

小学校のプログラミング教育についての文献研究

佐藤 修

1. はじめに

1.1 本稿の目的

文部科学省は、初等教育（小学校）へのプログラミング教育導入を進めている。その基礎理論としては計算論的思考がある。しかしこれがプログラミング教育導入の理論的根拠であるという記述は明示されていない。本稿では、小学校段階でのプログラミング教育導入の基礎理論としての計算論的思考と、これに基づく小学校段階でのプログラミング教育導入の欧米の実践・状況についての研究を概観することで、導入の現状と課題を検討する。

1.2 文部科学省の方針

高度情報社会では、インターネットに代表される情報通信技術（Information & Communication Technology: ICT）の活用が必須である。文部科学省は現代の高度情報社会における社会共通の教育基盤として、学習指導要領で「学習の基盤となる資質・能力」及び「情報活用能力」の育成（文部科学省，2017）を目指している。これにより、生徒の知識及び技能、思考力、判断力、表現力、学びに向かう力、人間性等の教育効果を実現しようとしている。このために、「学校教育分野，社会教育分野における情報化の推進のため」教育の情報化を推進している（文部科学省，2020）。このうち，小学校については平成29年（2017年）に小学校学習指導要領を告示し，令和2年（2020年）4月から実施している。これを実現するための補助資料として，令和2年6月には「教育の情報化に関する手引—追補版」を公開した。教育情報化の必要性については，変化の激しい将来の社会で情報や情報技術を主体的に選択し活用する力を付けさせることを求めている。このために「教育の情報化」を推進して教育の質の向上を目指すことを求めている。これには以下の3側面があり，これらを実現することが重要であるとしている。

- ①情報教育：子供たちの情報活用能力の育成
- ②教科指導における ICT 活用：ICT を効果的に活用した分かりやすく深まる授業の実現等
- ③校務の情報化：教職員が ICT を活用した情報共有によりきめ細やかな指導を行うことや，校務の負担軽減等

そしてこれらの教育の情報化の実現を支える基盤として、

- ・教師の ICT 活用指導力等の向上
- ・学校の ICT 環境の整備
- ・教育情報セキュリティの確保

が必要であると述べている（文部科学省，2020）。

上記のプログラミング教育推進によって日本でも小学校でのプログラミング教育が始まった。しかし 2021 年 4 月時点でその実施率は 28.1%（くもん出版，2021）と、まだ導入の途上である。

2. 概念的背景

2.1 計算論的思考

上記のような教育の情報化の必要性の認識は、文部科学省に限らず、情報処理又は計算機科学（Computer Sciences: CS）分野の世界中の研究者に共通である。海外各国とも同様の認識で、教育の情報化の必要性を認識している（Balanskat & Engelhardt, 2015; Heintz et al., 2017）。CS 分野の研究者はそのための方策として、計算論的思考（Computational Thinking: CT）を理解させ、国民のデジタル能力（digital competence）を育成・強化することが必要と考え、これが各国の教育情報化の基礎理論、特に初等中等教育にプログラミング教育を導入する根拠になっている。後述するように、欧州の先進国ではプログラミング教育が初等教育の必須項目になっている（Fagerlund et al., 2020）。

上記のキーワード CT は、Papert（1980）が Mindstorms で初めて紹介した。彼はリオデジャネイロのサンバスクールの見聞から着想を得て、学校がなくても計算機を活用して学習する社会のイメージを論じている。Microsoft の Wing（2006）は、Papert の CT の重要性を教育界に広く紹介した。それは問題解決・システムのデザインの考え方である。そして「読み書きそろばん」のように、全ての子供がその分析能力に加えるべきであると論じている。Wing（2006）は CT を「情報処理エージェントにより効果的に実行できるように記述

表 1 狭義の CT の内容

・計算論的概念（concepts）→主要概念の理解 順次・ループ・並列・イベント・条件・処理・データ等
・計算論的实践（practices）→過程・手続の理解 増分・反復・テスト・デバック・再利用・再結合・抽象化・モジュール化等
・計算論的見方（perspectives）→主体別の視点 再接続・問合せ・影響・評価・利用・表現等

Brennan & Resnick（2012）を参考に筆者が整理した。

される，問題と解の記述に係る思考過程」と定義して広く捉えている。

CTには狭義の理解と広義の理解がある。狭義には，計算論的概念（computational concepts），計算論的実践（computational practices），そして計算論的見方（computational perspectives）を含んでいる（Brennan & Resnick, 2012）（表1）。

2.2 広義の計算論的思考

Wing（2006）は，CTを計算機で使われる概念・実践・見方に狭く限定的に理解するのではなく，より広義に理解すべきであると述べている（表2）。狭義のCT（表1）は，広義のCT（表2）のうちの言語のスキル部分（2列目）に該当する。狭義のCTは重要な考え方である。しかし，それ以外の列の項目はそれ以上に教育上の価値が高い。広義の計算論的思考の学習がより重要である（Abbott, 2021）。初等中等教育へのプログラミング教育導入は，このための手段であると各国も教育指導者も認識している。Fagerlund et al.（2020）はCT概念とプログラミング実践から生徒が学べる概念・アイデア・技術の基礎理論として，CEPs（CT's core educational principles）を唱えている。

前記のCEPは以下の項目を含む：抽象化，アルゴリズム，自動化，協働，調整と並行処理，創造性，データ，効率性，反復，論理，モデル化と設計，パターンと一般化，問題分解，テストとデバック（Fagerlund et al., 2020）。これは狭義のCTよりも広いが，表2（Nouri et al., 2020）のCTよりも狭い。前記のように，CTが何を含むかは論者により異なる（Boom et al., 2022）。

文部科学省も，小学校段階でプログラミング教育を導入する目的として，表2と同様の成果を期待している（文部科学省，2016）。しかし上記資料は，小学校段階でのプログラミング教育の基礎理論としてのCTを明示的に記述していない。この状況は各国で同様である（Fagerlund et al., 2020）。但し表2で示すように，CTの教育目標は広範で，論者によりそ

表2 広義のCT

ロジカルシンキングスキル	言語のスキル（狭義のCT）	創造的問題解決のスキルと態度	他者と協働するスキルと態度
シンボルや抽象で考えるスキル	プログラミング文法	問題解決スキル	協働的問題解決スキル
正確性	コードの圧縮	創造性・ファンタジー	教育的コミュニケーションスキル
失敗したりテストしたりする勇氣	用語	再結合	
忍耐力・継続性			

Nouri et al.（2020）の表2から作成

小学校のプログラミング教育についての文献研究

の内容が一部異なる場合がある。保護者や小学校教師の CT の認識についても、かなり大きな幅がある（くもん出版, 2021; Garvin et al., 2019）。

本稿では、小学校段階でのプログラミング教育導入の基礎理論としての CT と、これに基づく小学校段階でのプログラミング教育導入の実践・状況についての研究を概観することで、導入の現状と課題を検討する。

2.3 デジタルリテラシー

上記の CT（計算論的思考）は抽象的で曖昧なので、国内ではあまり使われない用語である。国内ではほぼ同じ意味でデジタルリテラシー（Digital Literacy）或いはその類似語が使われることが多い。これには多数の類似語があり、論者により様々な用語が使われる。更にデジタルリテラシー及びその類似語の定義も論者により多様である（Ilomäki et al., 2011）。

3. 欧米の CS 教育推進

前記のように、欧米先進国は早くから積極的に初等教育に CS 教育とプログラミング教育を導入してきた。2015 年には European Schoolnet がイスラエルを含む欧州 21 カ国における CS 教育導入の調査をしている（Balanskat & Engelhardt, 2015）。このうち 17 カ国が公式に CS 教育をカリキュラムに取り入れた。但し、何をどのようにどこまで導入したかは国によって異なる（Heintz et al., 2017）。

エストニアでは 1990 年代から小学校でプログラミング教育を始めている。Sweden では、初等中等教育の STEM（Science, Technology, Engineering and Mathematics）担当科目でのプログラミング教育導入を、2017 年に決定、2018 年から実施している。

英国では 2014 年に他国に先駆けてプログラミング教育導入を決定した（UK Government Digital Service, 2014）。それまでは少数の教師が任意に CS 教育を導入する程度であったが、2014 年から導入を義務化した。しかし政府は十分な投資をしなかったために導入の実が挙がらなかった。そこで 2018 年 11 月以降、政府は大幅な投資（84 million pounds）を実施して、コンピュータ教育国家センター（National Centre for Computing Education, NCCE）を開設し、非営利団体の支援も得て、CS 教師の育成に真剣に取り組んだ（Fowler & Vegas, 2021; Wikipedia, 2022）。

3.1 教育標準の策定と標準化

国際連合教育科学文化機関（United Nations Educational, Scientific and Cultural Organization; UNESCO）は 1970 年代中旬に国際教育標準分類（International Standard Classification of Education: ISCED）を開発した。現在は ISCED 2011 が最新版で、これが各国の教

育システムを比較検討する際の基準になっている (UNESCO, 2012)。この基準は教育レベルを未就学児のレベル 0 から博士課程のレベル 8 迄に分類している。これらのレベルは、初等 (0~2)、中等 (3, 4)、高等 (5~8) 教育に大別される。教育分野としてのプログラミングは科学 (4) の計算 (48) に分類されている。

UNESCO は教師向けの ICT コンピテンシーの標準 (ICT Competency Framework for Teachers: ICT-CFT) を制定して、2010 年に公開した。これは教師の ICT コンピテンシーを包括的に開発するのを助け、教育計画に ICT を統合するコンピテンシー政策と標準を提供する。この改訂は現在も進んでおり、第 3 版が公開されている (UNESCO, 2018)。

3.2 デジタル・コンピテンシー

EU (European Union) では EC (European Commission) において、デジタル教育標準 “DigComp (Digital Competence)” を開発し、デジタル教育基準の標準化を推進している (European Commission, 2019)。EU はデジタル・コンピテンシーを「仕事・レジャー・コミュニケーションに情報社会の技術を、自信を持ってかつ批判的に利用する能力」と定義している。EU は以下の 8 つを重要なコンピテンシーとして示している (Ilomäki et al., 2011)。デジタル・コンピテンシーはこのうちのひとつ (4) として位置付けられている。

1. 自国語でのコミュニケーション
2. 外国語でのコミュニケーション
3. 数学的及び基礎的科学技術のコンピテンシー
4. デジタル・コンピテンシー
5. 学習することを学ぶコンピテンシー
6. 社会的・市民的コンピテンシー
7. 率先的・起業的感覚
8. 文化の認識と表現

DigComp は 21 世紀に全てのレベルの生徒・学生だけでなく全ての市民が必要とするデジタル・コンピテンシーを定義している。「欧州の学校のためのデジタル教育ビジョン」は「欧州の学校教育において全ての生徒は、学習・労働・社会参加のために、デジタル技術を、自信を持って、批判的に、責任を持って、創造的に使用するためのデジタル・コンピテンシーを育成しなければならない」として、DigComp をベースとするデジタル・コンピテンシー枠組み (Digital Competence Framework: DCF) を設定した (Schola Europaea, 2020; European Commission, 2022)。

DCF は以下の 5 つの領域 (Dimension) から構成される。

1. 情報とデータリテラシー
2. コミュニケーションと協働

3. デジタルコンテンツの開発
4. 安全
5. 問題解決

しかし具体的に何をどこまで含むのかは論者・国や状況によって異なる。社会や技術の進歩に伴い、内容が変化する (Ilomäki et al., 2011)。

Schola Europaea (2020) は、上記の各領域について、未就学児から初等教育 (小学校)・中等教育 (中学校) 迄の詳細なルーブリックを示している。DigComp プロジェクトは 2011 年に始まり 2013 年に最初の版が公開された。その後、2016 年には第 2 版が DigComp 2.0 として公開された (Pérez-Escoda & González-Fernández-Villavicencio, 2016)。現在は更に改版されて DigComp 2.2 になった (Vuorikari, et al., 2022)。プログラミングはこの中で「3. デジタルコンテンツの開発」の 3.4 である。

3.3 民間支援組織による標準提案

上記の公式組織の他に、民間レベルで CT 教育を支援する様々な組織がある。このうち International Society for Technology in Education (ISTE) は、ICT で教育革新を加速しようとする非営利団体である。ISTE は生徒・教師・教育指導者等 5 種類の利用者 (生徒, 教師, 管理者, 技術コーチ, コンピュータ科学教育者) を想定して、教育技術標準を策定して公開した (ISTE, 2007)。夫々の利用者は幾つかの役割を期待されており、夫々の役割で参照できる標準を提案している。結局、5 種類の利用者併せて 28 の標準を提案している。表 3 では例として、生徒に期待される 7 つの役割を示している。

Code.org は CS を教育に導入し、CS 教育を増やすことを目標にする非営利団体である。Code.org には以下の目標がある (Wikipedia の Code.org 頁から引用)

- ・アメリカ合衆国の全ての K-12 教育機関 (特に都市部, 農村部地域) に計算機科学授業を導入する
- ・公共学校教室でのオンラインカリキュラムの成功例をデモンストレーションする

表 3 生徒の 7 つの役割

生徒の 7 つの役割
1. 積極的な学習者 (Empowered learner)
2. デジタル市民 (Digital citizen)
3. 知識創造者 (Knowledge constructor)
4. 革新的設計者 (Innovative designer)
5. 計算論的思考者 (Computational thinker)
6. 創造的コミュニケーション者 (Creative communicator)
7. グローバル協働者 (Global collaborator)

- ・ 計算機科学を数学や科学における中核カリキュラムにするために全 50 州の政策を変える
- ・ 計算機科学教育を世界中に広め成長させるために技術コミュニティの力を集め活用する
- ・ 計算機科学分野において有色人種の女性や学生の存在感を増加させる

Code.org は 2025 年迄の長期目標として以下の目標（数値目標は省略）を示している (Code.org, 2020)。

1. 公立高等学校で CS 教育を普及させる
2. Code.org のコースを教える学校を増やす
3. 米国外の Code.org 利用者を増やす
4. Code.org を利用する URG や女性生徒の比率を増やす

(URG: Underrepresented Racial/Ethnic Groups, 少数派のエスニックグループ)

CSTA (Computer Science Teachers Association) は未就学児から高等学校まで (K-12) の CS 教師の国際団体である。2004 年に結成され、以来、初等中等教育・高等教育や産業界での CS 教育を推進している。ACM (Association for Computing Machinery) と協働して、初等中等教育における CS 教育の実態調査 (Wilson et al., 2010) 等も行っている。Code.org の連携団体でもある。

Bebras は、IT を用いて未就学児から初等中等教育段階迄の生徒の CT を向上させる国際的活動団体である。2004 年にリトアニアの Valentina Dagiene (Vilnius University) が始めた活動が国際的に広まり、現在は 60 カ国で 290 万人の生徒を指導している (Bebras, 2022)。日本でも情報オリンピック日本委員会ジュニア部会が組織され、ビーバーチャレンジという名前で、Bebras Challenge という国際情報科学コンテストへ小学 3 年生以上の生徒を派遣している (情報オリンピック日本委員会ジュニア部会, 2022)。Bebras Challenge は、CT 知識を測定する尺度として最も良く知られている (Boom et al., 2022)。

その他、子供にプログラミングを教える民間団体がコーダー道場 (Coder Dojo), code camp 等、いろいろある。

3.4 CS 教育枠組の統合

前記の ACM, Code.org, CSTA, 更に Cyber Innovation Center や National Math and Science Initiative は連携して、未就学児から初等中等教育迄の CS 教育の枠組み標準化を推進し、K12 Computer Science Framework という標準枠組みを公開した (K12 Computer Science Framework, 2022)。この枠組みを日本にも導入しようとする運動も既に始まっている (K12 Computer Science Framework Japan, 2022)。

4. 欧州におけるプログラミング教育例

本節では、プログラミング教育に絞って欧米各国の事例や関連研究を検討する。前記のように、学校教育全体で見ると、CTがプログラミング教育の理論的根拠、K12 Computer Science Frameworkに代表されるCS教育がドライバになっている (Kafai & Lui, 2020)。そしてこれに基づいたCS教育が推進されている。これに関する研究は、後述するように国際的に多数発表されている。プログラミングのような知的な活動を子供に教えるのは難しいので、プログラミング教育も初等教育の総じて中級及び上級の生徒（8歳から12歳程度）及び中等教育（13歳以降17歳迄）への教育が中心である。生徒の年齢が上がると担当教師も科目を限定的に担当するようになることで、プログラミング教育のより専門的な知識を獲得する機会が増える。また生徒の年齢が上がって知的に成熟してくると、より多様な教材を利用できるようになる (Szabo et al., 2019)。

英国は毎週1時間、専らCSを教える科目を設定して、未就学児である5歳から、小学校入学直後の7歳までの児童であるKey Stage 1からプログラミング教育を始めている。この年代では教育ロボットBee Botを使い、試行錯誤的にプログラムを設定させて、指示した走行をさせるように工夫させる等の教育を行う (STEM learning, 2022)。8歳から11歳迄のKey stage 2ではScratchを使ってプログラミング学習をさせる。

他方、FinlandやSwedenでは、CSを固有の科目としては設定せずに、多様なSTEM科目で選択的にCS教育を取り入れている。

初等教育におけるプログラミング教育では、中等教育以上におけるCS教育とは違う固有の課題があり、CS教育の導入よりも難しい。以下では初等教育におけるプログラミング教育に絞って、海外の事例研究を検討する。

4.1 プログラミング教育の代表的なツール

プログラミング教育で最もよく使われるのは前記のBee BotやScratchである。しかしSzabo et al. (2019)の文献調査によると、Scratchは中等教育以降で使われる例が多く、初等教育では、Scratchから改良されたブロック言語であるSnap!や、テキスト型プログラミング言語であるLOGOの利用報告頻度が高い。Fagerlund et al. (2020)はScratchによるCT教育効果を文献調査により調べている。以上の他によく使われるのは、Code.orgやMindStormである。Greifenstein et al. (2021)の調査によると、例えばドイツの小学校では、Scratchを使うのが65.5%、机上プログラミングが31.5%、19.5%がCode.orgであった。これらは無償でインターネットからダウンロードして使うことができる。しかしPCやタブレット端末の他、教育用教材ロボット (Lego NXT-G) 等、有償でしか入手できない教材もある。

机上プログラミングよりも実機（PC）を使うほうが生徒の関心を引くためにも教育効果でも有効なので、Scratch や Code.org を使うほうが望ましい。しかし、後述するような制約のため、ドイツでも机上プログラミングに頼る事例が多い。

4.2 Sweden におけるプログラミング教育導入

スウェーデンの教育は6歳から16歳までの子供・若者に対しては義務で無償である。2017年に小学校のカリキュラムを改訂し、デジタル・コンピテンシー教育とプログラミング教育を導入し、2018年から施行した（Heintz et al., 2017）。Heintz et al. (2017) はスウェーデンにおけるCS教育導入の歴史を紹介している。

新しいカリキュラムでは、全ての教師に大学で16時間のOnline Programming入門コースを履修させ、STEM科目担当教師には他に、最低7.5単位のプログラミング入門を履修させている。しかしこの制度導入前に就任したSTEM担当教師は受講していないので、担当教師の55%は公式の受講経験がない（2018/2019年時点、Vinnervik, 2022）。スウェーデンでは初等教育におけるSTEM科目でプログラミング教育が義務となったので、政府（National Agency for Education: NAE; Skolverket）は、教室内の情報処理環境の整備や前記のような教師の育成を推進している。しかし設備環境整備上の課題や小学校固有の制約要因があり、実際には様々な課題がある。

教師の課題としては、第一に初等教育（小学校低学年）では、教師はクラス担任制で、一人で全ての科目を教えなければならない。このためプログラミング教育のための十分な予習・準備をする時間も余力もない。第二に、前記のようにほぼ半分の教師は学生時代にCS/CT教育やプログラミング教育（教育法）を正式に勉強していない。このため教育法が分からない・授業でも自信がないという課題がある。要求された教育項目を単に並べているだけという意見もある。第三に、既存のカリキュラムで学習時間が埋まっており、新たにプログラミング教育を入れる時間枠がない。第四に、小学生は関心が移り（飽き）易く、長続きしない（Vinnervik, 2022）。

Vinnervik (2022) は、現場教師の会話を録音し、Finger & Houguet (2009) の課題分類枠組みを援用して質的分析し、プログラミング教育の課題を研究した。Finger & Houguet (2009) は教師が直面する課題を内的課題（intrinsic challenges）と外的課題（extrinsic challenges）に大別した。前者は教師が個人的に直面する心理的課題で、教師毎に異なる。後者は教育環境の課題で、同一組織で同一科目を担当する教師には共通の課題となる。夫々の内容は表4のようになる。

Vinnervik (2022) は現場教師が話す課題をこの枠組みに分類し、夫々について解説している。これによると、多くの教師は教育の必要性を認識している（I3）が、自分が十分な知識を持っていないと認識しており（I1）、どのように指導したらよいか分からない、自信を

表4 Finger & Houguet (2009) の内的課題と外的課題の分類

内的課題		外的課題	
I1	専門的知識や理解の不足	E1	資源
I2	専門的適切性	E2	時間管理
I3	専門としての態度と価値認識	E3	実行可能性
I4	教育アプローチ	E4	生徒の学習効果の評価
I5	責任感 (ownership)	E5	歴史と伝統
		E6	教師の教育と支援

もって授業をできない、科目の要請なので、取り敢えず Cover するだけ (I2) で、実機を使わないテキストベースでのプログラミング教育だけを行っている事例も多い。IT 基盤 (E1) についても、BeeBot や LEGO のような教育用教材ロボットを購入する資金がなく、教材が揃っていない。教室内の通信速度が遅くて必要なソフトウェアのダウンロードに時間が掛かる、壊れた PC の修理代がなく、PC の代替もできない等の資源制約もある。

Nouri et al. (2020) は、プログラミング教育で小学生が獲得するスキルについて、19名の初等教育 (小学校) のベテラン教師にオンラインでアンケート調査を実施し、狭義及び広義の CT スキルの視点で結果を集計して、報告している。その結果、狭義の CT スキルとしては Brennan & Resnick (2012) の3つのスキル (表1)、広義の CT スキルとしては認知スキルと態度、言語スキル、協働スキルと態度、創造的問題解決のスキルと態度を認識した。この結果によると、プログラミング教育は期待した成果を実現していたと言える。

4.3 英国の経験

Fowler & Vegas (2021) は、英国の経験から学んだことを以下の4点に纏めている。

- ・CS教育のプログラム設計でIT業界の支援を期待できる。しかし実際に教育を行う教師の状況理解が政府にとって重要である。
- ・教師の訓練が最優先事項である。これには多額の投資と長期的な計画が必要。
- ・高品質のCS教育へのアクセスは学校毎に不均一になりがち。人種・性別・地域によってCS教育に差ができる。しかし長期的にはこれは解消に向かう。
- ・CS教育で具体的に何をどのように教育するかは依然として課題である。どのように教育するかの研究が更に必要である。

5. おわりに

5.1 まとめ

CS 教育の一つとしてのプログラミング教育は、大学のような高等教育レベルから始まった。大学では受講生の知的成熟度が高いため、プログラミング教育は比較的实施し易い。更に各国政府は、高度情報社会が浸透するのに伴い、広義の CT を実現するために、国民全てに CS 教育とプログラミング教育を普及させようとしている (Informatics Europe & ACM Europe Working Group on Informatics Education, 2013)。各国は中等教育を経て、次第に小学校のような初等教育にもプログラミング教育を普及させるという政策を進めつつある。これは日本を含めて世界共通の傾向である。高等教育 (高専・短期大学・大学) におけるプログラミング教育は半世紀以上の歴史があり、十分な経験も知識の蓄積もある。しかし初等教育レベルでは、教師も学校も全く未経験である。プログラミング教育を CT に結び付けて考察する研究もまだ少ない (Nouri et al., 2020)。更に初等教育固有の制約要因もあり、教育行政当局 (日本では文部科学省) の強い指導で始まったにも拘らず、様々な課題に直面している。

初等教育へのプログラミング教育導入についても、概念的 연구や学術論文の形での経験の蓄積が少しずつ始まっている。本稿は概念基礎としての CT を端緒として、世界的な CS 教育の推進や初等教育へのプログラミング教育導入についての事例や文献を検討した。前記のように既に様々な CS 導入推進の動きや教育内容の標準化の動きがある。しかし初等教育におけるプログラミング教育導入には、組織 (学校)・教師・生徒の全てのレベルで固有の色々な課題がある。本稿ではこれらの事例論文を参照して、導入の課題を検討した。

5.2 プログラミング教育の広義の CT への効果

教育の全般的な目標としては、狭義の CT よりも広義の CT のほうがより価値あると前記した。これは前記のように文部科学省でも各国でも同じ認識である。しかし Boom et al. (2022) の実証研究によると、プログラミング能力に関係するのは広義の CT ではなくて狭義の CT である。即ち、プログラミング教育で狭義の CT は向上する。しかし広義の CT は (狭義の CT の影響を統制するならば)、プログラミング教育では影響がない。この帰結は、プログラミング教育を促進する場合、狭義の CT 改善 (教育) を重視して、敢えて広義の CT への効果を期待しないということである。これは、プログラミングは特殊な技術であり、プログラミング技術者は技術者であるということを示唆する。勿論、狭義の CT を統制しなければ、狭義の CT は広義の CT の一部なので、広義の CT 改善にも価値があると言える。よって、プログラミング教育に価値がある点は変わらない。但し、広義の CT 改善を大きく期待することはできないかもしれない。

但し, Boom et al. (2022) の実証研究は実証研究の一つに過ぎないので, 類似の研究・研究結果が他にあるかどうか, 今後更に追加的な研究がなされて同じ結論に至るかどうかは不明であり, 上記の推論は一時的なものであることを前提とする必要がある。

5.3 本稿の課題と次の研究段階

本稿は小学校におけるプログラミング教育の, 欧米における研究を一部紹介した。CS 教育についての研究は多数発表されている。しかし初等教育におけるプログラミング教育に限定すると文献研究論文は少ない (Szabo, et al., 2019)。本稿はこの点で学術的貢献があると考えている。

しかし時間的制約のため, 以下のような課題を残している。第一に, 研究論文の検討が散発的 (snowball survey) であり体系的でない。より組織的な文献探索・研究をするためには更に長期的な取り組みが必要である。第二に, 初等教育におけるプログラミング教育は欧米でもわが国でも始まってから間もない。このため, 研究成果の蓄積自体が少ない。基礎となる情報技術は変化が早いので, その内容や範囲が今後急速に変化すると予想されており, 定期的な改訂が必要である (Ilomäki et al., 2011)。更に, 長期的な研究蓄積により, より体系的な考察が可能になるが, 現時点では時期尚早である。

第三に, 本稿では理論的背景としての CT をベースに, そのドライバとしての CS 教育, 更に初等教育におけるプログラミング教育導入とその課題迄, 欧米各国で報告された研究成果を検討した。しかしこの何れのテーマも, それ自体で十分な検討を必要とするものである。本稿ではこれらの関連を重視して夫々を概観したが, 時間制約もあり断片的な考察に終わっている。何れについても, 更に深い検討が必要である。

以上のように, 本稿の内容についても幾つもの課題がある。今後更に再検討することが必要である。

参考文献

- Abbott, N. (2021) *Are we completely wrong in the way we're teaching coding? Yes, probably...* March 13, <https://edtechnology.co.uk/comments/completely-wrong-way-teaching-coding-probably/>
- Balanskat, A. & Engelhardt, K. (2015) *Computing our future: Computer programming and coding Priorities, school curricula and initiatives across Europe*. http://www.eun.org/documents/411753/817341/Computing+our+future_final_2015.pdf/d3780a64-1081-4488-8549-6033200e3c03
- Bebras (2022) *About Bebras*. <https://digitalcareers.csiro.au/en/Bebras/About-Bebras>
- Bell, T., Witten, I. and Fellows, M. (2015). *CS Unplugged: An enrichment and extension programme for primary-aged students*. <https://classic.csunplugged.org/documents/books/>

- english/CSUnplugged_OS_2015_v3.1.pdf
- Boom, K.D., Bower, M., Siemon, J. & Arguel, A. (2022) Relationships between computational thinking and the quality of computer programs. *Education and Information Technologies*, Vol. 27, pp. 8289–8310. <https://doi.org/10.1007/s10639-022-10921-z>
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association*, pp. 1–25. Vancouver, Canada.
- Code.org (2020) *Code.org 2025 Goals: Transforming Global Computer Science Education*. <https://code.org/files/2025goals.pdf>
- Crompton, H. & Sykora, C. (2021) Developing instructional technology standards for educators: A design-based research study. *Computers and Education Open*, December, Vol. 2, <https://doi.org/10.1016/j.caeo.2021.100044>
- European Commission/EACEA/Eurydice (2019) *Digital Education at School in Europe. Eurydice Report*. Luxembourg: Publications Office of the European Union. <https://op.europa.eu/en/publication-detail/-/publication/d7834ad0-ddac-11e9-9c4e-01aa75ed71a1>
- European Commission (2022) The Digital Competence Framework. EU Science Hub, https://joint-research-centre.ec.europa.eu/digcomp/digital-competence-framework_en
- Fagerlund, J., Häkkinen, P., Vesisenaho, M. & Viiri, J. (2020) Computational thinking in programming with scratch in primary schools: A systematic review. *Computer Applications in Engineering Education*, Vol. 29, Issue 1, pp. 12–28.
- Finger, G. & Houguet, B. (2009) Insights into the intrinsic and extrinsic challenges for implementing technology education: case studies of Queensland teachers. *International Journal of Technology and Design Education*, Vol. 19, No.3, pp. 309–334.
- Fowler, B. & Vegas, E. (2021) *How England implemented its computer science education program*. <https://www.brookings.edu/research/how-england-implemented-its-computer-science-education-program/>
- Garvin, M., Killen, H., Plane, J. & Weintrop, D. (2019) Primary School Teachers' Conceptions of Computational Thinking. *SIGCSE '19: Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, February 27–March 2, 2019, Minneapolis, MN, USA, <https://dl.acm.org/doi/proceedings/10.1145/3287324>
- Greifenstein, L., Graßl, I. & Fraser, G. (2021) Challenging but Full of Opportunities: Teachers' Perspectives on Programming in Primary Schools. *21st Koli Calling International Conference on Computing Education Research*, November, <https://doi.org/10.1145/3488042.3488048>
- Heintz, F., Mannila, L., Nordén, L.A., Peter, P. and Björn, R. (2017) Introducing Programming and Digital Competence in Swedish K–9 Education. In *Proceedings of 10th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, ISSEP 2017, Helsinki, Finland, November 13–15. ISBN: 9783319714820 (print) and 9783319714837 (online).
- Iilomäki, L., Kantosalo, A. & Lakkala, M. (2011). What is digital competence? In *Linked portal. European Schoolnet (EUN)*, Brussels, pp. 1–12. https://helda.helsinki.fi/bitstream/handle/10138/154423/Ilom_ki_etal_2011_What_is_digital_competence.pdf

- Informatics Europe & ACM Europe Working Group on Informatics Education. (2013). Informatics in education: Europe cannot afford to miss the boat. Report of the Joint Informatics Europe and ACM Europe Working Group on Informatics Education. Retrieved from <https://www.informatics-europe.org/images/documents/informatics-education-europe-report.pdf>
- ISTE (2007) Profiles for Technology (ICT) Literate Students. Retrieved August 24, 2010, from http://www.iste.org/Content/NavigationMenu/NETS/ForStudents/2007Standards/NETS-S_2007_Student_Profiles.pdf
- ISTE (2007) ISTE standard, <https://www.iste.org/iste-standards>
- Kafai, Y., Proctor, C., & Lui, D. (2020). From theory bias to theory dialogue: embracing cognitive, situated, and critical framings of computational thinking in K-12 CS education. *ACM Inroads* Vol. 11, No.1, pp. 44-53.
- K12 Computer Science Framework (2022) K12 Computer Science Framework, <https://k12cs.org/>
- K12 Computer Science Framework Japan (2022) K12 Computer Science Framework Japan, <https://k12csf.wixsite.com/site/home>
- Nouri, J., Zhang, L., Mannila, L. & Noren, E. (2020) Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, Vol. 11, No. 1, pp. 1-17.
- Papert, S. (1980) *Mindstorms: Children, Computers, and Powerful Ideas*, Basic Books, Inc., Publishers, New York.
- Pérez-Escoda, A. & González-Fernández-Villavicencio, N. (2016) Digital competence in use: from DigComp 1 to DigComp 2. November, in *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality*, DOI:10.1145/3012430.3012583.
- Royal Society (2017) *After the reboot: Computing education in UK schools*, Policy Report.
- Schola Europaea (2020) *Digital Competence Framework for the European Schools*. 2020-09-D-51-en-2, <https://www.eursec.eu/BasicTexts/2020-09-D-51-en-2.pdf>
- STEM learning (2022) *Bee-Bots 1,2,3 Programming Activity*, <https://www.stem.org.uk/resources/elibrary/resource/36026/bee-bots-123-programming-activity>
- Szabo, C., Sheard, J., Luxton-Reilly, A., Becker, B. A., & Ott, L. (2019). Fifteen years of introductory programming in schools: a global overview of K-12 initiatives. In *Proceedings of the 19th Koli Calling International Conference on Computing Education Research*, pp. 1-9.
- UK Government Digital Service (2014) *D5 London: teaching children to code*. 8 December, <https://www.gov.uk/government/publications/d5-london-summit-themes/d5-london-teaching-children-to-code>
- UNESCO (2012) International Standard Classification of Education ISCED 2011. <http://uis.unesco.org/sites/default/files/documents/international-standard-classification-of-education-isced-2011-en.pdf>
- UNESCO (2018) UNESCO ICT Competency Framework for Teachers. <https://unesdoc.unesco.org/ark:/48223/pf0000265721>
- Vinnervik, P. (2022) Implementing programming in school mathematics and technology:

- teachers' intrinsic and extrinsic challenges. *International Journal of Technology and Design Education* Vol. 32: pp. 213-242, <https://doi.org/10.1007/s10798-020-09602-0>
- Vuorikari, R., Kluzer, S. & Punie, Y. (2022) *DigComp 2.2: The Digital Competence Framework for Citizens - With new examples of knowledge, skills and attitudes*, EUR 31006 EN, Publications Office of the European Union, Luxembourg, 2022, ISBN 978-92-76-48882-8, doi:10.2760/115376, JRC128415.
- Wikipedia (2022) *National Centre for Computing Education*. https://en.wikipedia.org/wiki/National_Centre_for_Computing_Education
- Wilson, C., Sudol, L.A., Stephenson, C. & Stehlik, M. (2010) *Running on Empty: the Failure to Teach K-12 Computer Science in the Digital Age*. 6th October, DOI:10.1145/3414583
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, Vol. 49, No. 3, pp. 33-35. <https://www.cs.cmu.edu/afs/cs/usr/wing/www/ct-japanese.pdf> (最終アクセス日：2022年8月13日).
- 中島秀之訳 (2015) 「計算論的思考」情報処理 Vol. 56, No. 6, pp. 584-587.
- くもん出版 (2021) 「小中学校におけるプログラミング教育」に関する調査, https://www.kumonshuppan.com/naruhodo_kumon/
- 情報オリンピック日本委員会ジュニア部会 (2022) 「ビバーチャレンジについて」 <http://da.tani.cs.chs.nihon-u.ac.jp/Bebras/>
- 文部科学省 (2016) 「小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議の設置について」 https://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1370404.htm
- 文部科学省 (2017) 「平成 29・30・31 年改訂学習指導要領 (本文, 解説)」 https://www.mext.go.jp/a_menu/shotou/new-cs/1384661.htm
- 文部科学省 (2020) 「教育の情報化に関する手引―追補版」 https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/mext_00117.html