

プログラミング学習の他教科への波及効果 ～数学と考える力を対象とした調査と検証計画～*

井戸坂 幸男, 足利 裕人, 紅林 秀治, 鎌田 敏之, 兼宗 進, 久野 靖
松阪市立飯南中学校, 鳥取県立鳥取工業高校, 静岡大学, 愛知教育大学, 一橋大学, 筑波大学
〒515-1411 三重県松阪市飯南町粥見 566
idosaka@gmail.com, ashi@logob.com, eskureb@ipc.shizuoka.ac.jp,
tkamada@aeu.ac.jp, kanemune@acm.org,
kuno@gssm.otsuka.tsukuba.ac.jp

概要

プログラミング学習を行うことで、コンピュータの原理などに関する理解が向上することは多く言われている。我々はそれ以外にも、プログラミング学習には「考える力」一般を養う効果があり、それによって他教科の学習にもよい影響を及ぼすのではないかと考えた。本稿では、2005年度にオブジェクト指向言語「ドリトル」を使い実施した、中学校技術・家庭科のプログラミング学習授業およびそれに関するアンケートについて報告する。アンケートから、生徒はプログラミングを通して、数学や考える力を学んだと実感していることが明らかになった。さらに、2006年度に実施を予定している、プログラミング学習が他教科(とくに数学)の学習にどのような影響を及ぼしているかを検証する試みについても述べる。

1 はじめに

筆者の一人は、10年前から中学校の技術・家庭科の授業においてプログラミング学習を取り入れてきた。これは、コンピュータが普及する中で、ソフトウェアがどんなもので、プログラマによってどのように作られているのかを体験させるために取り入れたものである。

そしてこの授業を行う中で、生徒たちはソフトウェアやコンピュータの原理を学ぶだけでなく、思考力や創造力の面でもよい影響を受けていることに気付くようになった。

そこで、2005年度の授業において、プログラミング学習が他の教科にも関連する一般的な学習にもたらす効果について、生徒を対象としたアンケート調査を行った。また、2006年度は、この結果をもとに、具体的な他教科への波及効果の検証を計画している。

本稿では、2005年度に実施した授業を紹介し、

実施した生徒のアンケートに基づき、プログラミングが他教科に及ぼす影響について考察する。また、2006年度における検証計画についても議論する。

2 2005年度の授業内容

2005年度に、三重県の松阪市立飯南中学校において18時間のプログラミング学習授業を実施した。授業は技術・家庭科「情報とコンピュータ」領域の一環として行い、3年生全2クラス、合計55名の生徒が受講した。

言語には教育用プログラミング言語である「ドリトル[1]」を使用した。ドリトルは、日本語で命令ができるなど、初心者にわかりやすく学習ができるように配慮されており、また、オブジェクト指向を積極的に取り入れている[2][3]。

授業は、表1のカリキュラムで実施した。以下でその概要について解説する。

2.1 タートルグラフィックス

最初の4時間は、ドリトルのタートルグラフィックス機能を用いてさまざまな図形を描きながら、基本操作やプログラムの基本を学習した。

*Programming Study Effect on Other Subjects, by Yukio IDOSAKA (Iinan Junior High School), Hiroto ASHIKAGA (Tottori Technical High School), Shuji KUREBAYASHI (Shizuoka University), Toshiyuki KAMADA (Aichi University of Education), Susumu KANEMUNE (Hitotsubashi University), Yasushi KUNO (University of Tsukuba)

表 1: 授業カリキュラム

単元	時間数
(1) タートルグラフィックス	4
(2) タイマーによるアニメーション	4
(3) ボタンによる対話的な操作	4
(4) 音楽演奏	6

具体的には、前進命令(あるく)と角度を変える命令(みぎまわり)を基本として三角形を描くところから始め、くり返し命令(くりかえす)や、描いた図形の移動(いどうする)などを学習する。図1は例題プログラムの1つで、繰り返しを使って三角形を描き、できた三角形を図形として色を塗り指定位置に置く*。

かめた=タートル!つくる。
 さんかく=「かめた!100ぼ あるく 120 ど
 みぎまわり」! 3 かい くりかえす
 ずけいにする (赤)ぬる。
 さんかく!100 100 いどうする。

図 1: 図形を描き配置するプログラム例

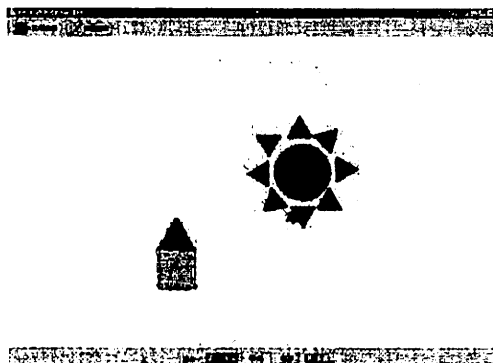


図 2: 生徒によるグラフィックス作品

ここでは、「プログラムは順序をもって実行される」こと、「くり返し命令を使うことで簡単に

*ドリトルの命令は、小学生の利用を考慮して、漢字とひらがなの両方が用意されている。中学生であれば当然これらの漢字は学んでいるはずだが、実際には漢字が苦手な生徒もいるため、今回の授業ではひらがなの命令を使用している。

同じことを繰り返せる」こと、「オブジェクトを作り、オブジェクトに命令できる」ことを学習した。また、プログラムの中で数学でも扱う図形(三角形、正方形等)の外角、座標、移動距離、回転角度について扱った†。

生徒はここまでの段階で、三角形、四角形、星、円、などの図形を組み合わせ、さまざまなグラフィックス作品を制作した。図2に生徒の作品例を示す。

2.2 タイマーによるアニメーション

次の4時間では、描いた図形を一定時間移動させたり、回転させるタイマーオブジェクトを学習した。ゲームソフトなどでキャラクターが動くしくみを体感することができる。図3はタイマーを使ったプログラムの例であり、時間とともにタートルが回転し、また図形が平行移動するものである。

かめた=タートル!つくる。
 さんかく=「かめた!100ぼ あるく 120 ど
 みぎまわり」!3 かい くりかえす
 ずけいにする (赤)ぬる。
 動き=タイマー!つくる。
 動き! 1 びょう さんかく 10 びょう じかん。
 動き!「かめた!15 ど みぎまわり。
 さんかく! -10 -10 いどうする」じっこう。

図 3: タイマーを使ったプログラム例

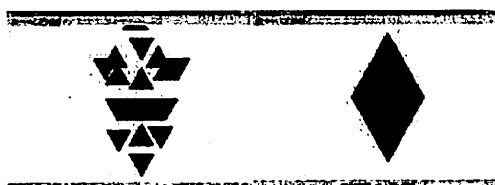


図 4: 生徒によるアニメーション作品

この単元では、タイマーによって図形の移動がゆっくりとくり返し実行されることを視覚的

†ドリトルの画面座標系はもともと、数学などでの学習に備えて、描画面の中央が原点(0,0)で右/上が正方向となるように設計されている。ただし、長さの単位は画面上のピクセルであるので、グラフを描くなどの用途であれば適切なスケーリングが必要となる。

にとらえることで、図形の回転移動と平行移動を学習できた。生徒はこの段階までで、図形が移動するアニメーション作品を制作することができるようになった。図4に生徒の作品例を示す。

2.3 ボタンによる対話的な操作

次の4時間は、クリックすると命令が実行されるボタンオブジェクトを使い、ボタンを押すことで画面に線や図形を描く作品(お絵かきソフト)を制作した。ここでは、メソッドの定義の方法を学習した。また、プログラムの中で直交座標を扱った。図5にボタンを使ったプログラムの例を、図6に生徒作品を示す。

かめた=タートル!つくる。
ボタン1=ボタン!"あるく" つくる。
ボタン1!-200 100 位置 150 50 大きさ。
ボタン1:動作="かめた! 50 ぼ あるく"。

図5: ボタンを使ったプログラム例

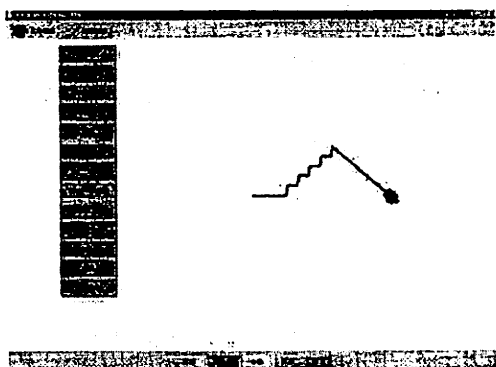


図6: 生徒によるボタンを使った作品

2.4 音楽演奏

最後の6時間は、今までのグラフィックスとは違う題材として、音楽を演奏するプログラムを学習した。具体的には、自作や既存の曲を題材に、いろいろな楽器を重ね合わせて演奏する作品を制作した。

ここでは、プログラムの複数の部分が並行して実行できること¹や、書き方の工夫によってプログラムをコンパクトで見やすくできることを学習した。図7にプログラム例を示す。

チューリップ=メロディ!作る。
チューリップ!
『どれみーどれみーそみれどれみれー』追加。
楽器設定=楽器!『クラリネット』作る
(チューリップ) 設定 演奏。

図7: 音楽を演奏するプログラムの例

3 アンケート調査とその結果

生徒が主観的に感じている学習効果を調べるために、2段階で調査を行った。具体的には、まず自由記述形式のアンケートを実施し、そこに現れた特徴的な事項をさらに細かく調べるために、選択式アンケートを実施した。以下でこれらについて解説/検討する。

3.1 記述式アンケート

自由記述式アンケートでは、『プログラミング学習をして役に立ったことがあれば記入せよ』という設問を提示し、生徒に自由形式で解答させた。実施時期はカリキュラム中の「(3) ボタンによる対話的な操作」が終わった段階である。自由記述ではあったが、回答で多数出された意見は大きく表2にある3種類に分類することができた。

表2: 自由記述アンケートの結果

回答	人数
A. 数学に関する理解	25
B. 考える体験	17
C. コンピュータに関する理解	8

1人が複数の分類に属する回答をした例は、14件あった。また、表2の分類以外の回答は14件、

¹サンプルには現れていないが、学習の進捗につれてメロディーと伴奏や打楽器を同時に演奏させられるようになる。

「なし」と回答した者は2件であった。具体的な記述内容の例を表3に示す。我々はC群の回答が多いことを予想していたが、実際にはA群やB群の回答が多かった。そこで、さらに詳しく調べるために選択式アンケートを実施した。

表 3: 自由記述アンケートの記述例 (一部)

A	XYの座標についてわかるようになった。
A	xyの関数が数学でやるよりよくわかった。
A	数学の図形の役にたった。
B	いろんな発想ができるようになった。
B	根気強く同じことができるようになった。
B	集中できるようになった。
C	ゲームがどんな風に作られているかわかった。
C	キーボードが速く打てるようになった。
-	タイマーのしくみが少しわかった。
-	ボタンを作って使うという面白さがわかった。
-	ボタンは難しくなくてわかりやすかった。
-	クリックしているんな図形がかけて面白かった。

3.2 選択式アンケート

選択式アンケートは、自由記述式アンケートの解答を整理し、項目ごとに「強く思う、思う、あまり思わない、思わない」の4段階で解答する形とした。設問の表現は自由記述式アンケートの解答にあった言葉をなるべくそのまま使うようにした。選択式アンケートはすべての学習が終了した時点で実施した。

以下に選択式アンケートの結果を、A群、B群、C群に属する設問ごとに分けて示す。「割合(%)」は、肯定的な回答(「強く思う」「思う」)を選んだ生徒の割合である。

A. 数学に関する理解

表4に、A群(数学)に関する質問群の結果を示す。(1)から、座標の理解については、ほとんどの生徒が学習効果として認めていることがわかる。(2)と(3)から、半分近くの生徒がプログラミング学習の方が数学の図形の授業よりわかりやすかったことがわかる。(4)と(5)から、生徒は関数に関しての学習効果はあまり感じていないことがわかる。

表 4: 選択式アンケート結果 (A)

項目	割合 (%)
(1) 数学の座標 (x座標 y座標 + や -) がわかるようになった。	68.7
(2) 図形の角度 (内角や外角) がわかるようになった。	41.2
(3) 数学で図形を学習するよりドリトルで「かめた」を動かしたほうがよくわかった。	41.2
(4) 数学の関数 ($y = ax$ など) の理解に役立った。	29.4
(5) 数学で関数を学習するより、ドリトルでタイマーで図形を動かす方がよくわかった。	29.4

表 5: 選択式アンケート結果 (B)

項目	割合 (%)
(6) 根気強く同じことができるようになった。	51.0
(7) 集中できるようになった。集中力がついた。	56.9
(8) 考える力がついた。	68.6
(9) 想像力が豊かになり、創作力が身についた。	52.9
(10) 難しいことでもあきらめず、自分で調べるようになった。	66.7

B. 考える体験

表5に、B群(考える力)に関する質問群の結果を示す。この群では、すべての項目において、半数を超える生徒がプログラミングの効果を肯定している。根気、集中、考える力、創造力など、考える体験についての学習効果があると生徒は感じていることがわかる。中でも、(8)については、7割近い生徒が肯定しており、生徒は考える力を学習効果として実感していることがわかる。

C. コンピュータに関する理解

表6に、C群(コンピュータ)に関する質問群の結果を示す。この群も、すべての項目で肯定

表 6: 選択式アンケート結果 (C)

項目	割合 (%)
(11) コンピュータソフトがどのように作られているかわかった。	62.8
(12) ゲームソフトの作り方がわかった。	64.7
(13) プログラムの構成や「どうやらこういう動作をする」ということがわかった。	74.5

する割合が高かった。全般に、ソフトウェアの仕組みを理解できたと感じている生徒が多いことがわかる。また、(13) から、ほとんどの生徒はプログラムの本質である「手順が自動的に実行されて行く」という概念を理解していると考えられる。

4 アンケートの考察

本節では、実施したアンケートについて考察し、課題について検討する。

A. 数学に関する理解

「(1) 座標」については、図形の移動、タイマーによる移動、ボタンの位置の決定などで XY 座標を指定しなければならないため、生徒は試行錯誤しながら座標を決めていった。数値と画面上の場所が視覚的に対応することから、数学が苦手な生徒において、座標の理解に結び付いた可能性がある。

「(2) 図形の角度」については、導入時に三角形や四角形を描かせる課題を扱ったため、生徒は試行錯誤しながら角度を決めていった。角度の値と画面上に描かれた角度が視覚的に対応することから、数学が苦手な生徒において、角度の理解に結び付いた可能性がある。

「(3) 図形の学習」の結果と数学の成績との関係を分析したところ、数学が苦手な生徒は「数学よりドリトルのほうが図形を理解できた」と回答する比率が高かった。

「(4) 関数」については、肯定する生徒の割合は多くなかった。関数をすでに理解している生徒がいたことと、理解できていない生徒にとってはタートルグラフィックスで描いた線と一次関数が結び付かなかった可能性がある。

一方、「(5) 関数の学習」と数学の成績との関係を分析したところ、数学が苦手な生徒は「数学よりドリトルのほうが関数を理解できた」と回答する比率が高かった。これは、プログラミングを通して座標や傾きを理解することにより、間接的に一次関数の理解に結び付いたと考えられる。

これらの結果から、プログラミング学習は、数学を苦手とする生徒にとって、図形や関数の理解を助ける効果があることがわかった。実際、「数学の座標や図形や関数がわかるようになった」と答えた生徒の自由解答には次のものが見られた。

- 「座標の xy のどちらが横の方かなどわかりました。」
- 「ドリトルのおかげかはわからないけど、一次関数ができるようになった。でも、グラフの傾きとかはドリトルでわかった。」
- 「関数とかグラフが今までよりは少しできるようになった。」

B. 考える体験

プログラミング学習の授業では、生徒は作品の制作を始めると黙々とパソコンに向かい作業を続ける。プログラムを作っては実行し、うまくいかないところを直したり、工夫する。授業中は、キーボードの音しか聞こえない。このような時間が何時間も続くことから、過半数の生徒が、「(6) 根気」「(7) 集中力」がついたと感じていることは納得できる。

「(8) 考える力」については、プログラミング学習では、順を追って考えたり、結果を予想して作業をする必要がある。また、プログラムを大きくまとめて考えたり、無駄な部分を省略することも必要になる。これらの学習体験からも、

生徒が「考えている」という実感を持つことができたことは納得できる。

「(9) 創造力」について、生徒に直接聞くと、「中学校の授業の中で、創造する場面は少ない。創造力がつくと思われる場面としては、国語の読み物で情景を考えるとときや美術で自由に描く場合くらいでほとんどない」と答えていた。今回のプログラミングの授業では、生徒が作りたい作品を想像し、実際に試行錯誤しながら作っていったことから、「創造性が養われた」と感じることができたものと考えられる。

「(10) 難しいことでもあきらめず…」については、授業では時間の関係もあり、すべての命令を扱うことができないため、マニュアルで調べるように指導した。自分の構想を実現するためには、教えられた命令だけでなく、自ら調べた命令も使うことが必要になってくる。このような指導方法の結果、自分で調べるようになったと思う生徒が多かったと思われる。

C. コンピュータに関する理解

プログラミング学習を行う目的の一つに「ソフトウェアの仕組みを理解させる」ことがある。制作する作品として、アニメーションやボタンオブジェクトを使ったお絵かきソフトを扱ったが、生徒たちはこれらの作品制作を通してソフトウェアの仕組みを理解することができた。特に、肯定している生徒の割合が高いことから、プログラミングがソフトウェアの仕組みを教えるための有効な手段であることを再確認できた。

プログラミング学習を行うことにより、ソフトウェアだけでなく、ネットワーク、電子メール、ロボット制御など、コンピュータに関するさまざまな理解が深まる学習効果が期待できる。

5 2006 年度の検証計画

2005 年度の授業経験およびアンケートの結果から、プログラミング学習が図形や関数に関する概念の形成を助ける効果があることが予想できた。そこで 2006 年度においては、数学におけ

る「図形や関数の概念」を扱う題材のうちから「一次関数」を取りあげ、この内容理解とプログラミング学習との関係について検証していくことを計画している。

数学における「関数」は、中学校で扱う他の題材に比べ、概念がとらえにくいもののひとつである。生徒に聞くと、「一次関数は難しいので嫌いだ」という生徒が多い。これは、単純な計算や視覚的にとらえられる図形などと違い、比例をはじめとする関数の概念が生徒の頭のなかに作られにくいところにあると思われる。

実際に筆者の一人が勤務する中学校（実験を行っている中学校）の数学の教員に尋ねたところ、例年約 4 割以上の生徒が一次関数を理解できない状況とのことであった。その裏付けのため、3 年生（昨年一次関数を学習済み、これからプログラミング学習を実施）を対象として一次関数の理解、思考能力⁵、科目の嗜好などを問う予備調査（テストとアンケート、図 8⁶）を実施した。その結果（表 7）を見ると、一次関数に関するごく基本的な問いでも極めて正答率の低い場合があり、研究というだけでなく生徒のために何らかの対策が必要ではとの考えを持った。この予備調査は内容をさらに改良の上、下記の実験時に 2 年生に対して実施する予定である。

一次関数の単元は 2 年生に割り当てられているため、2 年生の技術・家庭科の時間で並行してプログラミング学習を実施し、「一次関数」での学習効果が認められるかどうかを次のような対照実験によって検証することを計画した（上記予備調査の対象とした 3 年生のプログラミング学習も行うが、一次関数の学習は 1 年前に済んでおり時間がかかなり経過していることと、カリキュラムの都合で 2 クラスとも同時にプログラミング学習を行うので対照実験にならないため）：

実験概要：数学の「1 次関数」の単元と並行した

⁵ 大学入試センター「情報関係基礎」の過去の問題から特定の知識ではなく考える力を見ていると思われる問題を選んで易しく手直して作成した。

⁶ 調査用紙の検討も並行して行ったため問題の一部はクラスにより異なる。このため解答の母数も設問によって異なる。

第1回 プログラミング学習アンケート

年 月 日

次の問題は、プログラミング学習による学習効果を調べるためのものです。テストではありません。また、成績に関係することはありません。協力をお願いします。

1. アンケート (3分)

①パソコンによる授業が好きですか。 ア. 好き イ. 好きなら ウ. 嫌い エ. 嫌い
 ②パソコンの操作が得意ですか。 ア. 得意 イ. 得意なら ウ. 苦手 エ. 苦手
 ③プログラミング学習が好きですか。 ア. 好き イ. 好きなら ウ. 嫌い エ. 嫌い
 ④数学が好きですか。 ア. 好き イ. 好きなら ウ. 嫌い エ. 嫌い
 ⑤図形が得意ですか。 ア. 得意 イ. 得意なら ウ. 苦手 エ. 苦手
 ⑥図表 (比例、一次関数) が好きですか。 ア. 好き イ. 好きなら ウ. 嫌い エ. 嫌い
 ⑦関数 (比例、一次関数) が得意ですか。 ア. 得意 イ. 得意なら ウ. 苦手 エ. 苦手

⑧までに学習した数学の分野で、最も得意な分野を選んでください (3つ以内)。
 1年… ア. 正の数・負の数 イ. 文字の式 ウ. 方程式 エ. 比例と反比例
 オ. 平面図形 カ. 空間図形
 2年… ア. 式の計算 ケ. 連立方程式 ク. 一次関数 コ. 図形の調べ方
 サ. 図形と合同 シ. 確率

⑨までに学習した数学の分野で、最も得意な分野を選んでください (3つ以内)。
 1年… ア. 正の数・負の数 イ. 文字の式 ウ. 方程式 エ. 比例と反比例
 オ. 平面図形 カ. 空間図形
 2年… ア. 式の計算 ケ. 連立方程式 ク. 一次関数 コ. 図形の調べ方
 サ. 図形と合同 シ. 確率

⑩プログラミングを学習して、得意になったことや興味を持ったことがあれば書いてください。

(2)右の表のA, イの空欄にあてはまる数を書きなさい。

x	-2	-1	0	1	2	3	イ
y	-2	ア	1	3	7	11	

ア. わかりません

イ. わかりません

(3)一次関数 $y = \frac{1}{2}x + 3$ について、次の問に答えなさい。

① x が 4 のときの y の値を求めなさい。 わかりません

② x の値が -2 から 2 まで増加するときの y の増加量を求めなさい。 わかりません

(4)次の一次関数のグラフを書きなさい。

ア. $y = 3x - 4$

イ. $y = -\frac{2}{3}x + 1$

ウ. $y = 2$

3. 進捗的学習検証問題 (5分)

右の図のように2つの部屋が廊下でつながっています。ある人が多岐を渡って、入口から入り、廊下や廊下を通り、最後に出口から出ます。

廊下には、スタンプA(aと印字)とスタンプB(bと印字)があります。廊下は一方通行です。

部屋Aから部屋Bに移動する際 (廊下の廊下) では、スタンプAが押されます。この廊下は、部屋Bから部屋Aにはもどれません。スタンプは、左から廊下に押されているものとします。

(1)手紙に押したスタンプで、正しく読み取っていただけない場合は次の中から2つ選びなさい

ア. a イ. aab ウ. aba エ. aabab オ. abba

(2)手紙に押したスタンプの規則性について、正しければ○、間違っていれば×を記入しなさい。

ア. aが2回続くことがある。 わかりません

イ. スタンプは必ず奇数回押される。 わかりません

ウ. aが2回連続して押される。 わかりません

図 8: 予備調査 (テストとアンケート)

技術・家庭科の時間において、ドリトルによるプログラミング学習を行う生徒とそうでない生徒が半分ずつになるように分けて実施し、それらの間で「一次関数」の理解にどれくらい違いがあるかを調べる。

対象: 中学2年生 (A組24名、B組25名)

実験方法: 数学で、「一次関数」の単元を学習している時に、並行してA組はプログラミング学習を行い、B組はWebページの作成を行う。続いて、A組がWebページの作成、B組がプログラミング学習を行う。開始前、中間点、終了時点の3時点で「一時関数」の理解判定を含むテストを実施し、両クラスの理解度を比較する。

Webページの作成は、技術・家庭科において学習予定であり、なおかつプログラミング学習と順序を入れ換えても差し支えないために選んだものである。

実験の流れを図9に示す。ここでのプログラミング学習は、数学の直接的な題材を設定するわけではないが、タートルグラフィックスを中心として、プログラム学習の中に数学的な題材を適宜含めることを予定している。

検証テスト1~3については、教科書や副教材の問題集を参考に、適切と思われる数種類の問題を作成して出題する。この3回のテスト結果を学級別の正答数での比較、基礎学力の定着していない生徒と定着している生徒の点数比較などを行う予定である。

6 まとめ

プログラミング学習を通して、プログラムの方法が身につくだけでなく、それ以外のさまざまな能力を身に付けることが可能である。本稿では、タートルグラフィックスのプログラミング

表 7: 予備調査テスト部分集計

項目	正答率%
座標空間上に点を打つ	$\frac{29}{45} = 64.4$
座標空間上の点の座標を答える	$\frac{13}{15} = 86.7$
直線上の点数個を与える表で空欄となっている x/y 各一箇所を埋める	$\frac{28}{60} = 46.7$
一次関数の式から x に対する y の変化の割合を解答	$\frac{4}{45} = 8.9$
一次関数の式と x の値を与え y の値を解答	$\frac{1}{15} = 6.7$
一次関数の式を与えグラフを描かせる	$\frac{18}{45} = 40.0$
グラフを与え対応する一次関数の式を選択	$\frac{4}{15} = 26.7$
論理的思考問題 (複数の部屋をつなぐ一方通行の廊下を通り抜けるときの可能な順序に関する問題)	$\frac{28}{60} = 46.7$

を通して数学の理解との関係について考察した。

プログラミング学習が学習者の思考能力を増大させるという主張は [5] をはじめ古くからあるが、まだ決着のついた問題とは言えない。[4] は BASIC を学んだ大学生を対照群と比較し、手続き的な思考や言葉の理解では BASIC の学習が効果をもたらしたが、一般的な思考能力については対照群と差がなかったと報告している。しかし、この実験は学生の学年、使用している言語などの点で、われわれの実験とはかなり違った設定となっている。

われわれはもともと、現在の技術にマッチしたオブジェクト指向言語を用いることで、コンピュータやソフトウェアの原理を生徒に的確に理解させることを目的としてプログラミング学習を行って来た。しかし、この目的以外にも、プログラミング学習が他教科の学習や思考能力の向上などの点で効果をもたらし得るのであれば、そのことをきちんと把握し、生徒のために役立てて行きたい、というのがわれわれの立場である。

そのためにも、今後さらに多くの教科や教育内容との関連を検討し、実験などを通じて検証を進めていきたいと考えている。

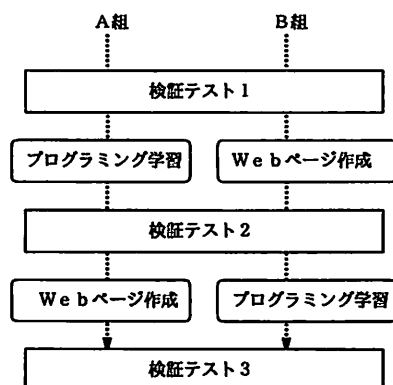


図 9: 実験の流れ

参考文献

- [1] 兼宗 進, 御手洗理英, 中谷多哉子, 福井眞吾, 久野 靖, 学校教育用オブジェクト指向言語「ドリトル」の設計と実情報処理学会論文誌: プログラミング, vol. 42, No. SIG 11 (PRO 12), pp. 78-90, 2001.
- [2] 兼宗 進, 御手洗理英, 中谷多哉子, 福井眞吾, 久野 靖, オブジェクトストーム: オブジェクト指向言語による初中等プログラミング教育の提案, 情報処理学会論文誌, vol. 43, No. 6, pp. 1610-1624, 2002.
- [3] 兼宗 進, 中谷多哉子, 井戸坂幸男, 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, 共著, 2003 年 10 月, 情報処理学会論文誌: プログラミング, vol. 44, No. SIG 13 (PRO 18), pp. 58-71, 2003.
- [4] Richard E. Mayer, Jennifer L. Dyck, William Vilberg, Learning to Program and Learning to Think: What's the Connection?, Communications of the ACM, vol. 29, no. 7, pp. 605-610, 1986.
- [5] Seymour Papert, Mindstorms, Basic Books, 1980.