

## テキスト型プログラミング言語へのスムーズな移行の検討

中村 佐里

自由学園高等科

sari.nakamra@pfr.jiyu.ac.jp

遠藤 敏喜

自由学園最高学部

end@pfr.jiyu.ac.jp

波多野 和彦

江戸川大学

khatano@edogawa-u.ac.jp

プログラミング学習では、学習者課題を工夫する以外にも、学習者に合わせ、開発環境や言語を選択することが重要である。本稿では、ビジュアルプログラミング言語から、テキストプログラミング言語へどのような移行が望ましいかを検討した。テキスト型プログラミング言語への「足場かけ」として、Excelの論理関数を用いた条件式を学習内容に取り入れ、実践した結果、学習者にとって身近なツールを用いた学習内容は「足場かけ」として有効である可能性が示された。

### 1. はじめに

新学習指導要領において、新科目「情報Ⅰ」で必修化されるプログラミングでは、対象の役割や働きを分析し、問題を切り分け、必要な手順などを論理的に組み立てること、並びに、試行錯誤を繰り返しながら不備な点を探して修正し、解決に至る道筋を探ることなどを学習することになる。

しかし、前述の対象の分析や仕組みの解明などの学習方略は、生徒にとって馴染みが薄く、「難しい」といった苦手意識を持つ者が多い。

このような苦手意識等を解消するためには、課題のレベルを明示し、生徒自身が選択できるように課題を用意することによって、本来の授業目標へ至ること示唆されているが<sup>(1)</sup>、それ以外にも、学習者の習熟度に合わせて、プログラミング学習の開発環境や言語などを検討することも重要である。

特に、プログラミング学習で使用する言語は、我々の日常生活で使う言葉とは異なり、独特の考え方や作法がある。学習者は文法的な誤りやタイプミスによるエラーが起きても、自分自身で間違いに気づくことができず、デバッグ作業に大きな負担を感じることが多い。そのため、学習の導入段階では、デバッグ作業の負荷軽減のため、間違いに気づきやすい「Scratch」のようなビジュアルプログラミング言語（以下、「ビジュアル言語」とす）を利用することは有効である<sup>(2)</sup>。しかし、プログラムの構造体や複雑な処理を学習するなど、学習者の学びを広げ、次の段階へ導くには、ビジュアル言語から、プログラムを文字で記述するテキスト型プログラミング言語（以下、「テキスト型言語」とす）へ移行していく必要がある。

そこで、学習者のレディネス等の調査を踏まえ、プログラミングの初学者がビジュアル型言語からテキスト型言語への移行するための学習内容を検討した。

### 2. ビジュアル言語からテキスト型言語への移行

#### 2.1 「足場かけ」の必要性

教育課程上2単位の学習時間数の中では、プログラミングを扱う時間数は限られており、すべての生徒がプログラミングを学習するのであれば、プログラミングの技術を習得するよりも、論理的な思考や創造力を育むことに注力しなければならない。ところが、生徒にパソコンの知識や技術に不安がある場合、プログラミングの技術の理解に追われ、授業目標や活動の意味を理解せずに終わる可能性がある。

そのため、生徒がつまづかないように、ビジュアル言語を用いて学習し、次の段階としてテキスト言語を学ぶなど、学習のステップを細かく設定していくことが必要である。

そして、学習者がプログラミングの有用性に気づき、学習内容に価値を見出すためには、その学習過程において、様々な「足場かけ(scaffolding)」を教員が整えていくことが重要である<sup>(3)</sup>。

#### 2.2 授業内容の検討

そこで、どのような「足場かけ」が必要であるかを検討するにあたり、生徒のプログラミングに関するスキルや意識を調査した。

教科情報の「社会と情報」履修者（高1女子：42名）にプログラミングの経験を質問したところ、88.1%が「学習経験がない」と回答した。そのため、学習上の負担の軽減と、テキスト型言語を習得する際の足場かけとして有用に機能されたことに鑑み<sup>(4)</sup>、本実践では、プログラミング学習の導入としてビジュアル言語（Scratch）を利用することを第1の「足場かけ」とした。

しかし、授業時の振り返りから、単純にオブジェクトを動かす授業回では「面白かった」「理解できた」などのコメントが多かったが、分岐処理や繰り返し処理、並列処理などを含む授業回では、「少し複雑であり理解できなかった」「上手くオブジェクトが動かない」などプログラム構文の難

しきを感じ、苦戦している様子が伺えた。

プログラミングの学習者が、初期段階で難しいといった印象を持つのは、プログラムの制御構造の理解が十分でないことが関係しており、プログラムの実行時の処理の流れを、プログラムを作成する段階で正しくイメージできるようにすることが必要である<sup>(4)</sup>。

そこで、プログラムの制御構造の理解を促すため、第2の「足場かけ」を検討した。生徒の54.8%が、表計算での「関数」を使うことができると回答していることから、生徒に馴染みのある表計算ソフトウェア(Excel)の論理関数を利用することとした。

論理関数である「IF関数」を用いて論理式を作成し、条件に応じて値を返すことを体験させる学習内容を検討した。まず、ビジュアル言語を用いて、プログラムの制御構造を説明し、実際にプログラムを作成した後、図1のように、Excelのワークシート上にテストの得点表を作成し、点数に応じて、「合格」「不合格」を表示させる関数を考えさせた。

Excelで関数を入力する場合、構文に間違い等があるとエラー表示されるため、自分で間違いを見つけやすく、ほとんどの生徒が表を作成し、論理式の確認をすることができた。余裕のあった生徒は別の条件を検討したり、論理式を入れ子にして、複数の条件を設定するなど、学習を深めることができた。

例えば、テストの点を判定する表を作成するとします。条件を60点以上なら合格、59点以下なら不合格とします。この場合、どのように条件を分岐させれば良いでしょうか？

A	B	C	D	E	
1	A組 数学テスト点数				
2	19	76	86	50	56
3	88	79	30	99	24
4	59	60	77	73	12
5					
6	A組 数学テスト判定				
7	不合格	合格	合格	不合格	不合格
8	合格	合格	不合格	合格	不合格
9	不合格	合格	合格	合格	不合格

実は、Excelには「IF関数」と呼ばれる関数があり、条件を設定し、その条件に従って、数や文字を返します。右の例ではセルA7からE9まで次のような式が入っています。

**=if(セル>59, "合格", "不合格")**

上の式の「セル」には、テストの点数が入っているセルを入力します。  
例えば、セルA7で判定する場合は、「19点」が入っている「A2」を入れます。

フィズバズゲームでは、3の倍数かどうかの判定式を作りましたが、Excelでも同じような考え方をして、条件式を作成することができるのです。

図1 Excelの論理関数を扱ったプリント

### 3. 学習者へのアンケート調査結果

プログラミング学習の授業後に、授業全般について尋ねたところ、「Excelを使ってプログラミングの考え方を学んだこと」については、78.6%が肯定的(大変良かった、良かった)に回答しており、生徒にとって、身近なアプリケーションを用いて、論理式を学習したことは、「足場かけ」として有効に働いたといえる。

授業時間の関係上、Excel VBAを使った実習には至らなかった。しかし、このような「足場かけ」が有効である可能性が示されたことで、Excelのセル参照機能を使って、定数変数の考え方を学んだ上で、同じプログラムをVBAで作成する、Excelの論理式を用いた学習にバリエーションを持たせるなど、プログラミング学習を広げる可能性が期待される。

### 4. 今後の課題

テキスト言語への移行がスムーズに行われることにより、生徒のプログラミング学習が広がっていくことは望ましいが、あくまでプログラミング技術は問題解決の道具立てであることを理解させる必要がある。本学の大学部では、特色ある教育課程を生かし、学生の立場から学内の学習環境の改善を検討し、専門ゼミでのアプリ開発を行っている<sup>(6)</sup>。このような学内での問題解決は、プログラミング技術と社会との関わりについて、生徒の「足場かけ」となるであろう。

様々な「足場かけ」を検討しながら、習熟度に応じたプログラミング学習のカリキュラムの開発を試みたい。

### 参考文献

- (1) 中村 佐里, 波多野 和彦, 三尾 忠男:「初学者の視点を踏まえたプログラミング授業の設計と実践 -「情報I」へのスムーズな移行に備えて-」, 日本情報科教育学会誌, Vol.11, No.1, pp.43-48 (2018).
- (2) 杉浦 学, 松澤 芳昭, 岡田 健, 大岩 元:「アルゴリズム構築能力育成の導入教育:実作業による概念理解に基づくアルゴリズム構築体験とその効果」, 情報処理学会論文誌, Vol.49, No.10, pp.3409-3427 (2008).
- (3) プロフィ, J.E.中谷素之(監訳):「やる気を引き出す教師 学習動機づけの心理学」, 金子書房 (2011).
- (4) 松澤 芳昭, 酒井 三四郎:「ビジュアル型言語とテキスト記述型言語の併用によるプログラミング入門教育の試みと成果」, 研究報告コンピュータと教育(CE), 2013-CE-119(2), pp.1-11 (2013)
- (5) 中村 佐里, 小池 万緑, 中田 樹, 中村 弥生, 遠藤 敏喜:「学生による学内ポータル開発の試みー調査に基づくアプリ機能の検討ー」, 日本教育工学会研究報告, JSET19(1), pp.213-216 (2019)