

大学の授業におけるコンピュータ・アンプラグドの実験 —プログラミングの授業での試み—

Computer Science Unplugged Experiment in the Class of Dokkyo University -Case Study of the Programming Class-

黄 海湘*¹、立田 ルミ*¹
Haixiang Huamng、Lumi Tatsuta

Email: h.haixiang@gmail.com

キーワード：アンプラグドコンピューティング、ソート、プログラミング
Keywords: unplugged computing, sorting, programming

本稿では、アンプラグドコンピューティングの例のうち整列アルゴリズムを取り上げ、実験した結果について述べる。獨協大学国際教養学部言語文化学科では情報科目の1にプログラミングの授業が設置されている。この授業では、整列アルゴリズムを説明する前に、コンピュータを使わないで学生たちにどのようなアルゴリズムが一番効率の良い整列のアルゴリズムかを考えさせた。授業実践では、グループごとにアルゴリズムを考えさせ、それらのアルゴリズムをグループごとに発表させ、それぞれのグループの比較回数を検討させた。これらの実践により、整列アルゴリズムが今までよりもよく理解され、その結果、プログラミングも容易になった。本論文では、その方法と結果について述べる。

In this paper, before explaining algorithm in the computer programming class which is one of information subjects in Dokkyo University, students were not using computer. And made students think what kind of way is the best way. We announced those ways every group and made them sort algorithm among the example of an unplugged computing is taken up by writing and the result on which we experimented is described. The class of the programming is installed in one of an information subject in Department of Interdisciplinary Studies, Faculty of International Liberal arts department of languages and cultures, Dokkyo University. Before explaining a sort algorithm at this session, a computer wasn't used, and made students think what kind of algorithm is the most efficient sort. We made them consider algorithm every group by class practice, and made them announce those algorithm every group, and made them consider the comparative number of times of the respective groups. A line algorithm was understood by these practice better even than now, and as a result, a programming also became easy. We describe the way and result. Algorithm of sorting was understood by these experiments better even than before, and a programming also became easy. We discuss the way and result in this paper.

*1: 獨協大学 情報学研究所

1. はじめに

「コンピュータを使わない情報教育」と呼ばれる新しい情報教育の手法である「アンプラグドコンピュータサイエンス」は、ニュージーランドカンタベリー大学の Tim Bell 氏らが始めたプロジェクトである。

この方法を使うと、コンピュータ科学のさまざまな概念を、小学生の年齢から楽しみながら学ぶことが可能である。

韓国の済州島で行われた情報処理学会情報教育シンポジウム(SSS2008)では、Tim Bell 氏の講演があった。

Tim Bell 氏は SSS2008 の講演では、10 年ほど前に自分の 5 歳の子供に情報科学の楽しさを教えたいと考え、それ以来、コンピュータを使わないコンピュータ科学の教育方法と具体的な教材を開発してきたとお話があった。⁽¹⁾

Tim Bell らは、1998 年に 20 章からなるテキストを出版した。1998 年に出版された本は各国語に翻訳され、日本でも 2006 年に出版されている。⁽²⁾そして、このプロジェクトは現在でも継続しており、Web 上で 12 章からなるテキストと活用事例を提供し続けている。⁽³⁾

日本語の本が出版されたことにより、いくつかの大学での事例が発表されている。⁽⁴⁾

本稿では、獨協大学国際教養学部言語文化学科学学生対象に行われた、プログラミングの授業内でのアンプラグドコンピュータサイエンスの試みについて述べる。

2. アンプラグドコンピュータサイエンス

ここでは、アンプラグドコンピュータサイエンスについて述べる。

2.1 アンプラグドコンピュータサイエンスの例

Tim Bell 氏は前述のように、コンピュータを使わないコンピュータサイエンスの授業例として、いくつかの例を取り上げている。そして、授業例の中では、その概要と教科学習との関連、技能、対象年齢、教材について詳細に述べている。そして教材として、解説シートとワークシートを用意している。その上、各問題に対する解答と解法のヒントが書かれており、非常に細やかな教材となっている。

本稿ではその中で、前述の日本語に翻訳された書籍の例について述べる。

(1) 2 進数

ここでは、コンピュータの中に情報をどのように入力するかという問題と、データと情報の違い、数値・文字・単語・画像はどのように 0 と 1 に変換されるかの問題を含んでいる。問題としては、①2 進数を計算する、②秘密のメッセージを送る、③電子メールとモデム、④31 より大きい数を数える、⑤2 進数のあれこれ、となっている。

また、実際のコンピュータではどのような処理が行われているか（ハードウェアの仕組み）に言及している。

(2) 画像表現

ここでは、コンピュータで絵や写真がどのように記録されるかの問題と、図形の問題を扱っている。問題としては、①色で数を表す、②子どもファクシミリ、

③自分の絵を描こう、④自分の絵を描こう（応用問題）、となっている。

また、実際のコンピュータでは、ファクシミリと同じ処理をしていることが説明されている。

(3) テキスト圧縮

ここでは、繰り返して使われる文字をどのように圧縮するかについて説明されている。問題としては、①それ、さっきも言った（文字のパターンを抽出）、②元の単語をどれだけ減らせるか、③単語はいくつ必要か、④文章を圧縮する、となっている。

また、実際のコンピュータで行われている圧縮方法(LZ 符号)について、説明されている。

(4) エラー検出とエラー訂正

ここでは、データをやりとりするときにデータの破損（エラー）を検出して訂正する方法についての問題を扱っている。問題としては、①カード交換の手品（パリティチェック）、②ISBN コード（エラーチェックコード）、③エラーを探す、となっている。

また、実際のコンピュータでは、複数のエラーを検出して修復するための複雑なエラー制御システムがあり、これらはハードディスクでディスクの一部が壊れても、他の領域を割り当てようになっていることを説明している。

(5) 情報理論

ここでは、情報量の測り方についての問題を扱っている。問題としては、①20 の扉、②決定木、がある。

実際のコンピュータでは、有名なクロード・シャノンが、情報量をビットで測定しており、メッセージの情報量をエントロピー（事象の数）と呼んでいることを説明している。

(6) 探索とアルゴリズム

ここでは、大量のデータから情報を探すアルゴリズムについての問題を扱っている。問題としては、①戦艦導入ゲーム、②戦艦線形探索ゲーム、③戦艦二分探索ゲーム、④戦艦ハッシュ法を使ったゲーム、がある。

実際のコンピュータでは、多くの情報の中から素早く情報を検索する必要があることを説明している。そしてアルゴリズムとして、線形探索法、二分探索法、ハッシュ法があり、実際にはハッシュ法がよく使われていることを説明している。問題としては、①一番軽いつと一番重い、②重りの整列、③分けて作業する（クイックソート）がある。

また、アルゴリズムとして、挿入ソート、バブルソート、マージソートの説明がある。実際のコンピュータでは、高速にソートできるクイックソートがよく使われていることを説明している。

(7) 整列アルゴリズム 1

コンピュータでデータを処理する場合、名前を 50 音順に並べ替えたり、学籍番号順に並び替えたり、得点の高い順に並べ替えたりしなければならぬが、この整列アルゴリズムについての問題を扱っている。

(8) 整列アルゴリズム 2

ここでは、コンピュータの処理速度を上げるため、複数のコンピュータを使って処理する問題を扱っている。

問題としては、①並び替えネットワーク、がある。実際のコンピュータでは、並列計算が行われていることと、並列計算を行えば処理速度が必ずしも処理速度が上がる訳ではないことも説明している。

(9) 最小全域木

現代ではコンピュータとコンピュータを結び付けてネットワークを構築しているが、この構築に際して、どのようなネットワークを構築すればよいかという問題を扱っている。問題としては、①マッディ市プロジェクト、がある。ここでは、都市と道路をグラフとして表現し、最適化の問題を取り扱っている。

(10) ネットワークにおけるルーティングとデッドロック

インターネットを介して情報を送る場合、あるサーバーから送り先のサーバーまでの経路をルーティングと呼んでいる。このルーティングは 1 通りではなくいくつかのルートを選択して情報を送るが、多数の人がデータを送る場合、どのルートも混雑していることがある。丁度、高速道路の渋滞のような状況である。これを輻輳と呼んでいる。この状態が続くと、ネットワークが機能しなくなり、デッドロック状態になる。この問題として、ミカンゲームがある。これは、ゲームに参加した人が協力して行う、問題解決のゲームとなっている。

実際のコンピュータでは、ルーティングは経路制御と呼ばれ、情報をどのような経路で送るかを制御する。

(11) 有限オートマトン

コンピュータのプログラムでは記号を処理するが、この処理を行うための有限オートマトンの考え方を学ぶ問題である。問題としては、①宝島を探す、②宝島の財宝を探す、③謎のコインゲーム、がある。

実際のコンピュータでは、文字などの入力列を処理するために、有限オートマトンを使って機械語に翻訳している。

(12) プログラミング言語

現在、プログラミング言語として、C、C++、Java、など様々な言語あり、それらの言語を機械語あるいは中間言語に翻訳するためのコンパイラが開発されている。問題としては、①出発進行、という問題が 1 問だけとなっている。

実際のコンピュータでは、プログラムで指示されたことだけを行う。そのため、プログラミングが非常に重要となる。

アンプラグドコンピュータサイエンスの例では、12 のセクションに分けて問題を提起しているが、どの問題もコンピュータサイエンスの基本となるものであり、筆者も学生時代に学んだ内容である。これを分かり易く解説し、ワークシートを使って理解しやすいようにしているところに教材としての価値がある。

3. プログラミングの関連科目への適用

ここでは、筆者の一人が担当しているプログラミングの関連科目である「データ構造とアルゴリズム論」への適用について述べる。

3.1 授業背景

大学でのコンピュータプログラミングに関する授業は、理工系の学部には所属する学生が学ぶ科目であるというイメージがあるが、コンピュータの普及と一般社会で高まる情報処理知識への要求により、文系大学においてもプログラミングに関連する授業がなされるのが当然であると考えられるようになった。

獨協大学においても、建学当初から早い段階で学生に情報教育を行う重要性を認識し、経済学部において、プログラミングに関連する授業を開講した。その後、国際教養学部言語文化学科が設立されて、自然言語処理の観点から一般の情報リテラシー教育以外に、プログラミングに関連する授業を行うようになった。

しかし、現状では、文系の学生は、理工系の学生と比べると、数学の基礎力が低く、プログラミングに必要なロジカルシンキング能力が欠けている。そもそも、文系学生にプログラミングの授業を行う目標の一つは、問題を与えられたときに、その解決の手順を考え、論理的思考を育成することである。

従って、著者の一人が担当しているプログラミングに関連する授業では、受講する学生の状況を考え、コンピュータを使ったプログラミング言語に特化した内容ではなく、アンプラグドコンピュータサイエンスの試みを行った。

図 1 は担当する授業のシラバスを示している。図 1 から分かるように、授業内容の組み立てはアルゴリズムの基本を中心としている。

今回の受講生は 18 名で、受講生全員がコンピュータ入門の授業を履修してはいるが、プログラミングの経験がない。以下、我々が第 1 回目の授業で採った授業方法と結果を例として報告する。今回の授業内容は前述した Tim Bell のアンプラグドコンピュータサイエンスの例で示した (6)、(7)、(8) の内容の一部である。

3.2 授業方法

第 1 回目の授業内容は「講義の概要、基礎知識、目標」である。学生全員のコンピュータスキルの状況を確認したところ、大多数の学生たちはパソコンを利用してはいるが、コンピュータの構造やアルゴリズムなどを勉強したことがないことが分かった。

そこで最初に、コンピュータは主に「CPU」、「メモリ」、「記憶装置」で構成されていることを説明し、CPU は人間の頭脳、メモリは仕事用の机、記憶装置は本棚に相当すると例え、一定な手順 (アルゴリズム) に従って仕事することを解説した。

次に、説明した内容をうまく理解させるのと同時に、勉強意欲を高めるために、学生全員を 2 グループに分け、以下のようなゲームを順番に行い、勝敗を競わせた。

(1) ゲーム 1 :

1. 日本語の五十音を書いた 46 枚の紙カードを 2 セット用意し、各グループに配る。
2. グループ 1 に大きい机、グループ 2 に小さい机を指定する。
3. グループ毎にカードを広げる。

(春) ◯	自然・環境・人間2 (データ構造とアルゴリズム論) ◯	担当者 ◯	黄 海湘 ◯
講義目的、講義概要 ◯		授業計画 ◯	
<p>【目的】 データ構造とアルゴリズムは難しそうなイメージであるが、実は日常生活と仕事の中でよく利用する身近なものである。情報爆発といわれる現代社会において、いかにデータを構造化し、必要なアルゴリズムで素早く処理する能力は不可欠である。本講義は実習を通してデータ構造とアルゴリズムに対する理解を深めることを目的とする。</p> <p>【概要】 データ構造とアルゴリズムに関する基本的な理論と方法について、講義形式とパソコンを使った実習形式で体験する。講義内容は、文系の学生でも理解できるように、ゲームを取り入れ、体験しながら学ぶ。</p> <p>【受講者への要望】 講義と実習を織り交ぜて授業を進めるため、休まず出席すること。</p>		<ol style="list-style-type: none"> 1. 講義の概要、基礎知識、目標 2. リスト 3. スタックとキュー 4. 再帰 5. 計算量解析 6. 解析木 7. 二分探索木 8. ソート 1 9. ソート 2 10. 二分探索 11. 平衡木 12. ハッシュ 13. グラフ 14. 動的計画法 15. 授業のまとめ 	

図1 講義のシラバス (一部分)

4. 教員が探したいカードを示し、早く見つけるグループが勝ちとなる。
- (2) ゲーム 2 :
1. ゲーム 1 と同じように、カードを各グループに配る。
 2. 各グループから代表者一人を選び、カードを手持ちにする。
 3. 教員が探したいカードを示し、早く見つけた代表者のグループが勝ちとなる。
- (3) ゲーム 3 :
1. 各グループに6つの箱を渡す。
 2. それぞれ箱を一行に並べる。
 3. 数字が書いてある六つのカードを2セット用意し、各組の箱に同じ順番で入れる。
 4. 箱中のカードを数字の昇順に従って並べ替え、手順数を数える。(手順の数え方: 比較 1 回、交換 1 回)
 5. 手数が少ない組が勝ち。

3.3 授業結果

ゲーム 1 では、繰り返す回数が増えると、大きい机のグループが有利であることを学生から指摘された。そこで、コンピュータの速さはCPU 以外に、メモリの大きさに左右されることを説明した。

ゲーム 2 では、最初は勝負にあまり差がなかったが、グループ 1 に、カードの順番を五十音順に変えてよいと指示した結果、グループ 1 がよく勝つようになった。

そこで、コンピュータ中のデータを一定的なルール

に従って構築すれば、仕事の効率が良くなることを説明した。

さらに、五十音順に順番を変えた後、どうやってカードを探しているかについて説明させ、データの探索方法及び整列の重要性を説明した。

ゲーム 3 では、各グループともデータ整列の重要性が分かったので、学生たちは様々な方法を使ってカードの順番並びにチャレンジするようになった。

そこで、各自のやり方を実演させるとともにその方法を説明させ、それらの方法を比較させた。

授業の終わりでは、整列問題におけるアルゴリズムの多様性について教員が説明した。

以上のように、ゲームを通じて、学生たちに勝敗の分け目は何であるかを考えさせることによって、学生たちはコンピュータの処理の速さに影響するものは何であるかということが理解できた。また、データを早く見つけるための方法と、その処理のための整列方法などについて深く関心を持つようになった。

また、数字の整列アルゴリズムを考えることによって、様々な手法が存在していることが分かり、後の授業で整列に関するプログラムの実習に良い影響を与えた。

4. おわりに

本稿では、Tim Bell らが提唱しているアンプラグドコンピュータサイエンスの例と、実際に獨協大学国際教養学部で行ったプログラミングの授業におけるアンプラグドコンピュータサイエンスの例について述べた。

今後は、別のクラスでアンプラグドコンピュータサイエンスの方法を利用して、アンプラグドコンピュー

タサイエンスを使ったクラスと使わないクラスの理解度の差を求めることが出来ればよいと考えている。さらに、アンプラグドコンピュータサイエンスの例として新しいものが提案できればよいと考えている。

謝辞

本研究の一部は、情報学研究所研究助成によるものである。

参考文献

- (1)Tim Bell: “Introducing the next generation of Computer Scientists to the magic and beauty of computer” 情報処理学会、情報教育シンポジウム論文集、Vol.2008, No.6, pp1-5(2008)
- (2) Tim Bell/Ian H.Witten/Mike Fellows、 Matt Powell、 兼宗進訳: 『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』、イーテキスト研究所(2015)
- (3)CS Unplugged のページ
<http://csunplugged.org/>(2015.10.5 現在)
- (4) 和田勉、“アンプラグドコンピュータサイエンスと板書講義を併用した大学でのアルゴリズムの授業”, 情報処理学会、情報教育シンポジウム論文集、Vol.2014, No.6, pp.32-38, (2014)

(2015年9月29日受付)

(2015年12月2日採録)