

小学生向けアンプラグド・プログラミング入門授業 「ハンバーガー・ロボ」の提案と実践

倉橋 農^{1,a)} 越智 徹² 尾崎 拓郎³ 島袋 舞子⁴

概要: 筆者らは、小学生向けにプログラミング的思考を促す目的で、対話的／スクリプト的実行、順次・分岐・反復を盛り込んだアンプラグド授業「ハンバーガー・ロボ」を設計し、実践を行っている。授業内では、教師扮するロボットが、口頭または指示カードによる児童の指示を受け、絵カードでハンバーガーを完成させる。ロボットに対する指示とロボットの行動に仕様が存在し、単純な順次的実行からデバッグまでを、段階に応じ実際のプログラミング言語と対照しながら理解することができるよう設計した。本発表では、この授業案とプログラミング論的な背景を紹介するとともに、2019年5月に大阪電気通信大学で開催した「小学生向けプログラミング教室」における実践について概要を報告する。

キーワード: プログラミング教育, 子ども向け, CS アンプラグド, Switched on Computing, ハンバーガー・ロボ

An Unplugged Programming Lesson “Hamburger Bot” for Elementary School Pupils: A Proposal and Practice

KURAHASI, MINORI^{1,a)} OCHI, TORU² OZAKI, TAKURO³ SHIMABUKU, MAIKO⁴

1. はじめに

1.1 背景と目的

2020年の小学校プログラミング教育導入に向け、小林ら[1]などに代表されるように、小学生向けの実践事例が多数、報告されている。一方で現状では現場の教師や保護者にも、「どの言語を習得するのか」という質問に代表される混乱が見られ、コンピューターショナル・シンキングやプログラミング的思考自体についての理解が追いついていないのが現状である。

筆者らは、小学生向けにプログラミング的思考を促す目的で、対話的／スクリプト的実行、順次・分岐・反復を盛り込んだアンプラグド授業「ハンバーガー・ロボ」を設計し、実践を行っている。授業内では、教師扮するロボットが、

口頭または指示カードによる児童の指示を受け、絵カードでハンバーガーを完成させる。ロボットに対する指示とロボットの行動には仕様が存在し、単純な順次的実行からデバッグまでを、段階に応じ実際のプログラミング言語と対照しながら理解することができるよう設計した。本報告では、この授業案とプログラミング論的な背景を紹介するとともに、2019年5月に大阪電気通信大学で開催した「小学生向けプログラミング教室」における実践について概要を報告する。

1.2 先行する事例

1.2.1 Switched on Computing

イギリスでは、2014年9月から公立小学校1年生(5歳)から、新教科 Computing が実施されている。これ以前には、1995年から教科 ICT が全学年で実施されていたが、コンピュータ・サイエンスの内容が大幅に取り入れられ、ICTが Computing に置き換えられた。Computingでは、教科書や教材、カリキュラムが特に定められてお

¹ 大阪大学 サイバーメディアセンター

² 大阪工業大学 情報センター

³ 大阪教育大学 情報処理センター

⁴ 大阪電気通信大学 メディアコミュニケーションセンター

a) kurahasi.minori@cmc.osaka-u.ac.jp

らず、学校の裁量で自由に決定できる。そのため、様々な教材が提供されているが、Rising STARTS が発行する“Switched on Computing”シリーズが比較的多くの学校で導入されている。この教材は、Computing を推進するための機関 CAS (Computing At School) や Naace (The National Association for all those interested in technology in Education) らと共同で開発され、様々な教育リソースも提供されている [2]。

1.2.2 サンドイッチ・ロボ

前述の Switched on Computing 1年生用の教材 [3], p. 25 では、教師扮するサンドイッチ・ロボが児童の指示を受け、実際のサンドイッチを作るという指導案が紹介されている。サンドイッチ・ロボは指示通りにしか動作せず、不適切な指示を与えると、ナンセンスな動作を行う*1。このサンドイッチ・ロボットは、アルゴリズムという概念の理解、コンピュータが厳密で曖昧性のない指示 (precise and unambiguous instruction) により動作することの理解を目的としている。本報告の提案する授業は、これに着想を得たものである。オリジナルの案と本報告の案の差異については、次節で述べる。

2. ハンバーガー・ロボ

2.1 概要

本報告の提案する授業は前述の Switched on Computing [3], p. 25 に着想を得たものであるが、本節で詳述するように、オリジナルの案とは大きく異なっている。本提案では題材をハンバーガーに変更しており、以後これを「ハンバーガー・ロボ」と呼び区別する。

教師扮するハンバーガー・ロボは、受講者の口頭の指示または単語カード (図 1) を配したスクリプト・シート (図 2) により、食材の絵カード (図 3) を組み合わせて、メニュー (図 4) やリクエストに応じたハンバーガーを完成させる。

ハンバーガー・ロボへの指示と実行は、単純化されたプログラム・コードとその実行を可視化したモデルと考えることができる*2。このロボットに指示を与えハンバーガーを作成する過程で受講者は、厳密で曖昧性のない指示、対話モード/スクリプトによる実行、順次・反復・分岐、ブロック、要件定義や仕様の策定といった、プログラミングの重要な概念を理解可能となっている。

2.2 一般的な特徴

2.2.1 アンプラグド

アンプラグドの手法を採用することで、コンピュータの

操作や環境に気を取られず、プログラミング的な思考・概念に集中が可能となると考えられる。また、実行の際に教師ないし受講者自身の労力が求められるため、あてずっぽうな実行ではなく、筋道を立てて考える必要を実感できる。

また、かつての情報教育には、コンピュータ自体と触れ合うことが目的の一部として含まれていたかもしれない。しかしながら、現在の学習者の多くは、すでにスマートフォンを始めとする情報機器に接する機会を得ており、実際の情報機器への移行に困難はないと思われる*3。

2.2.2 エラー

プログラミング教材の多くでは、できる限り構文上のエラーが生じないように設計されている。Scratch 2.0 以降では、スクリプト・エラー検知機能が除かれた [5], [6] はその例を紹介している。また、ビジュアル言語 Viscuit について開発者である原田は、エラーを出さないことに重点を置いたと述べている [7], [8]。

一方で、ハンバーガー・ロボへの指示は、自由度を高く設定しており、頻繁に構文上のエラーを発生させる。これは曖昧性のない指示が必要であることを強調するとともに、デバッグの訓練も視野に入れているためである。

2.2.3 日本語プログラミング

ハンバーガー・ロボへの指示は実質的に簡単なスクリプト言語の体をなしている*4が、すべて日本語を用いている。同様の日本語を用いたプログラミングの試みには、プロデ

したのパン	うへのパン
おにく	レタス
をとって	をおいて
↓ここから	↑ここまで
回	して

図 1 単語カード

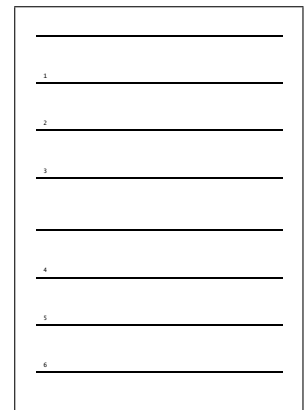


図 2 スクリプト・シート



図 3 食材カード



図 4 メニュー

*1 同書では、類似した「ジャムサンド・アルゴリズム」という授業も提唱されている [4]。

*2 このモデルと実際のプログラミングの対応については、3.3 で改めて取り上げる。

*3 提案手法は最終的に実際のコンピュータ上でのプログラミングへと移行する、あるいはそれと組み合わせて用いることを前提としており、アンプラグド以外の手法自体を否定するものではない。

*4 詳しくは 3.1 で述べる。

ル、ひまわり・なでしこ、ドリトル等が知られており、また Scratch の日本語化は単なる語の置き換えではなく、語順まで考慮した高度なものである。

本提案における言語の特徴は、日本語として完全に自然な文を実現している点であり、その発想はプロデルに近い。例えば、条件に「もし」を前置するのではなく「なら」を後置する、動詞に「～して」と依頼・命令の形を用いる、ブロックに「ここから」「ここまで」を用いる、などが挙げられる。無論、これは高度な処理が用意されていないからこそ実現できることであるが、本提案は基本的にアンプラグドでの基礎概念の理解に重点を置いており、読み上げたときに自然な日本語として理解可能であることに重点を置いた。

2.3 サンドイッチとハンバーガー

本節では特に、Switched on Computing [3] によるサンドイッチ・ロボとハンバーガー・ロボの違いに着目して特徴を述べる。

2.3.1 単語カードとスクリプト・シート

サンドイッチ・ロボは音声^{*5}による対話的な順次処理を扱う。これに対しハンバーガー・ロボでは、単語カードと、それを配置するスクリプト・シートによる指示を導入した。これにより、反復や分岐を含む複雑なスクリプトを記述可能となる。また、事前に文法を定義することでロボットの動作を統一したものとすることができる。サンドイッチ・ロボにおける指示は自由度が極端に高く、ロボットがどのように反応するべきか、その場で教師が判断しなくては行けない。

2.3.2 仕様と実装

サンドイッチ・ロボでは、サンドイッチの完成形を思い浮かべながら指示を出す。対してハンバーガー・ロボでは、あらかじめ用意されたメニュー、あるいは別の受講者の希望にそったハンバーガーを考え、作成のための指示を出す。これは要件定義や仕様の策定と実装を分けて考えることに相当する。

2.3.3 実物と絵カード

サンドイッチ・ロボは実際の食材を用いてサンドイッチを作成する。実物の導入は端的に受講者の興味を引き、また「食べられないものを作ってしまう」という緊張感が生ずることで、真剣な取り組みが期待される。ハンバーガー・ロボでは、食材を教室で扱うことの手間と、複雑なハンバーガーを作成することを考え、各食材を絵カードで表している。

3. ハンバーガー・ロボの仕様

先に述べたように、ハンバーガー・ロボへの指示とその

実行は、単純化されたプログラム・コードとその実行を可視化したモデルと考えることができる。

本節では 3.1 ハンバーガー・ロボへの指示である「ハンバーガー言語」の仕様、3.2 ロボの実行動作に節を分けて紹介し、その後 3.3 で実際のプログラミングにおける概念が、どのようにモデル化されているかを概観する。

3.1 言語仕様

ハンバーガー言語の仕様について、簡単に述べる。この文法について受講者が明示的に意識する必要はないが、ロボットは文法に厳密に従って動作し、非文法的な指令にはエラーを出すため、文法を正確に把握している必要がある。

3.1.1 語彙

単語カードを「カード」、引数を <引数> として表し、対で利用するカードは「=」で結んでいる。

開始命令 「ハンバーガーつくろう！」

関数

<材料> をとって

<材料/ソース> をとって

<ソース> をかけて

材料 うえのパン, おにく, トマト, ...

ソース ケチャップ, カラシ, マヨネーズ, ...

数 1, 2, 3, ...

3.1.2 制御構造

ブロック ↓ここから = ↑ここまで

反復 <ブロック> <数> 回 = して

分岐 <条件> なら = <ブロック> = して

3.1.3 例

図 5 は、図 6 に示される注文「レタス・ケチャップ・パテを 3 重にしたハンバーガーを 2 つ」作るためのスクリプトである。

3.2 実行と動作

3.2.1 動作

事前に定義された文法と語彙に基づく指示にのみ従い、それ以外の指示については「すみません ワカリません」または「エラー デス」との音声出力を行い、動作を停止する。

指示は音声による対話的指示と、スクリプト・シートによるスクリプト的指示が可能である。音声による指示では一文ごとに指示を実行する。すなわち、指示に従って食材コーナーから食材絵カードを取り（をとって）、白板上に貼る（をおいて/をかけて）ことを繰り返してハンバーガーを完成させる。

スクリプト・シートによる指示では、スクリプトの入力後「オイシイ ハンバーガーヲ ツクリマス」と音声出力を行い、バックグラウンド、すなわち白板の後ろでハンバー

^{*5} 「ジャムサンド・ロボット」[4] では単語リストに基づく自由筆記。

ガーを作成し、完成したハンバーガーを白板に提示する。

3.2.2 発話

サンドイッチ・ロボの音声出力は以下の通りである。

「オイシイ ハンバーガーヲ ツクリマス」

開始命令「ハンバーガーつくろう！」に応じて発する。

また、一定時間指示が与えられない場合および任意の質問に対して、ランダムに発する。

「シジ シテ クダサイ」

「スミマセン ワカリマセン」の後に発する。また、一定時間指示が与えられない場合、ランダムに発する。

「スミマセン ワカリマセン」

対話的実行中、指示が非文法的であった場合に発する。

また、任意の質問に対して、ランダムに発する。

「エラー デス」

スクリプト実行中にエラーが生じた場合に発し、動作を停止する。

3.3 モデルとプログラミング概念の対応

表 1 に示すように、ハンバーガー・ロボに対する指示や実行時の動作は実際のプログラミング概念と対応づけられたモデルと考えることができる。将来的に本格的なプログラミングを行う段において、各概念の明示的な理解への寄与が期待される。また、学習するプログラミング言語での実装も効果的であろう。

4. 実践

本節では、上記に紹介したハンバーガー・ロボによる、子ども向けプログラミング教室の実践内容を報告する。これは 2019 年 5 月 4 日に筆者らが大阪電気通信大学にお

る小学生向けイベント*6として実践したものであり、参加者は小学生 3-6 年の 10 人である。(図 7, 図 8)

4.1 概要

教師の扮するハンバーガー・ロボットに児童が指示を出し、ハンバーガーを完成させる。最初は声による命令・実行を順次行う対話モードでの指示を行い、次いで単語カードとシートを導入、スクリプトとして実行する。また、第三者の注文による実行を導入する。その後、注文に従う形で反復、ブロックの概念を導入する。

4.1.1 構成

- 教師：進行役、ロボット役
- 補助：3 名（教材配布、児童の補助）
- 対象：小学校 3-6 年、10 人：3 グループで作業
- 時間：45 分 3 コマ

4.1.2 教材

単語カード 名刺カードに印刷。カードは種類ごとに色を変え、対で利用するカードを紐で結びつける。(図 1)

スクリプト・シート 単語カードを配することができるように、20 行の行番号と罫線を付す。単語カードは両面の粘着テープで貼り付ける。(図 2)

材料絵カード 各食材を A4 に印刷。ロボットが受講者の指示を受け、材料コーナーから選び、白板に貼り付け

表 1 実際のプログラミング概念との対応

ハンバーガー・ロボ	プログラミング概念
命令ミス	タイプミス
ロボット	実行環境
音声による指示と実行	対話モード
カードによる実行	スクリプト・モード
回 = する	反復
なら = する	条件分岐
ここから = ここまで	ブロック
動詞	関数
カード	予約済みテラル
カードの色	シンタックス・ハイライト

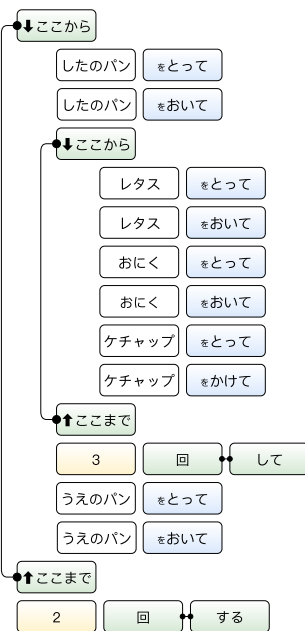


図 5 スクリプト

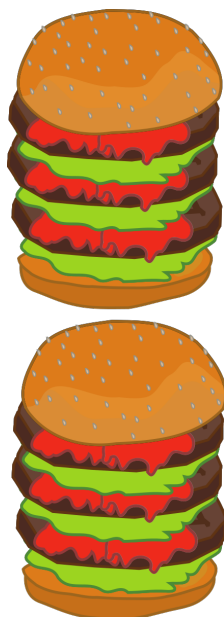


図 6 注文



図 7 ロボット、スクリプト、ハンバーガー



図 8 スクリプトシートと子どもたち

*6 <https://www.osakac.ac.jp/news/2019/1769>

てハンバーガーを作る。(図3)

メニューカード 注文を指定するとき用いる。グループごとに配布する。(図4)

BGM “Parade der Zinnsoldaten”。ロボおよび受講者の作業中に流す。

4.2 シナリオ

4.2.1 導入

【概念】厳密で曖昧性のない指示，対話モードによる実行，逐次処理，要件定義と仕様の策定

受講者を学年が偏らないように3グループに分け，各グループにメニューを配布する。

- (1) ハンバーガー・ロボに指示を出してハンバーガーを作ってもらうことを説明する。
- (2) メニューを提示し，開始命令「ハンバーガーつくろう！」以外の説明無しで指示を出させる。「チーズバーガー！」「チーズバーガー作って！」などが想定されるが，ロボットは「スママセン ワカリマセン」で答える。
- (3) 特定の方法で指示をすれば，正常に動作することを示す。

(厳密で曖昧性のない指示)

- (4) 事前に用意されたメニューから選ばせ，指示を出させる。非文法的な指示に対しては「スママセン ワカリマセン」と応答する。

(厳密で曖昧性のない指示)

- (5) 誰かのリクエストを聞き，どんなハンバーガーか把握した上で指示を出させる。

(要件定義と仕様の策定)

4.2.2 スクリプト・カードの導入

【概念】スクリプトによる実行，逐次処理，デバッグ，要件定義と仕様の策定

スクリプト・シートの導入以降，ロボットはスクリプトを読み，ホワイトボードの影で一気に入成する。完成したハンバーガーをスクリプトとともにホワイトボードに提示する。エラーが出る行までは1行ずつ実行し，エラーが生じた時点で「エラー デス」と言って，スクリプトと作りかけのハンバーガーを提示する。(デバッグ)

- (1) 一度に把握できないような複雑な注文を教師が出す。
「おにくが3枚，激辛がいいな，ピクルス山盛り，あと，おにくの間にチーズ挟んでね。」(要件定義と仕様の策定)
- (2) メモや絵カードを用いて，黒板上に注文内容を整理する。
- (3) 単語カードとスクリプト・シートを導入し，グループごとにスクリプトを作成，それぞれ実行する。

- (4) ロボットがスクリプトを実行する間，ほかのグループのスクリプトを皆でレビューする。

- (5) エラーが生じた場合，あるいは期待と違うハンバーガーが出来上がった場合にどうすればよいか検討する。(デバッグ)

4.2.3 反復とブロック

【概念】反復，ブロック，デバッグ，要件定義と仕様の策定

- (1) 「レタス・ケチャップ・おにくを3重にしたハンバーガー」という注文を，曖昧で人間的な表現で出す。(要件定義と仕様の策定)
- (2) 回 = する を導入。(反復)
- (3) ↓ここから = ↑ここまで の必要性を述べる。(ブロック)
- (4) 上記のハンバーガーを複数個注文する。(反復の入れ子，図5, 図6)

4.3 家庭用教材の配布

授業後に単語カード，食材カードをA4に印刷したものを配布し，ハンバーガー・ロボを家庭でも遊ぶように促した。

5. 実践の結果と考察

5.1 授業

5.1.1 授業中の観察

- ロボットが融通をきかせないことについて「ポンコツ」「壊れている」という(嬉しそう)な反応があった。厳密な指示が必要なことを楽しみながら理解できていたと思われる。
- 「すでに習ったので退屈」という発言があった。同一の受講者が反復の入れ子で間違ったコードを作成するなど，実際には有意義な部分があったが，経験者に「退屈」と感じさせない配慮が必要である。
- 一方で，複雑なスクリプト作成に参加できない受講者もいた。授業自体は楽しんでいる様子であった。
- 3コマ目には明らかに受講者に疲れが見られた。小学生に3時間は長く，実施時に配慮が必要である。これについては，アンケートでも意見としてあげられていた。
- 同室で類似のイベントが行われており，仕切がなくお互いの声も聞こえるため，集中が難しかった。こちらも同様に，アンケートでの指摘を受けた。

5.1.2 教材

- 単語カードには名刺カードを利用したが弱く，再利用ができない。また，スクリプト・シートに単語カードを貼り付ける際，粘着テープが剥がれてしまった。ラミネート加工，磁石やマジックテープを利用することを検討する。

- スクリプト・シートが短すぎる場合があった。長くするか、複数のホワイトボードを用意しシートの代りにすることが考えられる。また、事前にどの程度の行数が必要が確認が必要である。

5.2 事後アンケート

授業後、電子メールにてアンケート・フォームへのリンクを送付し、保護者からの回答を募った。回答者は7名である。

6年生の1受講者について、Scratchを体験済み、内容についての理解度も高かったが、保護者から見て「とても楽しそうだった」との回答であった。アンプラグドで本質的な概念を理解する、という目的が達成された例であろう。ただし、保護者は「とても楽しかった」との感想であり、配布した教材を家庭でも「利用するつもり」、今後も参加を「希望する」と回答しており、もとより意識・動機が高いことも影響している可能性がある。同じく6年生の受講者について、配布教材は「利用しない」、今後も「希望しない」、「実際にパソコンを使わないと、6年生にとっては退屈そうでした。」との感想が得られた。すでにコンピュータを用いた教育を体験しており、PCを使う授業を期待していた可能性もある。今後、プログラミング経験の有無によらない、あるいは応じたクラスを検討する必要がある。

5.3 総論

全体として、アンプラグドの手法を用い、プログラミングの基本的な考え方を自然に伝える授業として、最低限の目標は達成できたと考えている。ただし、明らかに改善が必要な点があることについては、5.1, 5.2 で述べたとおりである。

本実践では理解度テストを行わず、保護者へのアンケートに設問を設けるにとどめた。プログラミング概念を理解する教育の紹介という側面が強い企画であり、純粋な実験としてデザインしなかったことが主な理由である。今後は、この教材自体を用いた事後評価を計画している。

もう一点、本実践では10名の受講者について司会役1名、ロボ役1名、補助者3名を配した。実際に小学校で導入するには、スタッフを減らしての実現が可能であるかの検証が必要であろう。

6. 展望

以上、本報告で提案するハンバーガー・ロボと、それを用いた授業の実践について紹介した。本節では、今後の展望および計画について述べる。

6.1 大人向け授業

筆者らは、本報告の子ども向け授業とリンクした大人向けコースを提唱し、保護者・教師の理解を深めることを目

指している。これには子ども向けのコースの内容を検討・実践するだけでなく、型や変数への代入、関数の定義といった高度な概念の導入も含まれる。このコースについては、2019年6月に実施される情報処理学会コンピュータと教育研究会150回研究発表会で発表を行った[9]。

具体的な実践活動として、小規模クラスでの大人向けコースを試験的な開講を予定している。また、小学校教師を対象として、授業用小学生向けコースの紹介と、プログラミング入門を併せたコースの計画を進めている。

6.2 展望

最終的には、大人向け・子ども向け・教師向けの教材とアイデアを有機的に結びつけ、プログラミングの概念について学ぶ体系的な仕組みを作り上げることが、筆者らの構想である。それに伴い、以下のような計画を検討中である。

- 現在の大人向けのコースを大学生の初年度教育用にアレンジする
- 授業用リソースをウェブ上で提供するなど、広く利用可能な状態にする
- ハンバーガー言語のコンピュータ上での実装
- ハンバーガー・ロボのAR/VRアプリ化
- ハンバーガー・ロボ以外の題材を用いた教材の開発

謝辞 大阪電気通信大学における小学生プログラミング教室に参加いただいた小学生と保護者の皆さんに感謝します。

参考文献

- [1] 小林祐紀, 兼宗進, 白井詩沙香, 白井英成 (著, 監修, 編集): これで大丈夫! 小学校プログラミングの授業 3+α の授業パターンを意識する [授業実践 39], 翔泳社, 2018.
- [2] 石塚丈晴, 堀田龍也: 英国の公立小学校における教科「Computing」におけるプログラミング教育の内容, 情報処理学会研究報告, Vol. 2015-CE-131, No. 12, pp. 1-4 (2015).
- [3] Miles Berry, “Switched on Computing Year 1” (2nd Edition), Rising Stars, 2014.
- [4] “Jam Sandwich Algorithm” <http://code-it.co.uk/unplugged/jamsandwich> (2019/05/14 閲覧)
- [5] “Error Catching,” Scratch Wiki https://en.scratch-wiki.info/wiki/Error_Catching (2019/05/14 閲覧)
- [6] 「Scratch はエラーにおおらかなプログラミング言語」. <http://www.howisit.jp/2016/05/21/forgive-error/> (2019/05/14 閲覧)
- [7] 原田康徳: 子供向けビジュアル言語 Viscuit とそのインタフェース, 情報処理学会研究報告, Vol. 2005-HCI-116, No. 114, pp. 41-48 (2005)
- [8] 原田康徳: ビスケットー誰でもプログラミングの楽しさを一, E スクエア・エボリューション成果発表会, (2005). <http://www.cec.or.jp/e2e/symp/17seikapdf/E2-A2.pdf> (2019/05/14 閲覧)
- [9] 倉橋農, 越智徹, 尾崎拓郎, 島袋 舞子: 子ども向け授業にリンクした保護者・教師向けプログラミングコースの検討, 情報処理学会研究報告, Vol. 2019-CE-150, No. 9, pp. 1-5, 2019.