

令和 5 年 6 月 29 日現在

機関番号：58001

研究種目：基盤研究(C)（一般）

研究期間：2019～2022

課題番号：19K03104

研究課題名（和文）CSアンプラグドからフルフレジド・プログラミングへの展開

研究課題名（英文）Advancing from CS Unplugged to full-fledged programming

研究代表者

玉城 龍洋 (Tatsuhiko, Tamaki)

沖縄工業高等専門学校・メディア情報工学科・教授

研究者番号：60413837

交付決定額（研究期間全体）：（直接経費） 3,300,000円

研究成果の概要（和文）：CSアンプラグドはコンピュータを使わずに情報科学を体験的に学習する、学習効果の高い教育方法であるが従来の研究の多くはCSアンプラグドの体験段階で終わっており、本格的なフルフレジド・プログラミング教育への接続性の観点からは限定的であった。そこで本研究では、CSアンプラグドをフルフレジド・プログラミングに接続するCSアンプラグドから本格的なプログラムコード記述までの6段階のステップの典型的な教材の開発を行い、学生に対して評価実験授業を実施し、その効果を確認した。さらに双方向写経型学習という新しいメソッドの提案およびブロックプログラミング言語の活用という新たなアプローチの実践も行うことができた。

研究成果の学術的意義や社会的意義

これまでもCSアンプラグドを使ったコンピュータサイエンスの導入教育は数多くあったが実際にプログラミングを行おうとするとその要求される知識と技術の差で躓く学習者が多くいた。本研究では、CSアンプラグドの手法と本格的なプログラミング言語の学習の橋渡しを行う学習手法を提案し、教材を作成した。高等教育機関における学習手法を主題に置いているが、小中学校におけるプログラミング教育にも大いに活用可能であり、今後のプログラミング教育に活用可能だと考えている。

研究成果の概要（英文）：CS Unplugged (Computer Science Unplugged) is an effective educational method for experiential learning of information science without using computers. However, much of the previous research has remained at the stage of experiencing CS Unplugged, and its connectivity to full-fledged programming education has been limited. Therefore, in this study, we developed a typical teaching material consisting of six stages, from CS Unplugged to the description of full-fledged program code, to establish a connection between CS Unplugged and full-fledged programming. We conducted evaluation experiments in a classroom to confirm its effectiveness. We proposed a new Copying Learning method and also explore the practical application of block programming languages.

研究分野：情報工学

キーワード：CSアンプラグド CSプラグド フルフレジド・プログラミング プログラミング的思考

### 1. 研究開始当初の背景

先進国においてプログラミング教育の流れが加速していく中、コンピュータを使わずに情報科学を教授する手法としてCSアンプラグドが注目されているが、先行研究では、本格的なプログラミング教育への接続性についてあまり研究されて来なかった。そこで、本研究では、CSアンプラグドから本格的なプログラミング教育への橋渡しをする教育方法として、CSアンプラグドを補完する新たな教育方法である「CSプラグド」と本格的なプログラミングへの展開を図るための「6ステップメソッド」を提案し、導入段階における効果的なプログラミング教育のための普遍的な教育メソッドの確立を目指した。

### 2. 研究の目的

CSアンプラグドはコンピュータを使わずに情報科学を体験的に学習する、学習効果の高い教育方法であり、海外では多く研究されている。しかしながら、従来の研究の多くはCSアンプラグドの体験段階で終わっており、本格的なプログラミング(フルフレジド・プログラミング)教育への接続性の観点からは限定的であったりする。そこで、CSアンプラグドを補完する新しいアプローチとして「CSプラグド」という方法を提案する。すなわち、CSアンプラグドは「コンピュータを使わない」ことが基本思想であるが、フルフレジド・プログラミングへの展開を目指すためには、橋渡しとなる「コンピュータを用いた」教育手法が不可欠であることを提起する。そして、計算論的思考の中核となるプログラミング的思考を養うための教育メソッドとしての有効性を明らかにする。

### 3. 研究の方法

以下のフェーズで研究を推進した。

#### (1) CSアンプラグドの代表的なActivityのプラグド化

CSアンプラグドで取り上げるトピックのことを「Activity」(活動)とよぶ。また、我々が独自に定義する用語として、CSアンプラグドのActivityを、コンピュータを用いて実現することを「CSプラグド」と呼ぶことにする。



図1 点を数える(2進数)

CSアンプラグドのActivityには、「Activity1:点を数える(2進数)」(図1)や「Activity2:色を数で表す(画像表現)」(図2)など、情報科学の分野ごとにトピックが用意されている。本研究の特筆すべき点としては、「CSプラグド」の導入が挙げられる。これを「CSプラグド・ツール」と呼ぶことにする。

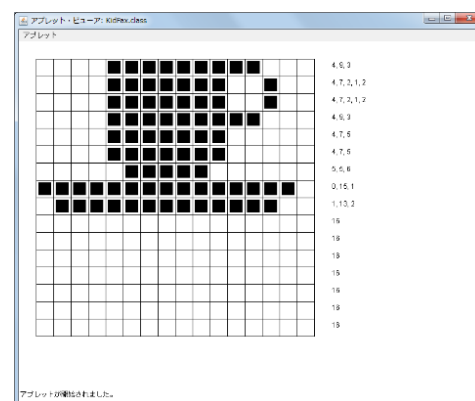


図2 色を数で表す(画像表現)

このツールを用いてコンピュータ上で実施したほうが効果的なActivityの利用や日本の標準的な授業環境での利用およびアンプラグド体験後の振り返りあるいは再体験による理解の深化と定着を目指すものである。

#### (2) 6ステップメソッドの開発

CSアンプラグドは情報科学の教育に有効であるとされており、関連する研究としては、CSアンプラグドのアルゴリズム学習において教具を利用するという研究などがある。しかしながら、CSアンプラグドから本格的なプログラミング言語への発展についてはほとんど研究されていないのが現状である。そこで、新たにCSアンプラグドから本格的なプログラミングへの展開を図るための手法を提案する。すなわち、Step1: CSアンプラグド Step2: CSプラグド Step3: Activityの処理過程の図解(可視化) Step4: Activityの処理過程の自然言語記述(抽象化、要素抽出) Step5: 独自の表現を付加したテーブル・トレースによるアルゴリズムの動作確認(検証) Step6: 本格的なプログラム・コードの記述(抽象化、符号化) という教授法を確立する。

#### (3) 双方向写経

6ステップメソッドのStep4を補完する学習メソッドとして新たに「双方向写経」を考案した。プログラミング教育の実質化に資する教育方法の一つとしてプログラミング初学者向けの「双方向写経型学習」を提案するものである。経験的に、プログラミング習得における「写経型学習」は効果的とされているが、我々は、さらなる効率化および理解度の深化を目指した手法を

模索した。写経のもとの意味は、仏教の経典を書き写すことである。写経には、宗教的意味があるが、ここでは「文字を写す」という外観的行為のみをとらえている。すなわち、お手本となるプログラム・コードを見て、そのままキーボードからコンピュータに入力する作業を意味する。そのため、プログラム・コードを入力して実行しながら理解してプログラムを作成する方法であることから「写経プログラミング」と呼ばれる。また、写経行為を通じて学習がなされることに着目して「写経型学習」と呼ばれることもある。そこで我々は事例からの概念学習を促進する学習方法として「双方向写経型学習」を提案する。すなわち、プログラム・コードの入力から実行という写経型学習に加えて、コードを自然言語により記述することにより抽象化の手順を踏み、さらに類似のコードについての説明から逆にコードを生成することにより概念移転を目指すものである。

双方向写経では、表と裏からなる、紙媒体のワークシートを準備する。ワークシートの表には教科書のサンプルプログラムを掲載し、裏にはそれに類似した内容のプログラムの説明を記載しておく。

ワークシートの表には、タイトルとして「双方向写経(表)」と記載する。その下部の左側にサンプルのプログラム・コードを記載し、その右横に行ごとに説明を書き込むことができる空欄を準備する(図3)。

ワークシートの裏には、タイトルとして「双方向写経(裏)」と記載する。その下部の左側に作成すべきプログラム・コードに対応した行ごとの説明を記載し、その右横に説明に従ったプログラム・コードを行ごとに書き込むことができる空欄を準備する(図4)。

タイトル	
サンプルプログラムの概要	
サンプルのプログラム・コード	説明記述欄(空欄)

図3 ワークシート「双方向写経(表)」の様式

タイトル	
作成すべきプログラムの概要	
作成すべきプログラムの説明	プログラム・コード記述欄(空欄)

図4 ワークシート「双方向写経(裏)」の様式

#### (4) ブロックプログラミング言語の活用

6ステップメソッドのStep6を補完する学習メソッドとして新たに「ブロックプログラミング言語の活用」を提案した。

プログラミング教育とりわけアルゴリズム教育における学習補助手段としては、フローチャートやトレースなどがよく使われている。それらを活用しながら、最終的には何らかのプログラミング言語を用いて記述・実装する訳であるが、初学者にとっては、その作成過程は必ずしも容易ではないとされている。

そこでまず、ブロックプログラミング言語を含むビジュアル型言語とテキスト型言語との関係について概観する。

ビジュアル型プログラミング言語からCやJavaなどのテキスト記述型言語へのシームレスな移行が考慮されていないという問題意識から提案され、プログラミングの学習が進行するにつれて、BlockEditorからJavaへ徐々に移行していくこと、およびそのタイミングには個人差があることが定量的に示された。また、プログラミングに苦手意識を持つ学生ほどビジュアル型言語の選択率が高く、言語の相互変換環境が言語の交ぜ書きを促進し、Java言語習得の足場かけとなることが示されており、ブロック型言語の有効性の一端が明らかにされている。

そこで、本研究では導入難易度が低く学習コストが少ないブロックプログラミング言語を併用することとした。代表的なブロックプログラミング言語にはScratch、MakeCode、Blocklyなどがあるが、本研究ではEduBlocksを用いることとした。その理由は、Python言語への変換機能を有するためテキスト型言語への接続性が高いことである。

EduBlocksはジョシュア・ロウが主宰するEduBlocksプロジェクトにより開発されているビジュアルブロックベースのプログラミングツールである。ブラウザ上でブロックを組み合わせる記述・実行でき、その名のとおりに教育に適した設計となっている。EduBlocksは、他のビジュアル

ル言語と同様に初心者にもわかりやすいツールでありながら内部で Python 言語が動作するため、将来の発展性や拡張性にも期待できるという特徴がある。また、グラフ表示や Web スクレイピングなどにも対応できる機能も備えている。その機能は、平成 30 年告示高等学校学習指導要領に対応した令和 7 年度大学入学共通テストからの出題教科・科目のサンプル問題『情報』の一部を容易にプログラムとして実装できることでも確認できる(図 5)。

したがって、アルゴリズム学習の入門レベルに十分対応できかつテキスト型言語への親和性も高いといえる。

EduBlocks の Python3 モードについて概要を説明する。機能的には Python3 のコア部分を実装されており、さらに直接 Python コードを埋め込むブロックを用いることでより応用的な利用も可能となっている。ユーザーインターフェースは日本語対応ではなく英語表記であるが、テキスト型言語への接続性を考えると必ずしも障壁ではなく、むしろ親和性が高いと言える。

ブロックを組み合わせで作成したプログラムはそのまま Python 言語のコードに対応しており、ブロックを配置すると即座に Python のソースコードが生成される仕組みとなっている。また、ブロックと Python コードを並べて表示することもできる。なお、EduBlocks プロジェクト・ファイルに加え Python コードのダウンロードも可能となっている。EduBlocks は Scratch のようなキャラクタを利用したゲーム性のあるプログラムの作成用途には向いていないが、アルゴリズム教育のような知的技能の醸成を目指した用途には十分な効果が期待できるものである。



```
# Start Code Here
Tomei = ["A 覧","B 覧","C 覧","D 覧"]
Tokuhyo = [1200,660,1440,180]
sousuu = 0
giseki = 6
for m in range(0,4):
    sousuu = sousuu + Tokuhyo[m]
kizyunsuu = sousuu / giseki
print("基準倍率数:", kizyunsuu)
print(" 比例配分 ")
for m in range(0,4):
    print(Tomei[m], ":", Tokuhyo[m] / kizyunsuu)
```

図 5 サンプル問題『情報』の実装例

#### 4. 研究成果

本研究では、CS アンプラグドをフルフレッド・プログラミングに接続する 6 段階である Step1 から Step6 (Step1:CS アンプラグド、Step2:CS プラグド、Step3: Activity の処理過程の図解(可視化)、Step4:Activity の処理過程の自然言語記述(抽象化、要素抽出)、Step5:独自の表現を付加したテーブル・トレースによるアルゴリズムの動作確認(検証)、Step6:本格的なプログラム・コードの記述(抽象化、符号化))までの各段階に対応する典型的な教材の開発を行い、学生に対して評価実験授業を実施し、その効果を確認した。さらに、双方向写経型学習という新しいメソッドの提案およびブロックプログラミング言語の活用という新たなアプローチの実践も行うことができた。

研究期間の後半では、新型コロナウイルス感染症の影響により、十分な実験授業を行うことができなかったが、導入段階における効果的なプログラミング教育のための普遍的な教育メソッドのフレームワークの一案として一定の評価が得られた。

今後は、本研究で得られた知見をもとに、さまざまな場面で実践を行うとともに、より汎用的かつ効果的な学習メソッドとしての確立を図っていきたい。

## 5. 主な発表論文等

〔雑誌論文〕 計4件（うち査読付論文 4件/うち国際共著 0件/うちオープンアクセス 3件）

1. 著者名 Tatsuhiko Tamaki, Atsushi Onishi, Hironori Kobayashi, and Yasuo Uchida	4. 巻 Vol. 4, No.3
2. 論文標題 AN ATTEMPT AT ELEMENTARY SCHOOL LEVEL PROGRAMMING EDUCATION UTILIZING WASAN, TRADITIONAL JAPANESE MATHEMATICS	5. 発行年 2022年
3. 雑誌名 International Journal of Education and Pedagogy	6. 最初と最後の頁 241-252
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -
1. 著者名 TAMAKI Tatsuhiko, ONISHI Atsushi, UCHIDA Yasuo	4. 巻 Vol. 3, No. 10
2. 論文標題 LEARNERS' CONFESSION FOR BIDIRECTIONAL TRANSCRIPTION EFFECTIVENESS FOR BEGINNERS IN THE PROGRAMMING COURSE	5. 発行年 2021年
3. 雑誌名 INTERNATIONAL JOURNAL OF MODERN EDUCATION	6. 最初と最後の頁 32-47
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -
1. 著者名 TAMAKI Tatsuhiko, ONISHI Atsushi, UCHIDA Yasuo	4. 巻 3(1)
2. 論文標題 A Trial of Learning Programming Using a Six-step Method	5. 発行年 2021年
3. 雑誌名 Asian Journal of Research in Education Social Science	6. 最初と最後の頁 10-24
掲載論文のDOI（デジタルオブジェクト識別子） なし	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -
1. 著者名 内田 保雄, 玉城 龍洋, 大西 淳, 田辺 誠	4. 巻 2019年1巻
2. 論文標題 プログラミング導入教育におけるCSアンブラグドの活用	5. 発行年 2019年
3. 雑誌名 情報教育	6. 最初と最後の頁 24-32
掲載論文のDOI（デジタルオブジェクト識別子） 10.24711/rrie.1.0_24	査読の有無 有
オープンアクセス オープンアクセスとしている（また、その予定である）	国際共著 -

〔学会発表〕 計3件（うち招待講演 0件 / うち国際学会 2件）

1. 発表者名 内田保雄（宮崎産業経営大学）, 玉城龍洋（沖縄工業高等専門学校）, 大西淳（津山工業高等専門学校）
2. 発表標題 アルゴリズム教育におけるブロックプログラミング言語の利用
3. 学会等名 情報処理学会 コンピュータと教育研究会 162回研究発表会
4. 発表年 2021年

1. 発表者名 Tatsuhiko Tamaki, Atsushi Onishi, Yasuo Uchida
2. 発表標題 A Trial of Learning Programming Using a Six-step Method, Conference Proceeding
3. 学会等名 International Multidisciplinary Conference on Education, Engineering and Social Sciences (IMCEES2020) (国際学会)
4. 発表年 2020年

1. 発表者名 Tatsuhiko Tamaki, Harumi Hashimoto, Atsushi Onishi, Yasuo Uchida
2. 発表標題 A Trial of Bidirectional Transcription Learning for Beginner Programmers
3. 学会等名 The 3rd International Conference on Humanities and Social Sciences (国際学会)
4. 発表年 2019年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

-

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
研究分担者	内田 保雄	宮崎産業経営大学・経営学部・教授	
	(UCHIDA Yasuo)		
	(70321487)	(37602)	

6. 研究組織（つづき）

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
研究 分 担 者	大西 淳  (ONISHI Atsushi)  (60311073)	津山工業高等専門学校・総合理工学科・教授     (55301)	

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件

8. 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関