート）」と等価である。

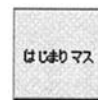


図2 はじまりマス

4.3.2 おわりマス

「おわりマス」はプログラミングの処理を途中で終了させるものである。スケッチ上に配置してもしなくても良く、また任意の個数配置することができる。「おわりマス」が実行されると、実行がそこで終了する。スケッチ上には「おわりマス」と表示される。図3に示す。双六においては、「おわり（ゴール）」と等価である。



図 3 おわりマス

4.3.3 おこないマス

「おこないマス」は C 言語でいう関数呼び出しである。すなわち、教材ライブラリに定義されている何かの動作を行う。「おこないマス」を図 4 に示す。図 4 は：

```
文;
```

の擬似コードと等価である。双六においては、状態によらず強制的にイベントが起こるマスと等価である。



図 4 おこないマス

4.3.4 よりみちマス

「よりみちマス」は C 言語でいう条件分岐 (if-else) である。「よりみちマス」だけの単体利用はできず、ユーザがメニューから選択した条件文の真の条件と偽の条件がセットになっている。条件には予め教材ライブラリで定義されたものだけが利用できる。

「よりみちマス」を図 5 に示す。図 5 は：

```
if (～のとき) {
} else {
}
```

の擬似コードと等価である。双六においては、ある条件のときにイベントが起こるマスと等価である。

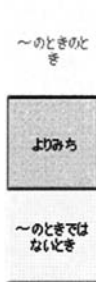


図 5 よりみちマス

4.3.5 もどりマス

「もどりマス」は C 言語でいう繰り返し文 (do-while) と繰り返し条件がセットになったものである。戻るマスの数は、マスをダブルクリックすることにより、学習者が変更することができる。

「もどりマス」を図 6 に示す。図 6 は：

```
do {
    : ← n 個
} while (!(～になる));
```

の擬似コードと等価である。双六においては、ある条件のときに「n マスもどる」というマスと等価である。



図 6 もどりマス

4.4 スケッチの具体例

ここではスケッチの一例を擬似コードと照らし合わせながら説明する。

4.4.1 例 1：おこないマス

「はじまりマス」を 1 個、「おこないマス」を 3 枚を利用した例を、図 7 に示した。図 7 は：

```
前へ進む;
前へ進む;
前へ進む;
```

の擬似コードと等価である。図 7 にあるように、自由に並べる方向も決めることができる。

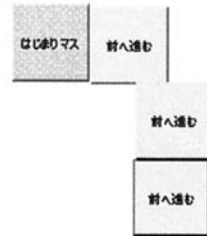


図 7 スケッチの例 (例 1)

4.4.2 例 2：もどりマス

任意のマスの数を「もどりマス」で戻るようにした例を、図 8 に示した。図 8 は：

```

前へ進む;
do {
  前へ進む;
  前へ進む;
} while(!出口につく);

```

の擬似コードと等価である。「もどりマス」は「出口につくまで2マスもどる」になっているので、直前の実行される2マスが繰り返される。

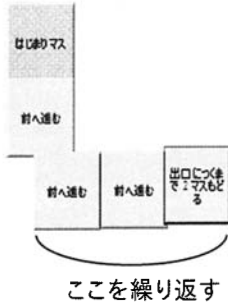


図 8 スケッチの例 (例 2)

4.4.3 例 3：よりみちマスともどりマス

「よりみちマス」で分岐した例を、図 9 に示した。図 9 は：

```

do {
  前へ進む;
  if (前がカベ) {
    右を向く;
  } else {
    前へ進む;
  }
} while(!出口につく);

```

の擬似コードと等価である。図 9 では「よりみちマス」の分かれた先にマスが存在しているが、必ずしもマスが存在する必要はない。マスがない場合は、空(φ)を実行したことに等価である。「よりみち」という言葉からも解るように、その先を実行し道が途絶えた時点で、「よりみちマス」に戻り、継続の処理が行われる。

4.5 システム画面

すぐ@ぶろの起動後の画面を 図 10 に示した。スケッチを描画するための (a) 入力ウィンドウと、実行結果を表示するための (b) 出力ウィンドウからなる。

4.6 スケッチの実行と停止

図 10 のシステムメニューから「スタート」を押すと、ソフトウェアがスケッチを解析し実行を行う。実行はリアルタイムに行われるが、学習者が実行の様子



図 9 スケッチの例 (例 3)

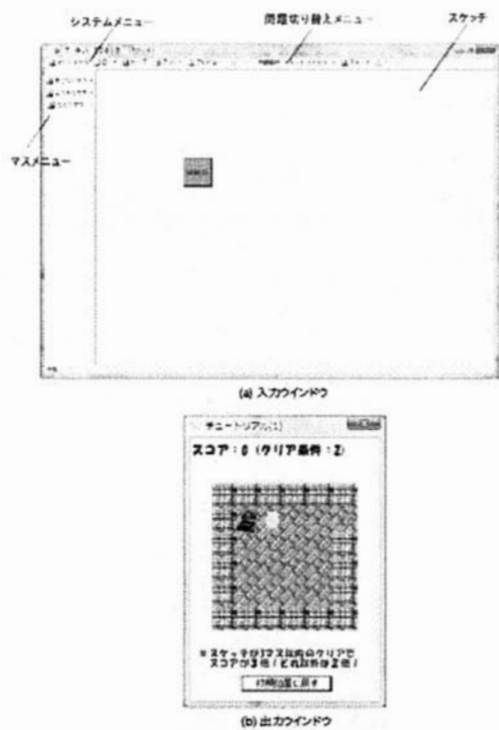


図 10 システム画面

を確認できるように、 n 秒ごとに段階的に実行される。 n は学習者が変更することができず、教育者が教育ライブラリを制作した段階で決定する。

途中で実行を強制的に停止したい場合は、システムメニューから「ストップ」を押す。

またスケッチの実行後は、対象である教育ライブラリのゲームをクリア・非クリアに関わらず、ダイアログで学習者に対し状況が通知される。実行後や強制停止後は、対象である教育ライブラリのゲームの画面は、

最後に描画された画面になるが、図 10 の出力ウインドウにある「初期位置に戻す」を押すことにより、初期状態に戻すことが可能である。「初期位置に戻す」を押さず、「スタート」をもう一度押したとしても、実行時に初期状態に自動的に戻る。

5. 実験

5.1 実験環境

サレジオ工業高等専門学校（以下、サレジオ高専）の 1 年生（15, 16 歳）の 2 クラス（1 クラス約 40 名）に対して 2007 年 12 月に実験を行った。講義の時間 100 分間で時間の内訳は：

- 13:20 ~ 13:50 説明
- 13:50 ~ 13:55 事前アンケートの記入
- 13:55 ~ 14:40 演習時間（45 分）
- 14:40 ~ 14:50 事後アンケートの記入
- 14:50 ~ 15:00 講義まとめ

となっている。2 クラスは別々の日程で講義を行っている。サレジオ高専は、4 学科（機械電子工学科、電気工学科、情報工学科、デザイン工学科）から構成されているが、1 年次は性別・学科問わず混成クラスである。

演習時間には自由にスケッチ（プログラミング）を行ってもらった。講義中は私と講師、TA の各 3 名体制で行った。

5.2 アンケート内容

5.2.1 事前アンケート

事前アンケートでは、被験者自身の一般的な内容を聞いた。内容としては、性別、コンピュータの経験、ゲームの経験、プログラミングの経験、配属予定の専攻などである。

5.2.2 事後アンケート

事後アンケートは、すぐおぼろに関する内容を中心に聞いた。内容としては、各課題が解けたか、面白かったか、解りやすいか、親しみやすいか、ゲームらしいか、得意か等や、今後プログラミングを勉強したいかである。面白さ、解りやすいかなどのパラメータは主観的な評価を 4 段階で行った。また、自由記述形式で評価の理由および講義を受けての感想を述べさせた。

5.3 結果

5.3.1 性別

有効回答数は 82 名で、男性が 71 名、女性が 11 名であった。図 11 に示した。

5.3.2 配属専攻

特に全体に大きな差が見られなかった。図 12 に示した。

5.3.3 プログラミング経験

プログラミング経験は、情報工学科の学生が C 言語の基礎（条件分岐等）まで経験している場合が多かった。図 13 に示した。

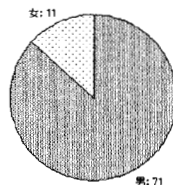


図 11 性別

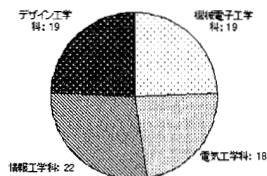


図 12 配属専攻

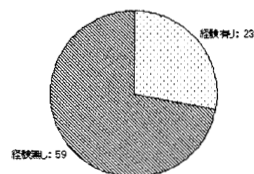


図 13 プログラミング経験

5.3.4 プログラミングを学習したいか

今回の講義を受けて今後プログラミングを学習したいかという質問の結果を図 14 に示した。8 割以上が「学んでみたい」「少し学んでみたい」のポジティブな評価になっていた。

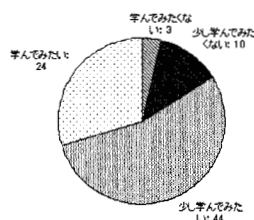


図 14 今回の講義を受けて今後プログラミングを学習したいか

5.3.5 主観的な評価項目

「面白さ」「分かり易さ」「親しみやすさ」などの、主観的な評価項目を図 15 に示した。それぞれの項目は既に述べている通り、4 段階で評価されている。もっとも良い評価を 3、もっとも悪い評価を 0 としたと

きの、有効回答数分の平均値を計算し、図示してある。概ね、良い評価に、データが集中していることがわかる。

また、平均値と標準偏差を表1にまとめた。概ね、標準偏差も安定しているの、データとしては信憑性が高い。

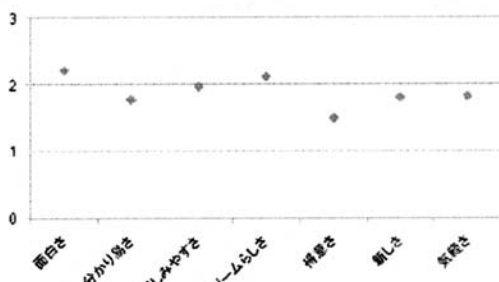


図 15 主観的な評価項目

表 1 主観的な評価項目の平均値と標準偏差

評価項目	平均値	標準偏差
面白さ	2.22	0.73
分かり易さ	1.77	0.79
親しみやすさ	1.97	0.73
ゲームらしさ	2.12	0.78
得意さ	1.49	0.78
新しさ	1.80	0.73
気軽さ	1.82	0.84

6. 考 察

6.1 アンケート結果に対して

講義の時間としては、100分間あったが説明やアンケート等の時間を除くと、実際の演習時間は45分間という短い時間であった。しかし、情報工学科の学生だけではなく、多くがプログラミング経験の無い機械電気工学科、電子工学科、デザイン工学科の学生に、プログラミングのイロハを教えることができ、80%以上の学生がプログラミングに興味を持たせることができたのは評価できるのではないかと。特に主観的な評価であっても、多くの学生たちに「面白かった」「親しみやすい」と評価された点は、特筆すべきことである。

6.2 実験の様子

スケッチという概念をすご@ぶろで提案したことにより、実験中に通常のプログラミング言語では起こりえない面白い様子が垣間見ることができたのでそれを報告する。一般的なプログラミング言語の演習であっても、自分の製作したソースプログラムを見せるという動作は良く行われている。現実問題として、学部の

講義中には学生間の中でソースプログラムを見せ合っているのもよく見かける。多くは単にデータの複製(コピーアンドペースト)のみで行われているのではないかと。

本実験においても、同様のことが演習中に行われていた。しかし、スケッチという自由にマスを配置するというクリエイティブな活動がある。すご@ぶろにおいて、「こんな面白い置き方が出来たよ」、「こんな不思議な置き方でゴールするよ」などというスケッチの置き方自体を楽しみ、それを仲間に見せ合うという様子を見受けられた(単純な複製(コピーアンドペースト)ではなく、絵を描いた後にお互いの作品を鑑賞するようなイメージ)。たとえばスケッチで「?」マークのような形を取ったり、「@」マークのような形に配置している学生もいた。それだけではなく、一般的なプログラミング言語のように、「こんな短いマスだけでゴールできたよ」という点を目指す学生もいた。そのように、プログラミング言語というものに対し、少しでも様々な角度から楽しみ(エンタテインメント)を提供することができたのではないかと。このように、本稿では、それぞれ個人の興味の対象が違っているのは自明であるのだから、プログラミングが得意になるきっかけとして、プログラミングの間口を広げ、その有効性を示せた。

7. おわりに

本稿では、双六をモチーフにしたビジュアルプログラミング言語「すご@ぶろ」を提案し、そのシステムを実現した。そして、狙いのとおり、マスを並べるという楽しさを提供したことにより、情報系以外の学生に対してもプログラミングの面白さを伝えることができた。

今後の展望としては、より多くのプログラミング教育に携わっている方に使って頂けるように、教材ライブラリを容易に開発できるように整備をしていきたい。

謝 辞

本研究の実験は、サレジオ工業高等専門学校のご協力があり、行うことができました。講義担当の各担当教員の間を調整して下さいました同校 山澤 明子 先生、講義のサポートをして下さいました同校 専攻科 宮田 渉 君、山澤先生をご紹介して頂いた同校 専攻科 市村 洋 教授には感謝いたします。

参 考 文 献

- 1) 情報処理学会情報処理教育委員会:日本の情報教育・情報処理教育に関する提言 2005, 情報処理学会, 2005.
- 2) 桑原 悟, 蛭田 雄一, 柴原 一友, 小谷 善行:状態をもった内部表現でプログラムを保持するプログラミング教育環境の有効性の検討, 信学技報, vol.

- 107, no. 205, ET2007-25, pp. 5-10, Sep. 2007.
- 3) 兼宗 進, 中谷 多哉子, 御手洗 理英, 福井 眞吾, 久野 靖: 初中等教育におけるオブジェクト指向プログラミングの実践と評価, 情報処理学会論文誌 (トランザクション), Vol.44, No.SIG13, pp.58-71, 2003
 - 4) 長 慎也, 兼宗 進, 並木 美太郎, 辰己 丈夫, 久野 靖, 西田 知博, 小井土 政範: 「情報 B」の教科書比較—「手順的な自動処理」の観点から, 情報処理学会研究報告. コンピュータと教育研究会報告 Vol.2006, No.46, pp. 27-34, 2006
 - 5) 吉良智樹, 並木美太郎, 岩崎英哉: 初心者入門用言語「若葉」の言語仕様と処理系の実装, 情報処理学会論文誌 (トランザクション), Vol.40, No.SIG10, pp.28-38(1999).
 - 6) Jon Schwartz, Jonah Stagner, Walt Morrison: Kid's Programming Language (KPL), SIG-GRAPH 2006 Educational program, 2006
 - 7) 中村亮太, 西田知博, 松浦敏雄: プログラミング入門教育用学習環境 PEN, 情報処理学会コンピュータと教育研究会, Vol.2005, No.104, pp.65-71, 2005.
 - 8) 蛭田 雄一, 但馬 康宏, 小谷 善行: 一人思考ゲームにおけるゲームデザインの比較, 第 1 回エンターテイメントと認知科学シンポジウム (シンポジウム発表賞), pp.16-17, Mar. 2007.
 - 9) 蛭田 雄一, 柴原 一友, 但馬 康宏, 小谷 善行: 将棋におけるゲームデザインの比較, ゲームプログラミングワークショップ 2007, pp.120-124, Nov 2007.
 - 10) 市村 洋, 原田健, 岩佐千夏, 蛭田雄一, 吉野純一, 杉本文司: ゲーム的興味を演出する数学の事例—その 1—一代数曲線符号の数学構造から導かれる人物描画改竄鑑定プログラム—, サレジオ工業高等専門学校 研究報告書, pp.107-118, Sep 2006.
 - 11) 和田 洋一, 馬場 章: 日本人の「遊び心」「ものづくり」が「ゲーム」世界—に, 教育家庭新聞, http://www.kknews.co.jp/maruti/2007/news/080101_0a.html, Dec 2007.
 - 12) 秋山 博紀 (AKI) 著: akiroom, <http://akiroom.com/> (キャラクター画像)