

プログラミング能力検定協会推薦論文

プログラミング教育における定着度評価の検討と実践

飯坂正樹¹ 五十嵐智生² 兼宗 進³ 中村めぐみ⁴ 内田 卓⁵

¹プログラミング能力検定協会 ²(株) インダストリー・ワン, 情報処理学会SIG部会 ³大阪電気通信大学, 情報処理学会初等中等教育委員会 ⁴つくば市教育委員会総合教育研究所 ⁵つくば市立吾妻小学校

日本におけるIT人材の不足と国際競争力の低下はIT教育の遅れに起因している。この課題を受けて学校におけるプログラミング・情報教育の必修化がなされたが、プログラミング能力の客観的評価・可視化が困難であることから教育現場では適切な指導を行うことが難しい。そこで本稿では子どもたちのプログラミング能力の定着度実地検証を行った結果から、プログラミング教育の量や質の差によるプログラミング能力定着度の違いについて述べるとともに、今後学校や大学、プログラミングスクール等で効果的な指導を行う上での汎用的かつ有用な情報を提供する。

※本稿はプログラミング能力検定協会 推薦論文です。

※本稿の著作権は著者に帰属します。

1. プログラミング教育の現状と課題

1.1 背景

文部科学省は、2017年2月14日「次期学習指導要領等の改訂案」[1]を公表し、小学校段階における英語教育、プログラミング教育の義務化を発表した。山本ほか(2015)の整理では、小学生のプログラミング教育義務化の動きは、2013年6月14日「日本再興戦略 JAPAN is BACK」としてアベノミクス「3本の矢」の1つとして世界最高水準のIT社会の実現の目標を達成するために、2010年代には1人1台の情報端末の推進を実現し、義務教育段階からのプログラミング教育の必要性が記載された。また、内閣に設置されている高度情報通信ネットワーク社会推進戦略本部(IT総合戦略本部)では2014年6月24日「世界最先端IT国家創造宣言」が改定され、プログラミング教育の必要性がうたわれたと施策を整理している。直近では、首相官邸で開催された2018年5月17日未来投資会議(第16回)で文部科学大臣から、「Society 5.0に向けた人材育成の推進」が提出され、初等中等教育におけるプログラミング教育の義務化、小中高を通じた学習指導要領改訂によるプログラミング教育・統計教育の充実などが提示された[2]。

新型コロナウイルスの蔓延による小中学校のロックダウンなどを経て初等中等教育の学びに関しては、その後も文部科学省で検討が行われ、2021年1月26日の中央教育審議会では、「令和の日本型学校教育」の構築を目指して～全ての子供たちの可能性を引き出す、個別最適な学びと、協働的な学びの実現～（答申）」として取りまとめられた[3]。2017年の次期学習指導要領等の改訂案の段階では小学校プログラミング教育の義務化の取り組みなど、実際の学校現場で対応できるのかが課題とされていたが、現場での取り組みや中央教育審議会での検討が進み、具体的な学びの取り組みについて踏み込んだ整理がなされた。

2023年3月8日の中央教育審議会では、「次期教育振興基本計画について（答申）」[4]がまとめられた。プログラミング教育の点で成果として、第1に初等中等教育に関しては、国際的に高い学力水準の維持、GIGAスクール構想、第2に高等教育に関しては、教学マネジメントや質保証システムの確立、連携・統合のための体制の整備が挙げられている。小学校、中学校に関してはプログラミング教育が整備され、高校に関しては情報I、IIが教科化された[5]。

「次期教育振興基本計画について（答申）」では、次期計画のコンセプトとして2040年以降の社会を見据えた持続可能な社会の創り手の育成の取り組みとして、社会課題の解決を経済成長と結び付けてイノベーションにつなげる取り組みや、一人ひとりの生産性向上等による活力ある社会の実現に向けて人への投資が必要とされている。教育環境に関しては、教育デジタルトランスフォーメーション（DX）の推進として、第1にDXに至る3段階（電子化→最適化→新たな価値（DX））において、第3段階を見据えた、第1段階から第2段階への移行の着実な推進を計ること、第2にGIGAスクール構想、情報活用能力の育成、校務DXを通じた働き方改革、教師のICT活用指導力の向上等、DX人材の育成等の推進を行うこと、第3に教育データの標準化、基盤的ツールの開発・活用、教育データの分析・利活用の推進を行うこと、第4にデジタルの活用と併せてリアル（対面）活動も不可欠、学習場面等に応じた最適な組合せを行うことが挙げられている。具体的な政策としてこれらを実現するためのSTEM教育の実現やデータサイエンスなどの利活用について、今後、学習指導要領等の改訂等を通じて具体的な取り組みが進むものと予想される。

1.2 課題

プログラミング教育の観点で見ると、新型コロナウイルスによる各種学校のロックダウンなどの影響で、オンラインによる教育の実施やITを利用した教育の実施など、教育を推進するために文部科学省を中心とした政策立案側も、実際に学校で教育を担う教育者の側も否応なく新しい教育環境に対応せざるを得なかった側面があると言えよう。中央教育審議会では、「次期教育振興基本計画について（答申）」でGIGAスクール構想の成功を成果として報告がされるのは、これら実務として実際に取り組んだ積み重ねが評価されたものと考えられるだろう。

一方で、プログラミング教育の観点で、これまでに議論されてきた「令和の日本型学校教育」に関して検討したい。2021年1月に中央教育審議会が発表した「令和の日本型学校教育」のこれまでの実践とICTとの最適な組合せを実現する取り組みとして、第1にICTや先端技術の効果的な活用により、新学習指導要領の着実な実施、個別に最適な学びや支援、可視化が難しかった学びの知見の共有等が可能、第2にGIGAスクール構想が実現されることを最大限活かし、教師が対面指導と遠隔・オンライン教育とを使いこなす（ハイブリッド化）ことで、学びの質を向上、第3に教師による対面指導や

児童生徒同士による学び合い、多様な体験活動の重要性がますます高まる中で、ICTを活用しながら協働的な学びを実現し、多様な他者とともに問題発見・解決に挑む資質・能力を育成することが挙げられている。

新時代の学びを支える環境整備については、基本的な考え方として、すべての子供たちの可能性を引き出す個別最適な学びと協働的な学びを実現し、教育の質の向上を図るとともに、新たな感染症や災害の発生等の緊急時であってもすべての子供たちの学びを保障するため、「GIGAスクール構想」の実現を前提とした新しい時代の学びを支える学校教育の環境整備を図るとし、「1人1台端末」や遠隔・オンライン教育に適合した教室環境や教師のICT環境の整備や「1人1台端末」の活用等による児童生徒の特性・学習定着度等に応じたきめ細かな指導の充実や、「新しい生活様式」を踏まえた身体的距離の確保に向け、少人数によるきめ細かな指導体制や小学校高学年からの教科担任制の在り方等の検討を進め、新時代の学びを支える指導体制や必要な施設・設備を計画的に整備するとしている。

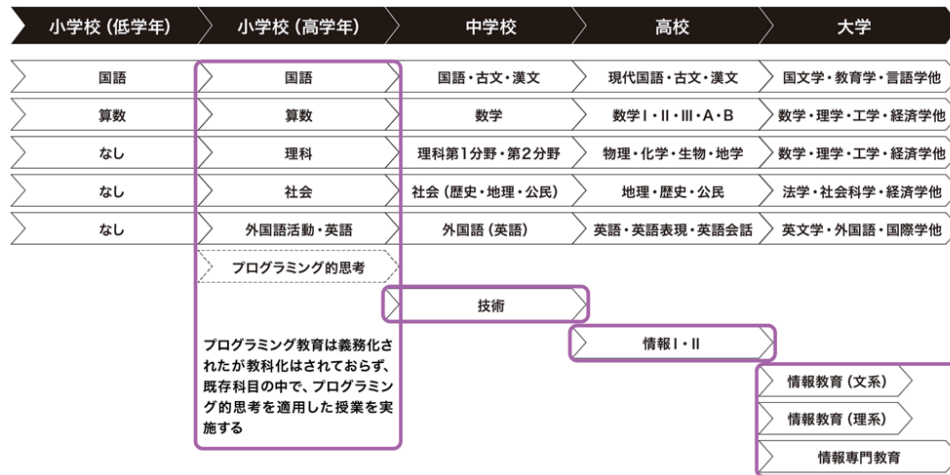
Society 5.0時代における教師および教員組織の在り方については、基本的な考え方として、第1にAIやロボティクス、ビッグデータ、IoTといった技術が発展したSociety 5.0時代の到来に対応し、教師の情報活用能力、データリテラシーの向上がますます重要であること、第2に教師や学校は、変化を前向きに受け止め、求められる知識・技能を意識し、継続的に新しい知識・技能を学び続けていくことが必要であり、教職大学院が新たな教育課題や最新の教育改革の動向に対応できる実践力を育成する役割を担うことも大いに期待する、第3に多様な知識・経験を持つ人材との連携を強化し、そういった人材を取り込むことで、社会のニーズに対応しつつ、高い教育力を持つ組織となることが必要と明記され、教師のICT活用指導力の向上方策が具体的に示されている。

国のこれからの計画を読むと、GIGAスクールで実現した「1人1台端末」を利活用する点に関して積極的に推進することが明記されているが、STEM教育やITやデータを利用した教育をきちんと教育課程のどこに位置づけて教育していくかの議論は少ない。小学校では「プログラミング的試行」で教科化が見送られており、国語、算数、理科、社会、家庭科など初等教育の中の既存の教科の中でプログラミングを組み入れた授業を行う取り組みがなされている。文部科学省では各学校での取り組みをさまざまに紹介して現場と情報の共有をしているが、プログラミングに詳しい、あるいは熱心な教員のいる学校は積極的にプログラミング教育が取り組まれ、そうではない学校ではそれなりの形になってしまっている。中等教育である中学校では、技術の授業としてプログラミングが行われているが、小学校で「プログラミング的試行」として行われていたプログラミング教育が中学校では技術として実務的な教科の中で取り組まれることにギャップもあるだろう。教科化に関しては高校で情報が教科化されたが、中学校、小学校の教育との連動性は議論されているように見えない。

課題としては、小学校から大学までの情報システムにかかわる一貫した教育カリキュラムが未整備である点が挙げられるのではと考える。図1のように、具体的には高校で情報、中学で技術、小学校でプログラミング的思考による全教科での展開が、教育制度として整備されつつあるが、小学校、中学校、高校と一連の教育システムとして見ると、教科の連続性の問題は「次期教育振興基本計画について（答申）」でも解消されているように見えない。

12. 教育カリキュラムの課題

日本で予定されている情報教育体系 (2020年)



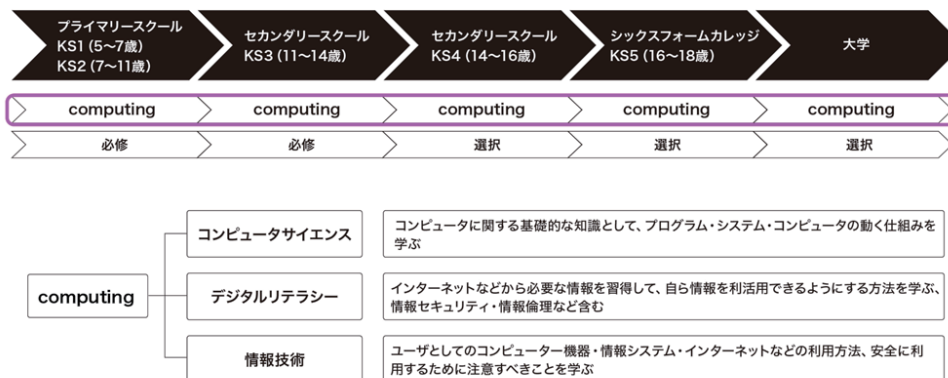
©2018一般社団法人情報処理学会認定技術者CITPコミュニティ プログラミング教育専門部会

図1 プログラミング教育の担当科目 (日本)

一方で“The Royal Society : Computing in Schools, Shut Down or Restart?”によれば、1980年代からプログラミング教育の義務化が始まった英国ではコンピューティング (Computing) という教科でプログラミング教育が教科化されており一貫性は明らかである (図2)。日本では、すべてを解消することはすぐには難しいが、プログラミング教育の効果を何らかの形で効果を見える化するなど、新たな打ち手が必要ではないかと考える。

13. 教育カリキュラムの課題

英国の情報教育体系 (2013年～)



©2018一般社団法人情報処理学会認定技術者CITPコミュニティ プログラミング教育専門部会

図2 プログラミング教育の担当科目 (英国) [6],[7]

2. プログラミング能力調査

2.1 プログラミング能力検定の概要

前述のプログラミング教育における非連続性などの課題がある中で、プログラミング教育の効果を最大化するために、「プログラミング能力検定」[8]を活用し、現状の教育機関における子どもたちのプログラミング能力、つまりプログラミング教育・学習の成果の可視化を試みた。

プログラミング能力検定はプログラミングの基礎知識を客観的かつ詳細に図ることができるテストである。民間／公教育を問わず、教育の現場では「プログラミング教育」という一律の評価が難しい領域において、学習者のモチベーション維持や目標設定に課題を持っている。プログラミング能力検定はこういった背景を踏まえ、学習内容によらずプログラミングの基礎知識を同じ基準で評価するための尺度として設計されており、今後さまざまな教育機関で採用されていくことが期待されている。

プログラミング能力検定はCFRP (Common Framework of Reference for Programming Skills) [9]という独自のプログラミングスキル標準に基づいて構成されている。このCFRPを用いることで、プログラミング言語によらず、一律の基準でプログラミングの学習・指導・評価を行うことができる。

実際にこのCFRPは多くの学校、プログラミングスクールでも指標として用いられており、教員だけでなく保護者等、大人が子どもたちのプログラミング能力を確認する上でも有効とされている。CFRPでは表1のように6つのレベルでプログラミングの習熟度を定義しており、さらに60以上の概念にプログラミングの基礎知識を細分化している。

表1 CFRP

Common Framework of Reference for Programming Skills (CFRP)

段階	レベル	レベル毎に「何ができるか」を示した習熟度一覧
実用性の高いプログラムを作成可能	6	プログラムで複数の処理を同時に行うことができることを理解している。オブジェクト指向の基本的な概念を理解し、他の人が利用しやすいプログラムを作ることができる。
	5	例外処理を理解し、より安定したプログラムを作ることができる。データ型の比較、多次元配列、ソートの理解により、複雑なデータ処理を行うことができる。
複雑な処理のプログラムを作成可能	4	定数の概念を理解し、使用することができる。繰り返しや分岐を様々な条件でコントロールする方法や配列要素の追加・削除・検索などの操作を理解し、複雑な処理のプログラムを自由に作ることができる。
	3	変数の型について理解し、多様なデータをプログラムで扱うことができる。乱数の概念を理解し、使用することができる。関数の基本的な概念を理解し、効率的に可読性の高いプログラムを作ることができる。
簡単な処理のプログラムを作成可能	2	文字列の操作をすることができる。論理演算により複雑な条件を作成することができる。配列の基本的な概念を理解し、使用することができる。特定の条件下で処理を繰り返すことができ、簡単な処理のプログラムであれば自由に作ることができる。
	1	簡単な演算、データの表示の方法を理解している。変数の基本概念を理解し、使用することができる。データの大小や一致／不一致の条件で処理を分岐させるプログラムを作ることができる。

プログラミング能力検定はビジュアル言語、JavaScript、Pythonの3種類の言語から選択して受験が可能であり、受験レベルはビジュアル言語が1から4（4が最高レベル）、JavaScript・Pythonのテキスト言語は1から6（6が最高レベル）で構成される。今回の実証では「ビジュアルレベル1」の結果にフォーカスして分析を行っており、その中で測られるプログラミング概念は主に表2のCFRPレベル1に含まれている。

表2 CFRPレベル1で定義するプログラミング概念詳細

CFRPレベル1 ^					
区分	概念	評価指標	ビジュアルブロック	JavaScript	Python
制御	順次実行	プログラムは上から順番に実行されることを理解している	スタートをクリックしたとき		
制御	条件分岐 (if 文)	if文を使用して条件によって動作を変えることができる	もし〇〇なら	if () { }	if :
制御	条件分岐 (if ~else文)	if~else文を使用して条件に合っている場合と合っていない場合で動作を分けることができる	もし〇〇なら~でなければ	if () { } else { }	if : else:
制御	条件分岐 (if ~else if 文)	if~else if文を使用して複数の条件によって動作を変えることができる	"もし〇〇なら、もし〇〇なら~でなければ"	if () { } else if () { }	if : elif :
制御	データ出力	値を出力することができる	〇〇という	console.log	print
数的処理	正負の数	正負の数を扱うことができる	[-10] 歩動かす		
数的処理	数字の大小	数字の大小で動作が変わることを理解している	大きさを [20] にする、大きさを [100] にする		
演算子	四則演算	四則演算を行うことができる	〇 + 〇 〇 - 〇 〇 × 〇 〇 / 〇	+, -, *, /	+, -, *, /
演算子	等号	値や文字が「等しいとき」という条件を作成することができる	〇=〇	==	==
演算子	不等号	値が「~より大きい (小さい) とき」という条件を作成することができる	"〇<〇 〇>〇"	<, >	<, >
変数	変数の宣言	変数を宣言することができる	変数を作る	let	
変数	変数代入	変数に値を代入することができる	変数を〇にする	=	=

またプログラミング能力検定は受験後には図3のような詳細な分析結果を伴う「成績表」を発行している。

XXXX年X月検定

成績表

氏名: 検定 太郎 | 学年: 小学3年生

会場名: プロ検塾 池袋校 | 登録番号: 999999

あなたの受験結果

受験レベル	合格	正答数	16 / 20 問
ビジュアル言語レベル1	合格	正答率	80.0%
		受験者全体の正答率	63.4%

● プログラミング概念別正答率

概念分類	説明	正答率
制御	条件などによってプログラムをコントロールするための概念をまとめたものです。繰り返し、条件分岐、順次実行などの概念を総称しています。	あなた 84.6% 全体 76.0%
数的処理	算数や数学で学ぶ概念をまとめたものです。角座、負の数、等号、不等号の概念を総称しています。	あなた 100.0% 全体 79.7%

解答結果の分析

2つの概念分類のうち、あなたは「数的処理」がよく理解できています。複雑なプログラムを作る上で必要な知識ですので、プログラミングの基本をさらに学習することで活かしていきましょう。「制御」は学習アドバイスを元にもう少し練習していきましょう。

● プログラム概念別の結果分析詳細

概念分類	概念 (対応するブロック)	概念の説明	結果	学習アドバイス
制御	順次実行 ▶をクリックしたとき	プログラムは上から順番に動くことを理解している	△	自分で作ったプログラムのブロックの順番を変えて動きを確認してみましょう。
	wait文 待まつ	プログラムを指定した時間止めることができる	○	
	フレーム更新 ずっこ	プログラムを何度も繰り返すことができる	○	
	if文 (条件分岐) もし~なら	条件によって動きを変えることができる	△	「もし~なら」のブロックに「座席の数の範囲」や「マス目がボタンに貼られた」などの条件を入れて動きを確認してみましょう。
	初期化 0に初期化する、大きさを100にする	プログラムを開始した時にキャラクターの位置や大きさを元に戻すことができる	○	
	ifelse文 (条件2分岐) もし~なら、でもなければ	条件に合っている場合と合っていない場合で動きを分けることができる	△	「もし~なら、でなければ」のブロックを使ったプログラムを作って動きを確認してみましょう。「もし~なら」のブロックの違いを認識してみましょう。
数的処理	並列処理 ▶をクリックしたとき	2つの異なるプログラムを同時に実行することができる	△	1つのキャラクターに「▶」をクリックしたとき、2つ以上あったプログラムを作成して、動きを確認してみましょう。
	正負の数 10歩動かす、大きさを10ずつ減らす	キャラクターを「前後」に移動させることができる	△	「10歩動かす」の中の数字にプラスの数字を入れた場合とマイナスの数字を入れた場合の動きの違いを確認してみましょう。
	数字の大小 10歩動かす、大きさを10ずつ減らす	キャラクターを前に移動させるスピードを指定することができる	○	
	角座 (絶対) 0°に向ける	キャラクターを決めた方向に向けることができる	○	
	角座 (相対) 15度回す	キャラクターをグルグルと回転させることができる	○	

図3 プログラミング能力検定 成績表

この成績表には詳細なプログラミング概念ごとの評価と学習アドバイスが記載されており、学習者の復習はもちろん、指導者の個別最適な指導の補助、あるいはクラス・学年・学校・自治体といった単位での傾向分析にも使用することができる。

プログラミング能力検定は2020年12月に運用を開始し、小学校でのプログラミング必修化に加え高校での情報I必修化も受けて、民間のプログラミングスクール等および学校からの急拡大するプログラミング教育における「評価」ニーズに応える形で、2022年9月からは毎月実施している。

受験会場は主に学校や民間のプログラミングスクールであり、対象の受験者は小学生から高校生・高専生である（今後は専門学校生、大学生、社会人にも展開予定）。

今回はこのプログラミング能力検定の特性を活かし、実際の受験結果からプログラミングの基礎知識の可視化と分析を行った。

3. プログラミング能力の定着度差の原因考察

3.1 年齢による差

表3は2022年2月から2023年1月に小学校にて実施したプログラミング能力検定ビジュアル版レベル1の正答率を学年別に示したものである。学年が上がるにつれて正答率が高くなっており、読解力や計算能力、論理的思考等の基礎学力の差が大きく出ていると思われる。

表3 小学校の学年別正答率

学年	正答率	受験者数
小学1年生	31.8%	135
小学2年生	40.1%	151
小学3年生	43.4%	678
小学4年生	54.8%	3,548
小学5年生	58.6%	3,983
小学6年生	64.0%	4,805

また表4のように小学校、中学校、高等学校の差を比較するとやはり年齢差が表れている。中学校の正答率が高いのは学年全体ではなくクラブ活動などのプログラミングが得意な生徒が中心に受験をしたためである。

表4 学校区分ごとの正答率

	正答率	受験者数
小学校	58.4%	12,976
中学校	72.1%	823
高等学校	70.3%	3,703

3.2 学校とプログラミングスクールの差

表5は2022年2月から2023年1月の間における、学校と民間のプログラミングスクール間のビジュアル言語レベル1の正答率の差を示している（プログラミングスクールにおける高校生のビジュアル言語レベル1の受験は受験者数が少ないため掲載していない）。

表5 学校とプログラミングスクールの正答率の差

	学校	プログラミング スクール
小学生	58.4%	77.5%
高校生	70.3%	-

このデータによるとプログラミングスクールで受験した小学生の正答率は学校で受験した高校生の正答率よりも高い。この結果から、プログラミングの基礎知識は前述の年齢による差もありながら、学習状況にも大きな影響を受けている可能性がある。

まずは「学習時間」の違いが挙げられる。学校では小学校でプログラミング教育が必修化されたとはいえ、教科化がされておらず、学校現場では時数の確保は明確に定められていない。よってプログラミングにどれくらいの時間を割くかは学校あるいは教員に委ねられていることが多い。また高等学校では2022年度より情報Iが必修化されたが、学校現場では前例がない中で膨大な量の教科書の内容を進める必要があり、プログラミングに十分な時間を割けていない実態もある。一方で民間のプログラミングスクールに通う生徒は少なくとも毎週1時間程度はプログラミングの学習を行う時間が確保されている。

次に「学習内容」について、学校では純粋に「プログラミング」に特化した授業を行うことが難しい実態がある。小学校では教科学習を深めるためのプログラミング的思考が大切とされ、各学年各教科の年間計画にてプログラミング学習が効果的な単元を洗い出し、実際にプログラミング学習を行う場面の学習指導案等を作成している。また総合的な学習の時間等を利用してプログラミングを学習している学校もあるが、現時点では決して多いとは言えない状況である。

こういった状況を鑑みると、むしろ学校現場では限られた時間や背景の中で児童のプログラミング的思考を育成し、高い正答率を実現しているとも言える。

3.3 学校間の差

表6は2022年2月のつくば市立吾妻小学校と全国平均との正答率の差である（受験レベルはビジュアル言語版レベル1）。小学3年生～小学5年生の3学年の比較においていずれもつくば市立吾妻小学校は全国平均を上回った。またその差は学年が上がるにつれて大きくなっており、小学5年生は表4の高校生の平均正答率をも上回った。

表6 つくば市立吾妻小学校と全国平均との正答率の差

	つくば市立吾妻小学校	全国平均
小学3年生	56.3%	52.9%
小学4年生	64.7%	59.2%
小学5年生	73.6%	63.7%

また表7はつくば市立吾妻小学校の児童が1年後に同様のレベルの受験を行った際の正答率の変化である。いずれの児童も1年で正答率を伸ばしており、特に3年生から4年生への1年間の伸びは13.7%と著しく、表3の全体の小学3年生と小学4年生との差である11.4%と比べてもその成長幅の大きさがうかがえる。また2023年3月時点の小学6年生の正答率75.6%は前述の民間プログラミングスクールにおける正答率に迫る。

表7 つくば市立吾妻小学校における正答率の経年変化

学年	2022年2月	2023年3月
小学3年生→小学4年生	56.3%	70.0%
小学4年生→小学5年生	64.7%	70.5%
小学5年生→小学6年生	73.6%	75.6%

つくば市ではつくば市総合教育研究所（教育委員会内）監修の「つくば市プログラミング学習の手引き」というプログラミング学習のモデルカリキュラムがあり、1年生から6年生のプログラミング教育の具体的実践例がまとめられている。特に1年生から「ScratchJr」ではなく「Scratch」を使用する実践が推奨されているのは全国でも珍しい。つくば市では、低学年であるから使用することが難しいというフィルタをかけずに、多様なビジュアル言語に触れさせる方針となっている。

今回実証に参加した吾妻小学校では普段からタブレットを使用した活動が推奨されており、たとえば「理科の授業のまとめをScratchを使ってプログラミングによって行う」児童もいる。従来は「紙」や「プレゼンテーションソフト」を用いて行っていた活動をプログラミングを用いて行っているのである。ただしこれは教員の指示で行っているのではなく、「紙」「プレゼンテーションソフト」といった従来の選択肢を残しながら、子どもたちに選択肢として「プログラミング」という手段

を開放し、あくまでも子どもたち自身が自ら選択している。ツールとしてのプログラミングを子どもたちに対して提供するだけで、従来の方法よりも分かりやすく魅力的な表現を生み出す児童が自然と出てきているのである。

また同校では休み時間やほかのさまざまな授業の局面で積極的に端末やプログラミングを使っている。子どもたちは休み時間に外に遊びに行くもよし、プログラミングに勤しむもよしと、プログラミングが身近に取り組める活動として常に用意されている。ある授業では「外で鬼ごっこをするか」「マイクラフトで鬼ごっこをするか」という選択をクラスの児童全員に問い、結果マイクラフトを使ったオンラインでの鬼ごっこを実施し、係の児童がルールやワールドを設定することで大いに盛り上がったとのことである。マイクラフトは本来ゲームであるが、昨今はマイクラフトを使ったプログラミングの学習環境も整備され、プログラミングスクールを中心にさまざまな教育機関で活用されている。現在のほとんどの子どもたちがその存在を知っており、かつ端末活用を含めたICTへの興味喚起を図るには十分なコンテンツである。

3.4 プログラミング概念ごとの差

表8は2022年2月から2023年1月におけるプログラミング能力検定ビジュアル版レベル1の小学生、中学生、高校生それぞれの各概念における正答率平均を示している。

表8 学校区分ごとの概念別正答率平均

概念	小学生	中学生	高校生
順次実行	80.4%	87.9%	87.3%
wait文	71.8%	82.6%	80.8%
フレーム更新	78.9%	86.4%	85.9%
if文（条件分岐）	85.5%	90.9%	92.2%
正・負の数	82.0%	91.8%	92.9%
数字の大小	75.9%	89.5%	89.6%
初期化	58.2%	59.9%	68.5%
if else文（条件二分岐）	76.8%	85.1%	87.4%
並列処理	51.3%	52.9%	63.4%
角度[絶対]	68.0%	82.1%	82.3%
角度[相対]	78.6%	86.9%	86.4%

このデータからは、年齢を問わず共通の傾向があることが分かる。たとえば、「wait文」「初期化」「並列処理」等の概念は年齢を問わず相対的に正答率が低い傾向がある。これらはプログラミングにおいて特に特徴的な概念であり、直感的にイメージしづらいものと思われる。一方でプログラミ

ングの代表的な概念である「条件分岐」についてはシンプルな「if文（条件分岐）」に対して「if else文（条件二分岐）」の正答率が相対的に低く、これも処理の流れをよりイメージしづらいものと思われる。

また「正・負の数」「数字の大小」「角度」等の数学的な概念は当然年齢差もあるものの、算数や数学で習っていない学年の子どもがプログラミングを通してその概念を理解することもあるため、プログラミングを学ぶことの1つのメリットと捉えている。

4. 教育現場への適用と今後

4.1 プログラミング学習と定着度の関係

これまでの分析から、プログラミング能力の向上には年齢による発達以上に時間と環境が重要であろうことが分かった。これは必ずしも民間のプログラミングスクールだけでなく、学校においても一定のプログラミング能力の向上が見られたことから明らかであろう。特に小学校3年生から4年生の間の成長は著しく、3年生までいかにプログラミングに触れさせるかが子どものプログラミング能力を底上げする1つのヒントになるかもしれない。

またプログラミングの概念には日常生活や他教科の学びからはイメージがしづらいものもあるため、実際にプログラミングを作って動かす機会を増やしていく必要があると考える。

以上のような考察はCFRPおよび「プログラミング能力検定」がプログラミング概念を詳細に分析し、可視化できるが故のものであり、従来にない評価方法を確立できたとと言えるだろう。

4.2 教育現場への適用案

次に今回の結果を踏まえて、今後の教育現場への適用について考えたい。

まず冒頭に挙げた課題である「プログラミング教育の非連続性」については、今回の方法で都度子どもたちのプログラミング能力を可視化することで、その子どもの特性を正しく理解するための1つのデータとなることが考えられる。その上で、別の指導者が、その子どもの指導にあたる場合に、特性を理解した上で学習支援にかかわることができるのではないだろうか。実際に高等学校の教員が新入生にプログラミング能力検定を受験させることで既学習差の可視化を行い、指導計画に反映するケースもある。

またプログラミング概念ごとの評価によって指導内容へのフィードバックを行い、効率的にプログラミング教育を行うことができるだろう。さらにその概念を身に付ける上で相性の良い教科や単元を検討することで教科外の時数の消費を抑えることもできると考える。

さらにプログラミング能力の向上が他教科で求められる能力に寄与すること、また、課題発見能力、課題解決能力の向上にも繋がることを証明することで、より教育現場におけるプログラミング教育の位置づけが重要となり、推進しやすくなると考える。

4.3 今後の展開

今回は限られた期間・範囲のデータを用いて分析を行ったが、今後さらにプログラミング能力検定の受験者数が拡大する中でより解像度の高い分析が可能となり、特に初等中等教育の領域におけるプログラミング教育と評価を検討するにあたり有益なデータが蓄積されていく。初等中等教育で学ぶ児童、生徒のプログラミング能力に関する定量的なデータを継続的に記録し分析していくことは、現在、政府で進めている「デジタル田園都市国家構想」などとも趣旨を同じくするものである。

デジタル田園都市国家構想では、すでに社会に出ている社会人や大学などの高等教育機関のリスキングに注目が集まっているが、義務教育領域で児童や生徒の段階でプログラミングに触れることは、将来のデジタル人材の教育、発掘に不可欠であり有意性のある取り組みだと考える。プログラミング能力検定を通して得られるデータは冒頭で述べた公教育における一連の端末整備・プログラミング教育・情報教育必修化の成果を測る一助ともなり得るだろう。

我々は今後もプログラミング能力検定の活用を通して研究を続けていくが、本活動は公益性もありかつ、国の推進するデジタル戦略との連動性も高い取り組みだと考えている。一方で一民間企業の取り組みや、学会などの活動に多いとは言えない初等中等教育の先生方が中心に現場で臨床的に取り組まれている現状は、不十分な部分もあり、現在進められている高大連携と同様に、大学と初等中等教育現場の先生方とも連携の上で展開していきたい。

たとえばプログラミング教育を主たる研究テーマとされている研究者の方と連携し、より効果的な教育方法や評価指標を作っていくことや、プログラミング能力検定のデータと普通の学習教科のテストスコアデータを用いて、どの教科を重点的に学ぶことでプログラミング能力のどの分野に効果があるのか、またプログラミング教育をより効果的に行う上で、現在の教科をどのように利活用してプログラミング的思考を鍛えれば良いのかを研究することなども考えられる。

多様な学校現場において子どもたちの実態を踏まえた分析を行い、さまざまな角度からプログラミング教育の成果をまとめることで、実効性の高い事例として全国の学校へ横展開することも可能である。

公教育における全国規模のプログラミング教育は始まったばかりであり、まさに今後その成果が求められる時期に入っていく。我々の活動にご興味があり、ご助力いただける教育機関、企業の方がいらっしゃればぜひお声がけいただきたい。

プログラミング能力はこれからの世の中を生き抜くための力であることは間違いない。1人1台の端末が学校にも普及し、プログラミング教育が必修化された今こそ、子どもたちに効果的な指導や適切な環境を提供し、彼らの将来の選択肢の広がりにつなげていきたい。

参考文献

- 1) 文部科学省：幼稚園，小学校，中学校，高等学校および特別支援学校の学習指導要領等の改善および必要な方策等について（答申）（中教審第197号），2016年12月21日，中央教育審議会，http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo0/toushin/1380731.htm（参照 2023-5-1）
- 2) 首相官邸：日本経済再生本部 未来投資会議（第16回），2018年5月17日，文部科学大臣提出資料，<https://www.kantei.go.jp/jp/singi/keizaisaisei/miraitoshikaigi/dai16/siryou6.pdf>（参照2023-5-1）
- 3) 文部科学省：中央教育審議会資料，「令和の日本型学校教育」の構築を目指して～全ての子供た

ちの可能性を引き出す、個別最適な学びと、協働的な学びの実現～（答申）（中教審第228号）

【2021年4月22日更

新】， https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/079/sonota/1412985_00002.htm

（参照 2023-5-1）

4) 文部科学省：次期教育振興基本計画について（答申）（中教審第241

号）， https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo0/toushin/1412985_00005.htm

（参照 2023-5-1）

5) 文部科学省：GIGAスクール構想の実現につい

て， https://www.mext.go.jp/a_menu/other/index_00001.htm（参照 2023-5-1）

6) The Royal Society：Computing in Schools, Shut Down or Restart?,
<https://royalsociety.org/topics-policy/projects/computing-in-schools/report/>,
<https://royalsociety.org/~media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf>（参照 2023-5-1）

7) The Royal Society: Computing Education, After the Reboot: Computing Education in UK Schools, <https://royalsociety.org/topics-policy/projects/computing-education/>, <https://royalsociety.org/~media/policy/projects/computing-education/computing-education-report.pdf>（参照 2023-5-1）

8) プログラミング能力検定協会：プログラミング能力検定， <https://programming-sc.com>（参照 2023-5-1）

9) プログラミング能力検定協会：CFRP（Common Framework of Reference for Programming Skills）， <https://programming-sc.com/cfrp/>（参照 2023-5-1）



飯坂正樹（正会員）

iisaka@programming-ri.com

プログラミング能力検定協会代表， 本会ジュニア会員サポーター， 国立大学法人長岡技術科学大学客員准教授， 国内外でプログラミング教育の普及に従事し， 大学ではプログラミングの講義も行う。



五十嵐智生（正会員）

hinonantaro@g.ecc.u-tokyo.ac.jp

（株）インダストリー・ワン事業統括ディレクター， 本会SIG部会， 東京大学大学院学際情報学府博士課程でプログラミング教育の研究をしている。

兼宗 進（正会員）

大阪電気通信大学副学長， 本会初等中等教育委員会委員， 本会論文誌教育とコンピュータ編集委員長， 検定教科書や学習指導要領を執筆。

中村めぐみ（非会員）

つくば市教育委員会総合教育研究所 情報担当指導主事。市内のプログラミング教育、ICT教育、EdTechを推進し、その取り組みを全国へしている。

内田 卓（非会員）

つくば市立吾妻小学校教諭。校内のICT端末の活用およびプログラミングを取り入れた授業実践を多数推進している。

採録決定：2023年6月26日

編集担当：長坂健治（キンドリルジャパン）