

プログラミング言語習得のための第二言語習得法の適用について About Application of Second Language Acquisition Methods to Programming Language Learning

梅澤 克之¹⁾中澤 真²⁾中野 美知子³⁾平澤 茂一³⁾

Katsuyuki Umezawa Makoto Nakazawa Michiko Nakano Shigeichi Hirasawa

1 はじめに

近年、「学習指導要領」が改定され英語教育とプログラミング教育の早期導入が強調された。英語教育に関しては、小学校外国語活動の教科化が含まれ、プログラミング教育に関しては、初等中等教育において本格的な情報教育が始まった。プログラミング学習における概念の習得に向けた具体的な方法としては「概念そのものの学習」と「事例学習（事例からの学習）」の2つが挙げられるが、学習の導入段階においてどちらから始めるべきかといった議論においては意見が分かれている。プログラミング学習の戦略に関する研究は多くなされているが、我々は、プログラミング言語と英語をそれぞれ母国語以外の言語である「第二言語」という共通項で捉え、いずれの言語の学習においても多くの研究がなされている第二言語習得 (SLA) のアプローチが適用できるのではないかと考えた。我々は、プログラミング言語と英語の学習時の生体情報を比較し、英文の読解問題とプログラミング言語の読解問題、英作文とプログラミングの問題のように同類のカテゴリの問題を解いている際の生体情報には有意差は観測されず、これにより、英語の学習戦略をプログラミングの学習に応用できる可能性を示唆した。本研究では、従来研究で定義されたプログラミング言語学習のレベル分類法を使って、著者が担当した科目を分類する。この分類により、著者の担当科目は SLA-aBLe プロジェクトで定義されたすべての理解度（最小限の理解～高度流暢まで）を網羅することを示す。

2 従来研究

2.1 SLA-aBLe プロジェクト

プログラミングスキル自己評価マトリックス [1] は第二言語習得のための CFER [2] を基にしてプログラミング言語習得のために応用したものである。このように第二言語習得法をプログラミング言語習得に応用するという研究は多い。プログラミング言語学習のための SLA アプローチの有効性を調べることを目的とした混合学習プロジェクト Second Language Acquisition in a Blended Learning (SLA-aBLe) [3] が知られている。このプロジェクトでは、プログラミング言語と自然言語の共通点に着目し、教え方の観点で第二言語習得の成功例をプログラミング言語教育に応用した。結論として第二言語習得のための教育方法を応用した方がプログラミングの成績は高くなったと述べている。この時の評価項目としては、内発的動機づけ指標（興味や緊張感等）と自己評価型の作業負荷指標（努力やフラストレーション等）をアンケートにより評価している。

この従来研究 [3] で示された 2015 年当時に適用されていた学習法と SLA-aBLe 学習法の比較を表 1 に示す。

- 1) 湘南工科大学 Shonan Institute of Technology.
- 2) 会津大学短期大学部 Junior College of Aizu.
- 3) 早稲田大学 Waseda University.

複数の画像を用いながら簡単に説明するといった基礎の部分や、学生をより主体的に学ばせるためにグループワークや、プレゼンの場を提供するなど細かい工夫点が表によってわかる。

3 担当科目の分析

本章では、コロナ禍（2020 年度～2022 年度）の著者のプログラミング関連の主な担当科目に関して、従来研究を参照することで分析を行う。具体的には、プログラミングスキル自己評価マトリックス [1] を使って、著者の担当科目のレベルを判定する。そして 2.1 節で示した SLA-aBLe プロジェクトで示されたレベル分け表に著者担当科目を当てはめる。

3.1 著者のプログラミング関連の担当科目

コロナ禍（2020 年度～2022 年度）の著者のプログラミング関連の主な担当科目は下記のとおりである。

- 基礎プログラミング実習（1 年次）
- 情報倫理とセキュリティ（2 年次）
- PBL プログラミング実習 A（2 年次）
- コンパイラ（3 年次）
- ソフトウェア工学（3 年次）

なお、2020 年度は Google Meet を使ってリアルタイムオンライン授業を行った。2021 年度は家族がコロナに罹患した場合など自宅待機が必要な学生はリアルタイムオンライン授業、体調に問題のない学生は教室での対面授業というようにハイブリッド授業を実施した。2022 年度はほぼ対面授業に戻った。

3.2 著者担当科目の難易度判定

CEFR [2] は言語の発達段階を A1, A2, B1, B2, C1, C2 の 6 段階の指標として提案されたもので現在世界基準になっている。CFER [2] を基にしてプログラミング言語習得のために応用したプログラミングスキル自己評価マトリックス [1] と、著者の担当科目を比較すると、「ソフトウェア工学」に関しては B1 あるいは B2 のレベルを目指している科目であるが、それ以外の科目は、ほぼ A1 あるいは A2 の範疇である。4 年目の卒業研究まで進めば C1 あるいは C2 のレベルまで習得を目指す研究室もあるが、3 年生までの著者の担当科目は CFER のレベル分けで言うところの「基礎段階の言語使用者」ということになる。

3.3 著者担当科目の SLA-aBLe へのマッピング

2.1 節で示した SLA-aBLe プロジェクトは、エンジニア向けコンピューティング入門コース向けの授業で実施したとのことであり、前節の著者の担当科目の学生とほぼ同程度のプログラミングレベルの学生が対象であると言える。そこで、SLA-aBLe プロジェクトで示された表 1 の教育戦略を基にして、この表に、著者の担当科目の授業内容をマッピングする。マッピングした結果を表 2 に示す。表中の下線を引いた文言が、筆者担当科目の授

業内容に関連するであろうと考えた根拠となる。表 2 に示したように、著者の 1 年次向けから 3 年次向けまでの担当科目を割り当てると、SLA-aBLe 戦略のすべての理解度（最小限の理解～高度流暢まで）を網羅することができた。当然学科内で複数の教員でプログラミング関連の授業を担当しているの、一人の教員だけで SLA-aBLe 戦略のすべてを網羅する必要は無い。しかしそれぞれ個別の担当科目間の連携などを考慮すると、一人の教員ですべてのレベルを網羅することは無意味では無いと言える。

4 まとめと今後の課題

本研究では、第二言語学習法をプログラミング言語学習に応用する従来研究として、プログラミングスキル自己評価マトリックスおよび SLA-aBLe プロジェクトに着目した。それらで定義されたプログラミング言語学習のレベル分類法を使って、著者がコロナ禍で担当した科目を分類した。この分類により、著者の担当科目で SLA-aBLe プロジェクトで定義されたすべての理解度（最小限の理解～高度流暢まで）を網羅することを示した。今後は他の教員の科目を含むカリキュラムツリー全体のマップ化を推進したい。また、湘南工科大学は

2023 年度より情報学部が新設されたため、新学部のカリキュラムツリー全体でのマッピングを進めたい。その際には、学科全体として CEFR のもっと上級レベルまで引き上げるか等々の検討も行っていきたい。

謝辞

本研究は JSPS 科研費 JP22H01055, JP21K18535, JP20K03082 の助成を受けたものです。また、本研究の一部は、経営情報学会「ICT と教育」研究部会の助成を受けたものです。本研究成果の一部は早稲田大学理工総研プロジェクト研究「次世代 e-learning に関する研究」の一環として行われたものです。

参考文献

- [1] Raphael 'kena' Poss. How good are you at programming? a ce-fr-like approach to measure programming proficiency. 2014 (modified 2018). Accessed: 29 April 2023.
- [2] Council of Europe. Common european framework of reference for languages: Learning, teaching, assessment. *Language Policy Unit, Strasbourg*, Vol. First Edition, , 2001.
- [3] Lulu Sun, Christina Frederick, Li Ding, and Rebecca Rohmeyer. The application of second language acquisition to programming language study. *American Society for Engineering Education, ASEE Annual Conference*, pp. #18842 1-12, 2017.

表 1 2015 年当時の混合学習と SLA-aBLe 開発の比較

	プリプロダクション (最小限の理解)	早い制作 (限定理解)	スピーチエマージェンス (理解力の向上)	中級流暢 (非常によく理解できる)	高度流暢
現在 (2015 年当時) のブレンド学習	写真とビジュアルが少ない。一部のトピックは十分に説明されていない。スクリーンキャストに十分な自己テストの質問がない。	多肢選択問題があるが、簡単なプログラムはない。Facebook は利用するが、グループディスカッションはない。	学生は、さまざまなエンジニアリングの問題を解決することによって、プログラミング言語での読み書きを開始する。	学生に与えるもっと彼らが学んだことを統合するための挑戦的な問題。	彼らの理解に挑戦し、知識を拡大するためのオープンエンドのエンジニアリングプロジェクト。
SLA-aBLe の教育戦略	写真とビジュアルを使用する。ゆっくりと話しシンプルで短い単語を使用して、SLA とプログラミング言語を関連付ける。プレッシャーを加えることなくより多くの自己テストの質問をすることで学習を強化する。	複数の選択肢の質問に加えて、簡単なプログラムを作成するよう学生に求めることで、学習を強化する。ディスカッションボードを使用して、グループディスカッションを促進する。	階層化された質問を強調し、学生に「考え、ペアにし、共有する」ことで新しい概念を処理するように指示する。	異なる概念を比較対照することを強調する。学生が問題解決のプロセスを説明できるようにする。	プロジェクトのプレゼンテーションの機会が学生に提供され、理解が深まる。

表 2 著者の担当科目の SLA-aBLe への当てはめ

	プリプロダクション (最小限の理解)	早い制作 (限定理解)	スピーチエマージェンス (理解力の向上)	中級流暢 (非常によく理解できる)	高度流暢
著者の担当科目の実施内容	【情報倫理とセキュリティ】身近な事件・事故などについて調査を実施。IPA が提供しているセキュリティの動画を授業に取り入れ、常日頃から学生が使用している LINE やその他のツールを対象とする。	【基礎プログラミング実習】 Java 言語を用いてプログラムの基本を学ぶ。プログラミングが苦手な学生は paiza ラーニングというサイトの「Java 入門編」を利用する。 【PBL プログラミング実習 A】 グループワークをメインとする科目である。教育版マイクラフトでプログラミングとチームワーク作業を体験する。	【ソフトウェア工学】 6 名 1 チームで自動販売機 (物理的な機械ではなく PC 上で動くシミュレータ) の開発を進める。 【PBL プログラミング実習 A】 4 人 (±1 人) のグループで同一ワールド内で構造物をプログラミングで作成することを行う。	【ソフトウェア工学】 開発のフェーズごとにレビュー (教員・教員補助者への説明) を受けて初めて次のフェーズに進む。 【PBL プログラミング実習 A】 教育版マイクラフトでプログラミングを体験しながらチームワーク作業を体験する。 【コンパイラ】 その際に様々なアルゴリズムがある中で再帰的下向き構文解析法を使った関数電卓を作成する	【ソフトウェア工学】 最終的に開発物の発表会および仕様書通りに作られているかのチェックを受ける。 【PBL プログラミング実習 A】 最終週にはグループごとの発表会を行い、構造物のコンセプトやプログラミング上の工夫点をアピールする。