

子供向けコンピュータ科学教育法の e-Learning 化

An e-Learning System for "Computer Science Unplugged"

多久和 希美†
Nozomi Takuwa

福岡 久雄†
Hisao Fukuoka

1. はじめに

コンピュータ関連技術の発展と普及によって、我々の日常生活はコンピュータ関連技術なしには成立しなくなっている。これに伴って、コンピュータの誤った使い方や、それに対する過度の期待など負の側面が散見されるようになってきた。コンピュータ関連技術の限界を知り、それを正しく使いこなすことは、技術者のみならず、一般の人々にとっても重要な課題となっている。

このような状況を受けて、平成15年からは、高校課程において科目「情報」が必修科目として設置され、コンピュータ関連技術を一般常識とみなした教育が開始された。今後、この傾向が加速し、初等・中等教育課程においても、同様の基礎教育を行う方向に進むと考えられる。

初等・中等教育課程におけるコンピュータ関連技術の教育では、生徒に数学等の前提知識を大きく期待できないことから、教授法に関して何らかの工夫が必要である。そのような工夫の一例として、ニュージーランドCanterbury大学のTimothy Bell氏等によって提唱されているComputer Science Unpluggedがある¹⁾。Computer Science Unpluggedでは、一見してコンピュータ関連技術とは無関係の各種カードゲームなどを用いることによって、「コンピュータを使うことなく」、各種コンピュータ関連技術の考え方を教授することを目指している。我が国でも、高校の科目「情報」において、この教授法を試行する動きが見られる²⁾。

本研究では、このComputer Science Unpluggedのe-learning化を行い、初等・中等教育現場での実地検証を通して、その有効性を評価することを目的とする。

2. e-Learning 化の意義

Computer Science Unpluggedの特長は「コンピュータを使わない」という点にあり、その指導書には「時には、コンピュータは学習から意識をそらす存在になる」という記述まで見られる。これとコンピュータを積極的に使おうとするe-learning化は矛盾するように思われる。これに対する筆者らの考えは以下の通りである。

最近の子供達は、生まれた時からコンピュータ関連技術に取り囲まれて育っており、携帯電話やゲーム機を含めて、コンピュータは余りも身近な存在となっている。そのため、コンピュータの存在が学習意識をそらすとい

う危険性は低く、むしろ、慣れ親しんだ存在であるコンピュータを使うことによって、より自然な学習環境を提供できると判断している。

† 松江工業高等専門学校専攻科
Advanced Engineering Faculty
Matsue College of Technology

従って、本研究では「コンピュータを使わない」という記述の意味を、「直接的にコンピュータを対象とした学習をさせない」という意味に解釈している。

上記判断の妥当性については、実地検証による従来型のComputer Science Unpluggedの比較を行うことが必要であり、それもスコープに入れて本研究を進める予定である。

3. Computer Science Unplugged

Computer Science Unpluggedの教員向け指導書は、各種コンピュータ関連技術の原理を分かりやすく教えるための12個のActivities(学習項目)で構成されている。表1に12個のActivityを示す。このように、コンピュータの基本的な概念が12のActivityに割り当てられており、いずれのActivityもゲーム要素を取り入れて、子供達の興味を引く内容となっている。指導書には、各Activityの終了後に、そこで学んだ原理が実際のコンピュータではどのように生かされているかの解説がまとめられており、必要に応じて、教員がその解説を行うようになっている。

この指導書は一般に公開されており、Canterbury大学のホームページからダウンロード可能である¹⁾。また、最近になって、その訳書が出版されている³⁾。

表1 Activities

Activity	title	content
1	点を数える	2進数
2	色を数で表す	画像表現
3	それ、さっきも言った!	テキスト圧縮
4	カード交換の手品	エラー検出とエラー訂正
5	20の扉	情報理論
6	戦艦	探索アルゴリズム
7	一番軽いと一番重い	整列アルゴリズム
8	時間内に仕事を終わろ	並び替えネットワーク
9	マッディ市プロジェクト	最小全域木
10	みかんゲーム	ネットワークにおけるルーティングとデッドロック
11	宝探し	有限状態オートマトン
12	出発進行	プログラミング言語

4. システムの構成

多数の生徒による同時使用および遠隔地からのインターネット経由のアクセスを考慮して、本システムはウェブサーバ上に実装する(図1)。生徒側端末としては通常のWindows PCを想定し、ウェブブラウザを通して本システムを利用する。システムの特性上、アニメーション機能の使用が望ましいことから、サーバ側アプリケーションの実

装言語にはAdobe Flashを使用する⁴⁾。

また、一部のActivityでは2人以上の生徒が共同して取り組むものがある。これらのActivityに関しては、排他制御機能などを備えた共有型アプリケーションとして実装することが必要である。

なお、生徒側端末として携帯ゲーム機を想定したシステムを別途開発中である。携帯ゲーム機では一般的にFlashが稼動しないことから、その開発ではPHPを用いたCGIプログラムとして本システムを実装中である。

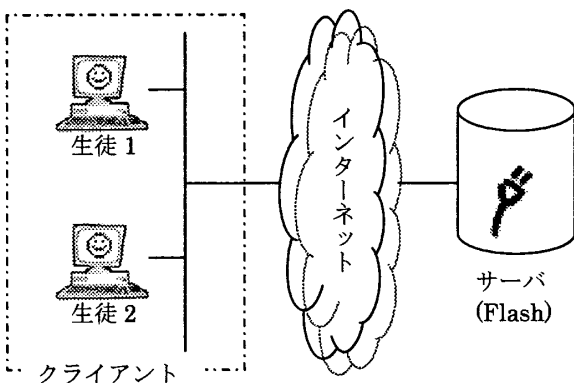


図1 システムの構成図

5. Activityの実装

5.1 実装したActivities

現在はActivity1, 2, 4, 6の基本問題について実装が終了した段階である。ここでは、各Activityの実装内容を簡単に説明する。

Activity3と5においては、言語に依存した課題を扱う。Computer Science Unpluggedは英語圏で生まれたものであり、Activity3と5は英文の構造に依存したものとなっている。従って、これを直接的に日本語化するのは困難と考え、その対策を検討中である。この詳細については後述する。

(1) Activity1

Activity1では2進数について学習する。対象は数を数えることができる7歳以上の子供である。

まず、片面だけにそれぞれ8個、4個、2個および1個の点が描かれた4枚のカード（4桁の2進数の場合）を用意し、それらを降順に整列させておく。各カードの点の個数は右側のカードのその倍になっていることを確認させる。すなわち、左にもう1枚カードを加えたとしたら、点の数が16のカードを用意する必要があることを理解させる。

このようなカードの表裏状態を変更することで、いろいろな数を作ることが可能となることを体験する。例えば、10の数を作るよう子供たちにクイズを与える。子供たちは点の数を数えて10になるようカードをめくり、図2.1(a)のような状態にする。カードは、裏返した状態を0、表の点が見える状態を1とすると、これは左から1010という2進数となる。1010は10進数で表現すると10であり、点の数と等しくなる。このようなクイズを繰り返すことで、2つの記号を使って数字を表現できるという事実を知ることが出来る。同時に、10となるカードの状態が一意である

ことを体験的に学ぶこともできる。

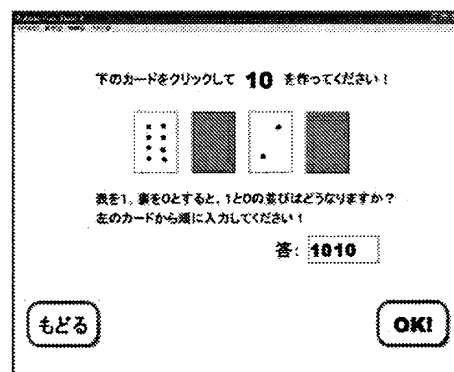


図2.1(a) Activity1 (2進数への変換)

また、2進数をカード表現することから始めて、見えている点の総数を求めることで、2進数から10進数への変換を印象付けるクイズも実装した(図2.1(b))。

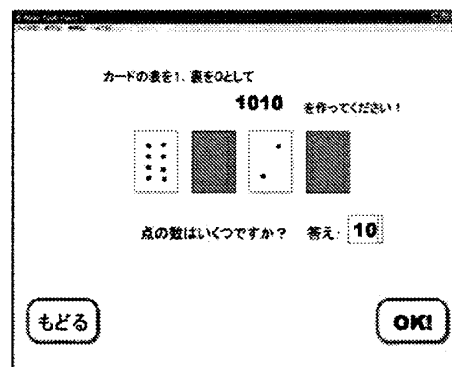


図2.1(b) Activity1 (10進数への変換)

(2) Activity2

Activity2では画像表現の学習を行う。対象は、数を数えることができる7歳以上の子供である。

用意するものは縦横に区切られたマス（ピクセル）の書かれた紙で、各行には数字のコードを書いておく必要がある。マス全体は画像のビットマップ表現に相当し、各行に付随する数字コードは、その行のピクセル情報をランレングス表現したものとなっている。

このActivityでは、各行のランレングス表現からビットマップを描く作業や、逆にビットマップから各行のランレングス表現を書き出す作業を繰り返す。これによって、コンピュータでは画像がビットマップにより表現されることを知識として習得するとともに、それと等価な表現方法としてのランレングスの考え方を身に付けることができる。

Activity2の実装画面を図2.2に示す。これは、かな文字「あ」のフォントのビットマップ表現とランレングス表現である。

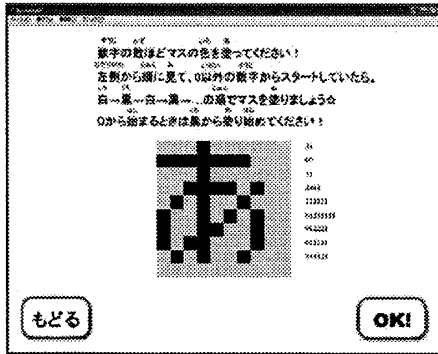


図 2.2 Activity2

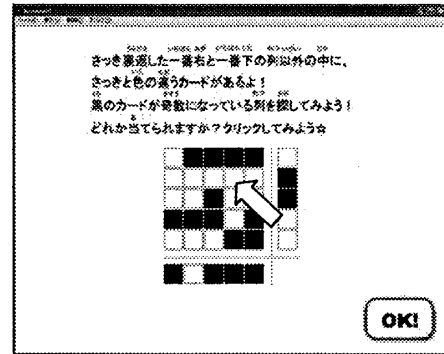


図 2.3(b) Activity4 (パリティチェック)

(3) Activity4

Activity4の実装画面を図2.3に示す。Activity4では水平垂直パリティ符号を用いたエラー検出とエラー訂正の学習を行う。対象は、数を数えることができ、偶数と奇数が理解できる9歳以上の子供である。

このActivityを学習する際に必要となるものは表が白、裏が黒のカードである。まず、コンピュータが図2.3(a)のように正方形に配置された25枚のカードの表裏状態をランダムに設定する。子供たちは、各行、各列の黒の個数が偶数となるように、右端、下端のカード(パリティカード)の表裏を設定する。その後、コンピュータが1枚のカードをランダムに選び、そのカードを裏返す。これに対して、子供たちが、図2.3(b)のように、黒カードの数が奇数となっている列と行の交点から、コンピュータが変更したカードを検出する。

この作業を繰り返すことで、カードの状態が変化した場合、どのようにして変化したカード(エラー)を検出して訂正するかを学習させることが可能となる。

本来、Activity4は2人の子供が対戦型で行うものであるが、現時点のe-Learning化では、エラーの発生をコンピュータに行わせることによって、1人でも実施できるようにしている。

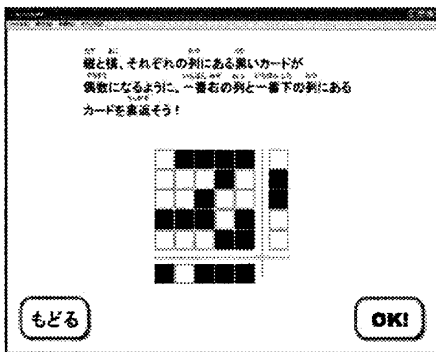


図 2.3(a) Activity4 (パリティ作成)

(4) Activity6

Activity6の実装画面を図2.4に示す。Activity6では探索アルゴリズムの学習を行う。対象は数の比較など論理的推論が可能である9歳以上の子供である。

このActivityでは、探索アルゴリズムの中でも、線形探索、二分探索、ハッシュ法の3つのアルゴリズムを学習することが可能である。図2.4(a)は線形探索と二分探索の実装画面で、図2.4(b)、(c)はハッシュ法の学習の際に使用する画面である。船の名前を一つ選び、その船の番号と相手の船の番号が一致すると勝ちとなる。

この学習で重要になってくるのは船の並びである。線形探索は船の並びをランダムとし、二分探索は割り振られた番号が小さい順に船を並べる。そして、ハッシュ法では、船の並びは0から9の列によって管理され、船の番号の各桁を足し合わせた最後の桁の数字ごとに、船を並べる。

学習の前には、探索アルゴリズムによって船の順序に違いがあることを子供たちに明示する必要がある。

この学習を通して、コンピュータが多くの情報の中から、目的の情報を探索する際、どのような手順で探索を行っているか知ることができる。船がランダムに並んだ線形探索では一つ一つ探索を行うしかなかったが、二分探索のように船が整列されていると、船の数を二分するように攻撃する船を選択していくと一番効率よく相手の船を探索できることが分かる。ハッシュ法については、ハッシュキーから相手の船がある場所をすぐに特定することができ、ハッシュキーの中に船が少ないほど探索が速くなることも分かる。

このActivityを通して、以上のことを実体験させ、コンピュータにおける探索手法を理解させる。

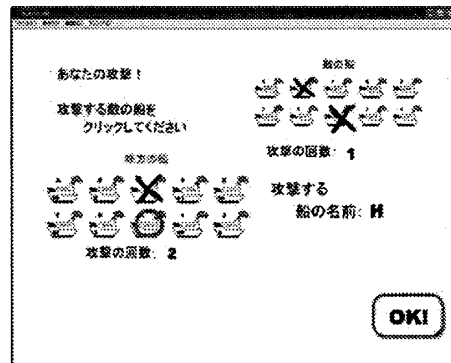


図 2.4(a) Activity6 (線形探索・二分探索)

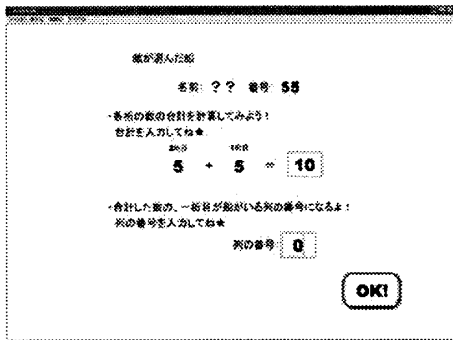


図 2.4(b) Activity6 (ハッシュキーの算出)

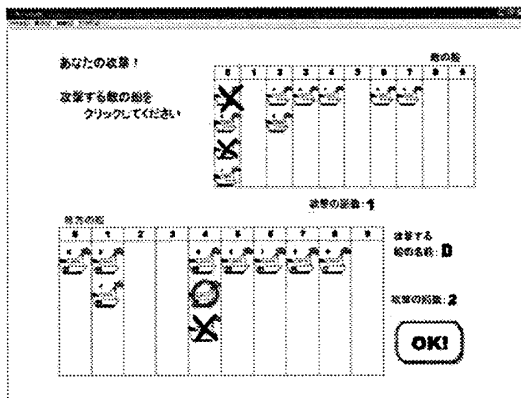


図 2.4(c) Activity6 (ハッシュ法)

る詩なども日本語のものである方が自然である。しかし、日本語の場合、その文字種の多さに起因して、英語の場合と比較して、同じ文字列が繰り返し現れる頻度が低くなると思われる。

従って、この Activity の実装に際しては、例えば国語の教科書などから繰り返し文字列の多い日本語文を抽出することによって、直接的にコンピュータ化する方法が考えられる。こうすることにより、圧縮効果を分かり易くすることができる。

または、幅広い日本語への適用を考慮して、別種の圧縮方式に基づいた日本語用 Activity を新規に考案する方法も考えられる。

現在、いずれの方策を採るべきか検討中であり、結論には至っていない。

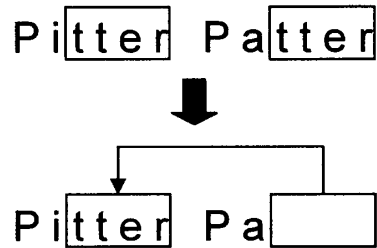


図 3 圧縮作業の例

5.2 考察

以上の実装では、Computer Science Unplugged の指導書の内容をほぼ直接的にコンピュータ化することができた。コンピュータ化することによって、何らかの新機能の追加（一部作業の自動化なども含む）なども可能と思われるが、現時点ではそこまで踏み込んでいない。

一部作業の自動化等によって Computer Science Unplugged 本来のフレーバが失われてしまう危険性もあると考える。従って、それらの実現は子供たちによる実使用とそのフィードバックを見ながら、慎重に行うべきと考える。

6. 言語依存の Activity

Activity3 を例にとって、言語依存の Activity の問題点を概説する。

Activity3 ではテキスト圧縮について学習する。その方式は一般に「辞書式圧縮法」と呼ばれる圧縮方式である。具体的には、まず詩などの文章を用意する。子供たちは、その文章に繰り返し出てくる 2 文字以上の文字の並び、または単語のパターンを探し、省略が可能となることを知る。このような作業と、圧縮された文章の復元作業を繰り返すことで、テキスト圧縮の基本原則を学習することができる。

圧縮作業の例を図 3 に示す。ここでは、「tter」という文字列が 2 箇所に見れている。その後半の「tter」部分を前半の「tter」部分へのポインタで置き換えることによって、テキストデータの圧縮が行われている。

この Activity を日本の子供たちに適用する場合、用意す

7. まとめ

本稿では、子供を対象としたコンピュータ科学教育法の e-Learning 化について提案した。現在、Activity1, 2, 4, 6 の基本問題について実装が終了した。

今後の計画としては、残りの Activity の実現方法について検討後、基本問題について実装を行い e-Learning 化を完了する。その後は、応用問題の実装を行い充実したコンテンツを目指す。

さらに、従来型の Computer Science Unplugged との比較を行い、子供たちにとって使いやすい教材を提供するため、実際に小中高生を対象とした実験授業を行う。その際、インタフェースの改善点や e-Learning 化の問題点などを調査し、更なる改良を行う計画である。

参考文献

- 1) COMPUTER SCIENCE unplugged
<http://csunplugged.com/index.php/activities.html>
- 2) コンピュータを使わない情報科学教育—Computer Science Unplugged—の翻訳と実施
<http://kanemune.eplang.jp/data/sss07unplugged.pdf>
- 3) Tim Bell, Ian H. Witten and Mike Fellows, コンピュータを使わない情報教育アンプラグドコンピュータサイエンス, 兼宗 進, 株式会社 イーテキスト研究所, 東京都, 2007 年.
- 4) ActionScript 逆引きクイックリファレンス, 田中 康博, 林 拓也, 古旗 一浩, 株式会社 毎日コミュニケーションズ, 東京都, 2005 年.