

コンピューティング先史時代に関する教育

中森眞理雄

東京農工大学 工学部 情報工学講座

1. はじめに

今日のコンピュータが出現する以前に，“コンピュータには何ができて何ができないか”が精力的に研究された時期があった。前世紀末から今世紀前半にかけての約60年がその時期にあたる。この時期をコンピューティング先史時代とよぶことにする。

本シンポジウム“コンピューティングの歴史”の目的は、コンピューティングの歴史における必然と偶然を見分け、コンピュータサイエンスの新たな発展の基礎とすることであり、コンピューティング先史時代に先人が築いた業績の正当な評かも本シンポジウムの目的と合致する。

大学等における教育では、今日のコンピュータ技術を教授するだけでなく、コンピューティングの歴史における必然性の洞察力を涵養することが重要である。先人のしたことを追体験させることは、今後のコンピュータサイエンスの発展を担う若い人々の基礎的素養として不可欠であろう。筆者の属する学科において、(事実上)世界初のプログラム内蔵方式のコンピュータEDSACの機械語プログラミングの実習を課していること[1]、手回し計算機による数値計算実習を課していること[2]等々は、これらの観点から実施されているものである。

本稿では、手作業による整列(ソート)の教育(演習と講義)をとりあげ、筆者の教育体験を報告したい。

2. 整列に関する演習と講義

2.1 手作業による整列の演習

これは、データ(数値や文字列)が記載されたカードをキーにしたがって手作業で昇順や降順に並べるものである。この演習は、1978年～1980年に実施され(筆者が担当)、その後1988年から現在まで(一時中断したが)実施されている([2])。

この演習では、種々の整列アルゴリズムの概略を学生に示し、アルゴリズムの違いによる整列時間の違いを比較させる。アルゴリズムとしては、基数法(radix sort)、マージソート(merge sort)、クイックソート(quick sort)、選択ソート(selection sort)、を用いている。

(C) 中森眞理雄

カードの枚数 n を変えて、アルゴリズムごとに、 n と整列時間の関係をグラフに描かせている。

手作業を演習に取り入れる場合、手作業とプログラムとの対応をどの程度厳密にとるかが問題となる。例えば、作業場所とコンピュータのメモリとの関連づけを重視する立場からは、カードを置いたり積み上げたりする場所は予め指定した場所に限るべきであるという考え方が出てくる。一方、手作業とプログラムとは本質的に異なるものであり、両者の厳密な関連づけにこだわる必要はないとする考え方も有り得る。後者の考え方を推し進めるならば、手作業によるソートの演習では、それに即したモデル化や手間の評価方法を考えることが重要であり、それによってコンピュータプログラムとの間に乖離が生じても止むを得ないという考え方に到達する。筆者の学科では、後者の考え方を採用している。

当初は、筆者は、手作業による整列の時間をまとめる際には、両対数方眼紙を用い、横軸に枚数、縦軸に時間をとるように指導していた。しかし、これはあまり効果がなかった。その理由は、手作業で整列できるカード枚数は、限られた演習時間の中では、100枚が限界であり、 $n = 10$ から $n = 100$ の範囲とそれに対応する時間を両対数方眼紙に描いても通常の方眼紙に描いても、顕著な違いはないからである。

2.2 整列アルゴリズムに関する講義

手作業による整列演習は2年次学生に対して実施している。3年次学生には、アルゴリズム論の講義を筆者が担当している。この講義において、整列を取り上げている。時間数は1回(90分)である。整列を専門的に研究するのではない限り、アルゴリズム論における整列の扱いは1学期(15週)中1週で十分である。

講義は、配列中のデータの探索から始める。線形探索のアルゴリズムを示し、次に“番兵(sentinel)”つき線形探索のアルゴリズムを示す。整列アルゴリズムの手間の違いを比較させ、わずかな工夫が手間の違いにつながることを観察させる。ここで、アルゴリズム論とは、このような工夫を体系化・理論化したものであると述べる。

次に、もしデータが整列されていれば、二分探索が可能であることを指摘する。二分探索の手間を線形探索や番兵つき線形探索の手間と比較させる。

ここから整列アルゴリズムの説明に入る。まず、初歩的な選択ソートを説明し、手間の評価を示す。バブルソートも本質的に同じことをしていることを観察させる。さらに、マージソートとクイックソートを提示する。前者においては、マージの度に連の数が半分になるのでパスの回数が $\log n$ 回であることを示す。後者においては、再帰の考え方を示す(ただし、再帰は別の講義や演習でも既に取り上げられている)。

時間的余裕があれば、二分木の構造を示し、ヒープソート(heap sort)にも触れる。

以上の考察の上に、ソートの手間の下界が $O(n \log n)$ であることを理論的に証明する。

最後に、基数法 (radix sort) を説明する。ここで、学生は初めて整列が分類であったことを知る。筆者は、このとき、80桁のカードを学生に“本日の出席記念”として配布している。今日の学生にとって、カードはたいへん珍しいのである。

以上に述べた探索と整列のアルゴリズムの教育は、ACMのコンピュータサイエンスのカリキュラムに関する1988年の報告書[3]でも採用されている。

3. 整列アルゴリズムの歴史

3年生の対するアルゴリズム論の講義で基数法に触れた後に、整列アルゴリズムの歴史を述べている。講義では[4]を参考にしている。

[4]によれば、米国の1880年の国勢調査の集計にたいへんな苦勞をし、この苦勞を解消する目的で、H. Hollerithが作表機 (tabulating machine) を発明した。その機会は1890年の調査では約100台が使われ、大成功を収めたとされている。

そこで用いられたカードの大きさは、 $6\frac{5}{8}'' \times 3\frac{1}{4}''$ であった。

Hollerithの分類機において、分類箱は26個あり、13個ずつ2列ならんでいたらしい(写真による)。操作卓の読み取り機にカードを入れて“プレス”すると、多数の針が下りて、カードに穴の開いている場所の針だけが下の水銀槽に達し、電流が流れ、しかるべき分類箱の蓋が開く。同時に、対応するダイヤルが1進む。オペレータはカードを読み取り装置から取り出し、蓋が開いた分類箱にそのカードを入れ、蓋を閉じる。ダイヤルは40個あり、写真で見る限り、針が2本あったようである。ダイヤルに数値が示されることを以て“作表”と称したらしい。これが“作表機”の意味である。熟練したオペレータは、 $6\frac{1}{2}$ 時間で19071枚のカードを処理したとの記録があるそうである。平均すると、1分あたり49枚のカードを処理したことになる。大半のオペレータの処理時間はこの3分の1だったらしい。

Hollerithは、1901年から1904年にかけて、この作表機を改良している。改良点の一つは、カードの送りを自動化することであった。Hollerithはこの作表機に関して特許を申請している。特許の内容は、“2桁の数値項目を桁ごとに分類する”というものである。ただし、Hollerithは1の桁と10の桁のどちらを先に分類するかについては触れていない。もし、10の桁を先に分類するのであったのなら、Hollerithが考えていたアルゴリズムは今日の基数法とは異なるものとなるが、一般には、彼は基数法のための機械を発明したと見なされているようである。

[4]によれば、1の桁についての分類を最初にせよと明記されている最初の文献はIBM分類機マニュアル(1936)である。もっとも、右から左へという一般的テクニックは、それより古く、1930年にL. J. Comrieが*Trans. of the Office Machinery User's Assoc.* (London, 1930), pp.25-37で言及しているらしい。このComrieこそは、統計用・会計用に作られた作表機が科学技術計算に使えることを喝破した最初の人物であろう。Comrieによれば、当

時の英国では毎分 360~400 枚の処理能力のある分類機のレンタル代は月 9 ポンドだったとのことである。

[4]によれば、最初に作られたコンピュータプログラムは整列プログラムであった。1945 年には J. von Neumann がマージソート(これは内部ソートである)のプログラムを書いている。

以上の話は、学生にとって、たいへん興味のある話のようで、筆者はアルゴリズム論で整列を取り上げるときは必ず触れることにしている。

4. おわりに

コンピューティング先史時代の教育と題しながら、現実には、整列アルゴリズムの教育に限定した報告になってしまった。今後は、今世紀前半までに行われていたさまざまな事務作業を調べ、今日のコンピュータ技術に継承されているものを明らかにしたい(例えば、事務処理言語 COBOL への流れ等)。整列アルゴリズムの演習に関しては、東京農工大学野瀬隆助手の努力に負うところが大きい。

参考文献

- [1] 清水, 阿刀田, 高橋, “情報工学系学科の計算機初期教育における EDSAC の活用の試みと効果”, 情報処理学会論文誌, 24, 3, pp.281-292 (1983).
- [2] 野瀬, 中森, “手作業とプログラミングによる整列の演習”, 情報専門学科のコアカリキュラムシンポジウム, pp.1-8, 1991.
- [3] P. J. Denning, D. E. Comer, D. Gries, M. C. Mulder, A. Tucker, A. J. Turner, and P. R. Young, *Report of the ACM Task Force on the Core of Computer Science*, ACM, 1988.
- [4] D. E. Knuth, *The Art of Computer Programming*, Vol.3, Addison-Wesley, Reading, Mass., 1973.