

教育用オブジェクト指向言語「ドリトル」による 授業実施とその評価*

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*

筑波大学大学院 経営・政策科学研究科*

(有)エス・ラグーン† 松阪市立鎌田中学校‡ (株)アーマット§

〒112-0012 文京区大塚3-29-1*

{kanemune,fukui,kuno}@gssm.otsuka.tsukuba.ac.jp*

tina@slagoon.to† idosaka@logob.com‡ rie@armat.com§

概要

プログラミング言語「ドリトル」を利用して行なった情報教育を報告する。中学校第2学年の132人を対象に、技術・家庭科の情報基礎領域において、ドリトルを扱う授業を行なった。生徒たちは11時間の授業の中で、繰り返しやGUI部品の活用を含むプログラミングを体験することで、プログラミングの楽しさやソフトウェアの仕組みを学ぶことができた。評価は2種類のアンケートと2回の期末テスト、そして生徒の作品プログラムを分析することで行なった。

1 はじめに

2003年度から高等学校で教科「情報」の実施が開始されることに象徴されるように、「情報」の適切な理解や取り扱い方法を身につけることは我が国の初中等教育における重要な課題の1つとなってきた。情報教育のアプローチの1つとして、S. パパートの「MINDSTORMS」に代表される、プログラミングを学ぶことを通じてコンピュータの性質やコンピュータによる情報の取り扱い方を理解するという方法がある(プログラミング自体を学ぶことに主眼を置いた、いわゆる「情報処理教育」とは目的が異なることは注意しておきたい)。

しかし、これまで教育用プログラミング言語として広く使われて来た BASIC や LOGO は既に

20年以上前に設計されたものであり、これらを用いたプログラミングを体験しても、その成果物は普段生徒らが使っているソフトウェアとは全く隔たったものには見ええず、学習意欲を維持することは難しい。このため筆者らは、今日のソフトウェア技術において主要な位置づけを占めるオブジェクト指向の考えに基づいた教育用オブジェクト指向言語「ドリトル」を設計・開発し、これをもちいた情報教育の可能性について探求を行なっている。

2 プログラミング言語「ドリトル」

ドリトル [2] [3] [4] は教育用に設計されたオブジェクト指向言語である¹。簡潔な日本語による構文を採用しており、オブジェクトに呼び掛ける形でプログラムを記述できる。

図1に、ドリトルのプログラム例を示す。以下

K12 Education and Evaluation using "Dolittle" Object-oriented Programming Language, Susumu Kanemune, Takako Nakatani†, Yukio Idosaka‡, Rie Mitarai§, Shingo Fukui*, Yasushi Kuno* (Tsukuba University*, S.Lagoon Co.,Ltd.†, Kamata Junior High School‡, Armat Corporation§)

¹最新版はプログラミング言語「ドリトル」のサイト (<http://www.logob.com/dolittle/>) から入手できる。

カメ太=タートル!作る。
 カメ太!100 歩く 120 右回り 100 歩く 閉じる。
 三角形=カメ太!図形にする (赤) 塗る。
 時計=タイマー!作る 1秒 間隔 10秒 時間。
 三角形:ぐるぐる=「時計!「!36 右回り」実行」。
 実行ボタン=ボタン!作る。
 実行ボタン:動作=「三角形!ぐるぐる」。

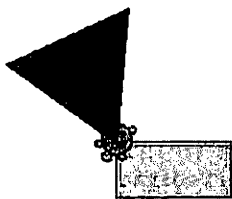


図 1: ドリトルのプログラムと実行例

ではこの例を使い、ドリトルの構文を解説する。

カメ太=タートル!作る。

“カメ太”というタートルオブジェクトを作っている。画面にはカメの姿をしたタートルオブジェクトが現れる。タートルオブジェクトはタートルグラフィックスを実現するオブジェクトであり、軌跡の線を描きながら画面を移動することができる。

カメ太!100 歩く 120 右回り 100 歩く 閉じる。

ドリトルでは、“!”でオブジェクトに呼び掛ける形でプログラムを記述する。この部分は、『カメ太! 100 歩歩いて、120 度右に回って、100 歩歩いて、元の場所に戻って』と読み下すことができる。実行すると画面に正三角形が描かれる。

三角形=カメ太!図形にする (赤) 塗る。

カメ太に“図形にする”を送り図形オブジェクトを作る²。そして、図形オブジェクトに“塗る”を送って色を塗り、“三角形”という名前を付けている。

時計=タイマー!作る 1秒 間隔 10秒 時間。

タイマーは決められた時間の間、一定間隔でプログラムを実行するオブジェクトである。ここでは“時計”という名前のタイマーオブジェクトを

²タートルが描いた線は、“図形にする”で明示的に切り放さない限り、タートルオブジェクトの一部になる。カメの尻尾が長く伸びた姿を想像すると理解しやすい。

作り、実行時間と実行間隔をセットしている。

三角形:ぐるぐる=「時計!「!36 右回り」実行」。

オブジェクトには、メソッド(埋め込まれた手続き)を定義できる。ここでは、三角形に“ぐるぐる”という名前のメソッドを定義している。このメソッドが実行されると、時計は“!36 右回り”を1秒間隔で10秒間(すなわち10回)繰り返し実行する。

実行ボタン=ボタン!作る。

“実行ボタン”という名前のボタンオブジェクトを作っている。画面には、ボタンの形をした GUI 部品が現れる。

実行ボタン:動作=「三角形!ぐるぐる」。

“実行ボタン”に、マウスで押されたときに実行するメソッドを定義している³。画面に表示された“実行ボタン”を押すことにより、三角形に定義された“ぐるぐる”というメソッドが実行される。その結果、三角形が画面上で1秒ごとに36度ずつ回転するアニメーションが表示される。

3 実施した授業

三重県松坂市立鎌田中学校では、第2学年の技術・家庭科の情報基礎領域の授業(必修で132名全員が受講)として2学期~3学期に11時間、ドリトル言語を用いたプログラミングの学習を実施した。表1に実施したカリキュラムを、図10に使用したテキストの例を、図2と図3に授業の様子を示す。

表 1: 鎌田中学校のカリキュラム

学期	時限	内容
2	1-2	タートルグラフィックス
	3-5	図形オブジェクトの生成と操作
3	6-8	タイマーによるアニメーション
	9-11	GUI 部品の活用

³いくつかのオブジェクトには、特別な名前のメソッドを定義することにより、ある条件が満たされたときにそのメソッドを実行することができる。たとえばタートルや図形オブジェクトでは、“衝突”という名前のメソッドを定義しておくことで、他のオブジェクトとぶつかったときの動作を記述できる。

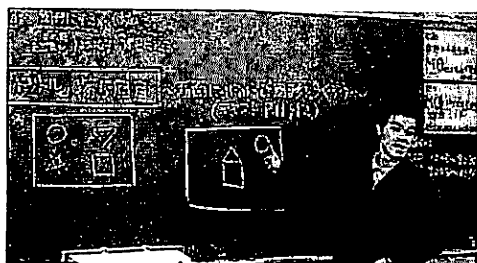


図 2: 授業の様子 (1)



図 3: 授業の様子 (2)

4 評価

客観評価の材料として、2回のアンケートと2回の定期試験を実施した。

4.1 毎時間のアンケート

授業では、毎回簡単なアンケートを実施した。アンケートは毎時間の最後に行ない、数行の感想の他、「難しさ」「課題の達成度」「楽しさ」を4段階で記入した。図4～図6に集計結果を示す。

11回の授業を前半(2学期の内容。1～5時限)と後半(3学期の内容。6～11時限)に分け、授業が進むにつれてどのように変化するかを符合検定で比較したところ、次の結果が得られた。

- 授業が進むにつれて難しさが増している
- 授業が進んでも課題の達成度に大きな変化はない

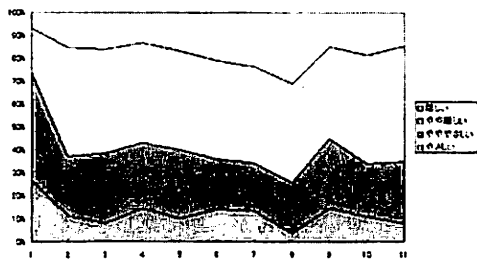


図 4: アンケート結果 (難しさ)

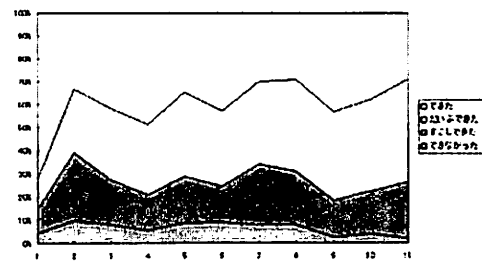


図 5: アンケート結果 (達成度)

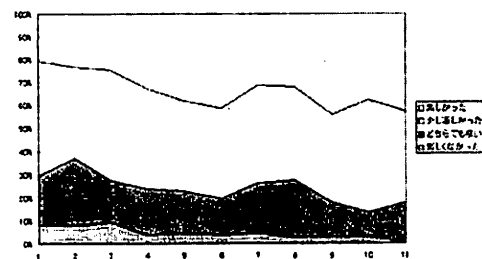


図 6: アンケート結果 (楽しさ)

● 授業が進むにつれて楽しさが増している
集計結果と検定結果から、次のことが明らかになった。

- 無理なく授業が成立した
生徒がその時間の課題を達成できなかったと感じた割合は、常に10%未満と低かった。
- 授業が進むにつれて楽しさが増した
11回の授業は、基本的な逐次的な実行を中心とした前半5回と、タイマーによる繰り返しやスレッド、ボタンによるメソッド定義やイベントなどの高度な概念を含む後半6回に分けられる。前半と後半を符号検定で比較した

ところ、後半のほうが、楽しさが増している生徒の割合が多いという結果になった。

- 達成したときの喜びが大きかった
難しさを感じて、それを乗り越えた喜びを感じている生徒が多かった [1]。アンケートの結果からも、前半に比べて後半を難しいと感じる生徒は増しているが、逆に楽しさを感じる生徒の比率が高まっていることがわかる。

4.2 授業終了後のアンケート

最終回の授業終了後に、理解度を見るためのアンケートを実施した。結果を表 2 に示す。

このアンケートでは、プログラミングに関する 8 個の概念を質問し、それぞれを理解している割合を見た。授業は実習を中心に進める形式であったため、これらのすべての概念を授業の中で明示的に教えたわけではない。

回答は 5 段階で行なったが、ここでは「1(そう思う)」「2(少しそう思う)」を「肯定」、「4(あまりそう思わない)」「5(そう思わない)」を「否定」として扱っている。

このアンケートの結果から、次のことが明らかになった。

- 各設問で肯定的な回答をした生徒は 60～80% であり、プログラミングの基本的な概念を理解した生徒が多いことがわかった。
- 否定的な回答の割合は全設問を通して 10% 未満と低く、プログラミングの概念を誤って理解している生徒は少ないことがわかった。

4.3 期末試験による評価

4.3.1 定期テスト(1)

5 時限までの内容を 2 学期の期末テストとして出題し、評価を行なった。テストは技術・家庭科の技術科の部分 (25 分間、50 点満点) として行なわれた。出題のねらいを表 3 に、テスト問題を図 11 に示す。

テストは、平均点が 32.3 点であり、満点 (50 点) が 4 人という結果であった。点数の分布を図 7 に、各問題の平均点を表 4 に示す。選択問題である問題 4 の平均点は約 9 点であり、記述問題である問

表 2: 授業後のアンケート結果

問題	設問	肯定 (%)
(1)	コンピュータは自分が書いたプログラムの通りに厳密に動作するようにできている。	60.6
(2)	プログラムの書き方はきっちり決まった規則があって、その規則を少しでもはずれていると動作しない。	83.2
(3)	プログラムは原則として書かれた順番どおり上から下、左から右の順番にきっちり実行されていく。	70.4
(4)	タイマーによる動作だけは (3) と異なり、タイマーに指示したタイミングで実行される。	62.5
(5)	「タートル」「図形」「タイマー」「ボタン」など、ものの種類ごとに見え方やできることが違っている。	79.4
(6)	さまざまなものに命令するときは、そのものを指定した後に「!」をつけて、それから命令を書く。	68.2
(7)	「あるく」「移動する」「塗る」などの命令の前には、「どれくらい動くか」「どこに移動するか」「何色に塗るか」などの指定をつけることができる。	81.7
(8)	繰り返し実行させる動作、タイマーに実行させる動作、ボタンが押されたときの動作などは「…」という形で囲んだもの(ブロック)で指定する。	53.2

題 1,2,3,5 の平均点はおおむね 6 点であった。

記述問題については、回答の内容を分析した結果、75～90%の生徒が出題の意図した内容を理解しているものと考えられる (理解率)。「正解ではないが理解している」と判定した場合の基準を示す。

- プログラムの概念は理解しているが、図形などの数学的な概念で誤答となった場合
- プログラムの概念は理解しているが、記号の抜けや書き間違いなどの細かなミスにより構文が不完全だった場合

4.3.2 定期テスト(2)

11 時限までの内容を 3 学期の期末テストとして出題し、評価を行なった。テストは技術・家庭科の技術科の部分 (25 分間、50 点満点) として行なわれた。出題のねらいを表 5 に、テスト問題を図 12 に示す。

テストは、平均点が 27.7 点であり、満点 (50 点) が 2 人という結果であった。点数の分布を図 8 に、

表 3: 2 学期末テスト 出題のねらい

問題	内容
1	逐次実行の理解 ・プログラムの実行結果を図示できる
2	プログラムの記述 ・課題の図形を描くプログラムを記述できる ・繰り返し実行が分かり、使いこなせる
3	文法の理解 ・言語規則が分かり、構文エラーを発見できる
4	オブジェクトの区別 ・オブジェクトの種類ごとに使える命令の違いが分かる
5	応用問題 ・与えられたプログラムを理解し、プログラムを追加して完成できる

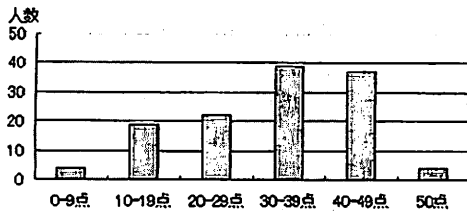


図 7: 2 学期末テストの点数分布

各問題の平均点を表 6 に示す。

記述問題については、回答の内容を分析した結果、問題 1,2,4,5 については、70%以上の生徒が出題の意図した内容を理解しているものと考えられる(理解率)。

4.4 生徒の作品と感想

課題として、授業の最後に、ボタン操作で線や図形を描くペイントソフトを作成した。

図 9 に、生徒作品の例を示す。このプログラムでは、画面左にボタンが配置されており、それぞれのボタンに次の種類のメソッドが定義されている。これらの機能を活用した結果、図のような豊富な表現力を持つペイントソフトを作成できた。

- ・ タートル操作ボタン。タートルを操作する(移

表 4: 2 学期末テスト 問題ごとの平均点、理解率

問題	内容	平均点	理解率 (%)
1	逐次実行の理解	5.6	89.6
2	繰り返しの概念	6.1	84.0
3	文法の理解	6.0	92.0
4	オブジェクトの区別	9.1	91.2
5	オブジェクトの操作	5.5	73.6

表 5: 3 学期末テスト 出題のねらい

問題	内容
1	オブジェクトの区別 ・オブジェクトの種類ごとに使える命令の違いが分かる
2	行なえる動作の判断 ・与えられたプログラムが可能な動作を判断できる ・プログラムを修正し拡張できる
3	応用問題 ・与えられたプログラムを理解し、プログラムを追加して完成できる ・タイマーオブジェクトの動作が分かる
4	プログラムの記述 ・課題の図形を描くプログラムを記述できる ・提示された目標を計画を立てて実行できる。
5	文法の理解 ・言語規則が分かり、構文エラーを発見できる ・オブジェクトにメソッドを定義して活用できる。

動、回転、ペンの上げ下げ)

- ・ 図形描画ボタン。繰り返し命令でタートルを移動し、図形オブジェクトを生成する
- ・ 色指定ボタン。ペンの色を変更する
- ・ 姿の指定。タートルの画像を変更する

担当教員の観察によると、生徒はボタンの機能に特に大きな興味を示し、何時間あっても夢中になって取り組む姿が観察された。興味を持った理由としては、Windows に付属する「ペイント」のようなソフトを作ることで、市販されているソフトウェア製品と自分の作ったプログラムが結び付いたことが考えられる。

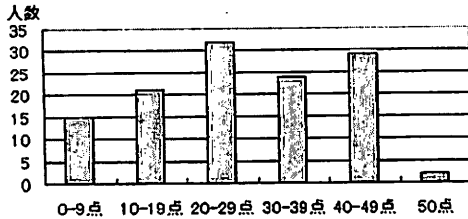


図 8: 3 学期末テストの点数分布

表 6: 3 学期末テスト 問題ごとの平均点、理解率

問題	内容	平均点	理解率 (%)
1	オブジェクトごとの違い	7.0	90.2
2	可能な動作の判断	6.6	92.7
3	文法を正しく書く	3.4	42.3
4	プログラムを設計する	4.9	74.8
5	メソッドの定義	5.8	71.5

表 7 に、「ペイント」を含む各種の生徒作品の中で使われているドリトルの概念を示す。ボタンやタイマーを始めとする高度な概念を、80~90%以上の生徒が活用していることが明らかになった。

5 考察

中学校の技術・家庭科において、プログラミング言語「ドリトル」を用いた情報教育を行なった。授業を通して得られた知見は次のものがある。

- タートルグラフィックスを用いて自分が作成し

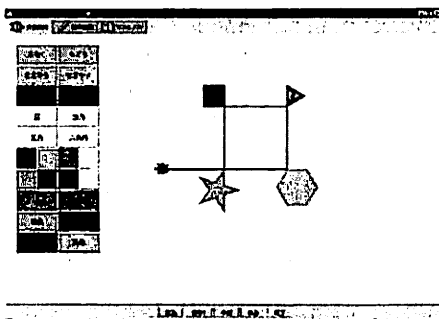


図 9: 生徒作品 (ペイントソフト)

表 7: 生徒作品の分析

機能	使用数	(%)
タートル	123	(100)
図形オブジェクト	119	(97)
繰り返し	118	(96)
ボタン	115	(93)
タイマー	104	(85)

たコードが動作する様子が即座に見られてフィードバックが得られることで、学習効果を高めることができた。

- プログラミング言語の構文として日本語による変数名、メソッド名 (命令語) を採用したことで、プログラミング言語に対する初期の敷居を下げる事ができた。
- 図形をオブジェクトとして位置付け、色を塗ったり位置を移動したりできるという形でのオブジェクト指向の利用は、生徒にとって抵抗なく受け入れられた。
- タイマー (定期的なイベント) を用いたアニメーションも前項と同様の流れのなかで問題なく学習できた。
- GUI 部品 (特にボタン) を学び、「ボタンを押すと何らかの動作が起きる」形のプログラムを作成することに多くの生徒が強い興味を示した。

特に最後の GUI 部品の利用により、生徒に現代のソフトウェアがどのようにして作られ動作しているのかを理解させるきっかけが作れたものと考えている。

6 まとめ

全体として、ドリトル言語を用いた情報教育では従来の教育用言語 (BASIC、LOGO) を用いた場合よりもスムーズに生徒の学習を進めることができ、またさほど多くない時間数にも関わらず、生徒のコンピュータの原理に対する理解を高めるという目標が達成できたと考えている。

本研究の実験授業に協力頂いた松阪市立鎌田中学校の教員と生徒の皆様へ感謝致します。

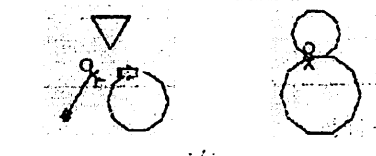
内容	説明	実行時
初期準備	初期準備を動かすようにします。	1回、初期
初期準備	初期準備を動かすようにします。	1回、初期
実行	1. 初期準備を実行します。	
	例: 実行「カメス」は白く「あぐ」実行。	
終了	自分自身を「複製」して新しいタイマーを立ち上げます。	
	例: 実行「タイマー」は白く。	

説明: 初期準備を実行して、複製を実行してみよう。
 実行: 初期準備を実行して、複製を実行してみよう。
 例: 「カメス」は白く「あぐ」実行。 (OOの実行する動作をみる)
 (例) 「あぐ」は白く「あぐ」実行。 (OOの実行する動作をみる)
 (例) 「あぐ」は白く「あぐ」実行。 (OOの実行する動作をみる)

説明: 自分自身を複製して動かすようにしよう
 「カメス」は白く「あぐ」実行。 (OOの実行する動作をみる)
 (例) 「あぐ」は白く「あぐ」実行。 (OOの実行する動作をみる)

説明: タイマーを実行してみよう。
 実行: 「あぐ」は白く「あぐ」実行。 (OOの実行する動作をみる)
 (例) 「あぐ」は白く「あぐ」実行。 (OOの実行する動作をみる)

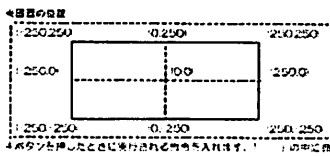
※サンプルファイルの紹介 (ドットに入っています)



ボタンを作ろう

1. 次の命令を実行してみよう。
 - カメス: タートル「あぐ」をつくる。 (1)
 - ボタン: ボタン「あぐ」をつくる。 (2)
 - ボタン: 200 (0) 実行。 150 50 実行。 (3)
 - ボタン: 実行「あぐ」は白く。 (4)

実行ボタンを押した時、画面上にある「あぐ」と表示されたボタンを押してみてください。
 (実行の結果)
 1. 「あぐ」という名前のあるタートル(スプライト)を1匹作り出す。
 2. ボタンを作ります。ボタンの中に入る名前「あぐ」の中に入ります。
 3. 「あぐ」は白く「あぐ」実行。
 4. ボタンの位置と大きさを決めます。
 5. 200 (0) 実行。 ボタンの実行を止めたい時、0実行。 テキストが1行目。
 6. 150 50 実行。 ボタンの大きさを決めます。幅が150。高さが50のサイズです。



- 実行ボタンを押した時、画面上にある「あぐ」と表示されたボタンを押してみてください。
- 実行: ボタンを作ろう。
 1. 「ボタン」という名前のあるタートル「あぐ」を作ろう。
 ボタンの位置は、(200, 0)。
 実行させる命令は、「あぐ」は白く「あぐ」実行。
 実行: 「あぐ」ボタンを作ろう。
 実行: ボタンを押して実行。画面上にボタンが表示される。ボタンを作ろう。

図 10: 授業で使用したテキストの例

参考文献

- [1] 井戸坂幸男, 紅林秀治. 「ドリトル」ではじめる情報教育 第3回 中学校 技術・家庭科での授業実践. NEW 教育とコンピュータ, Vol. 18, No. 3, pp. 84-87, 2002.
- [2] 兼宗進, 中谷多哉子, 御手洗理英, 福井真吾, 久野靖. オブジェクト指向言語「ドリトル」を利用した情報教育について. 情報処理学会 情報教育シンポジウム (SSS2001), pp. 275-282, 2001.
- [3] 兼宗進, 御手洗理英, 中谷多哉子, 福井真吾, 久野靖. 学校教育用オブジェクト指向言語「ドリトル」の設計と実装. 情報処理学会論文誌, Vol. 42, No. SIG11(PRO12), pp. 78-90, 2001.
- [4] 中谷多哉子, 兼宗進, 御手洗理英, 福井真吾, 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.

