

ソート (その3)

山本昌志*

2005 年 10 月 25 日

1 本日の学習内容

本日は，以下の学習を行う．

- クイックソート (Quick Sort)
- マージソート (Merge Sort)
- コームソート (Comb Sort)

2 クイックソート

原理やプログラムのテクニックについては前回のプリントで示しているので，ここには記述しない．ただし，前回のプリントで分かりにくかったクイックソートの様子を示すことにする．

*独立行政法人 秋田工業高等専門学校 電気情報工学科

2.1 クイックソートの様子

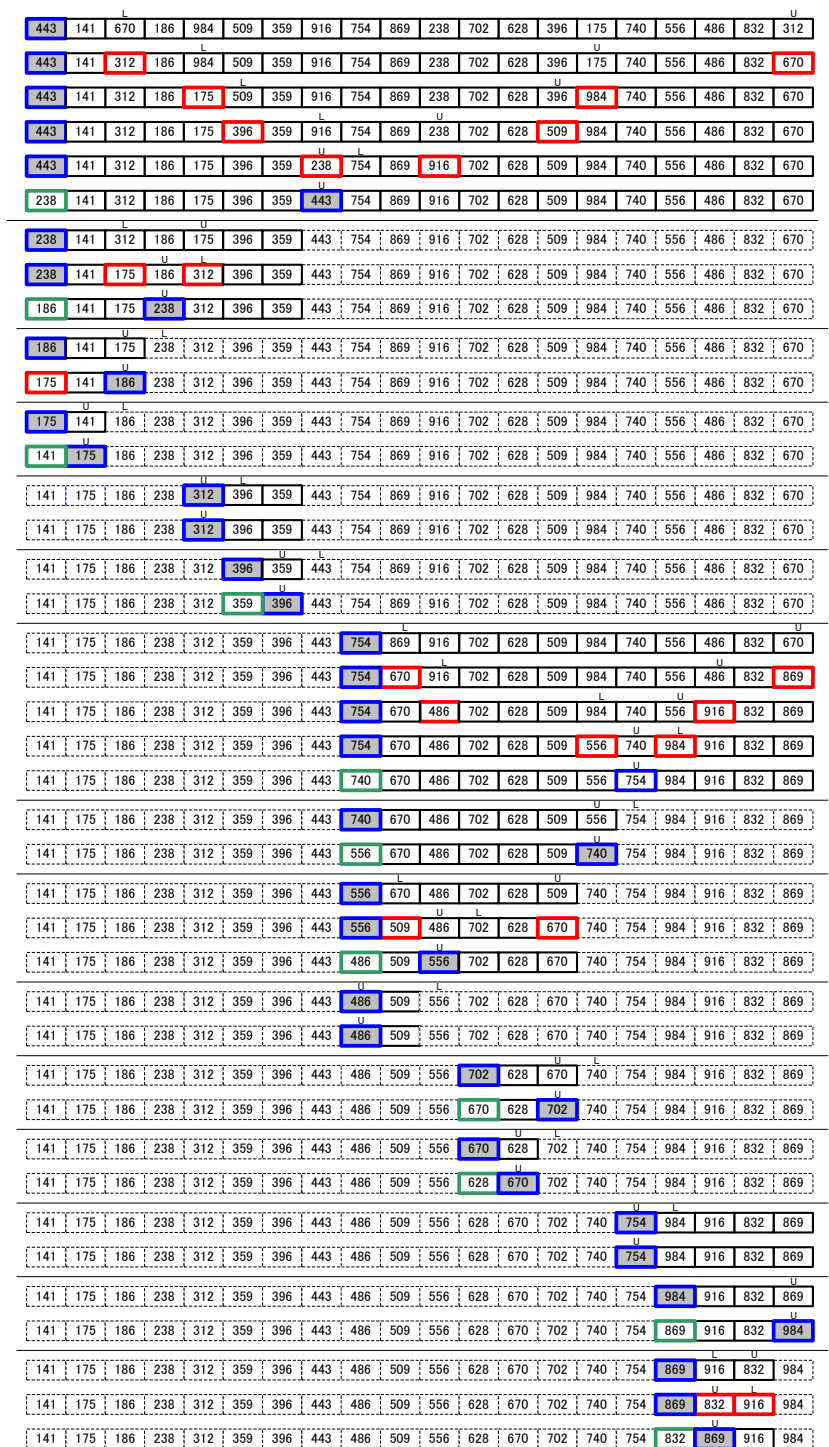


図 1: クイックソートの様子 .

3 C 言語の qsort() 関数

C 言語には qsort() が実装されており、それはクイックソートが使われていることが多い。そのため、実際のプログラムでは、プログラマーがソートのルーチンを記述することは希で、この qsort() 関数をつかう。お手軽なので、諸君はこれを使うのが良いだろう。

この関数は、標準ライブラリー関数で、stdlib.h をインクルードすることにより使うことができる。プログラムの先頭付近に

```
#include <stdlib.h>
```

と書けば、インクルードできる。

この qsort() 関数を使う場合の引数の渡し方であるが、それは教科書 (Table 1-1) に書いてあるとおりである。正確な記述は教科書の通りであるが、もう少し分かりやすく書くと

```
void qsort(配列名, 配列の数, 配列一つのバイト数, 比較関数)
```

となる。返値はなく、配列そのものがソートされる。引数はデータを入れてある配列に関するものと比較関数である。

ここで、ちょっと難しいのは、比較関数の作り方である。それを、教科書の例 (List 1-4) で示すことにする。教科書の比較関数は、

```
int compare(const void *arg1, const void *arg2)
{
    return (*((int *)arg1) - (*((int *)arg2)));
}
```

となっている。返値は整数型と宣言されている。qsort() 関数の引数で渡す比較関数の返値は整数と決められているので、それに従っている。問題は引数で、const は引数が増えないことを明示的に示している。arg2 を左辺値として、代入文を使うとエラーが発生する。つぎの void は、arg1 や arg2 はポインターであるが、その型は後で決めるということである。実際のこのプログラムでは、強制型変換 (キャスト) を使って決めている。

つぎにこの関数の計算であるが、次の文が難しい。

```
*((int *)arg1
```

まずこれは、(int *)arg1 で arg1 を整数型のポインターに強制的に型変換している。そして、もっとも左にあるアスタリスク*で、そのポインターが示す値を取り出している。

ちょっと難しいが、C 言語では、引数として変数のみならず関数も引数で送ることができるのである。

4 マージソート

原理やプログラムのテクニックについては前回のプリントで示しているので、ここには記述しない。ただし、前回のプリントで分かりにくかったマージソートの様子を示すことにする。

4.1 マージソートの様子

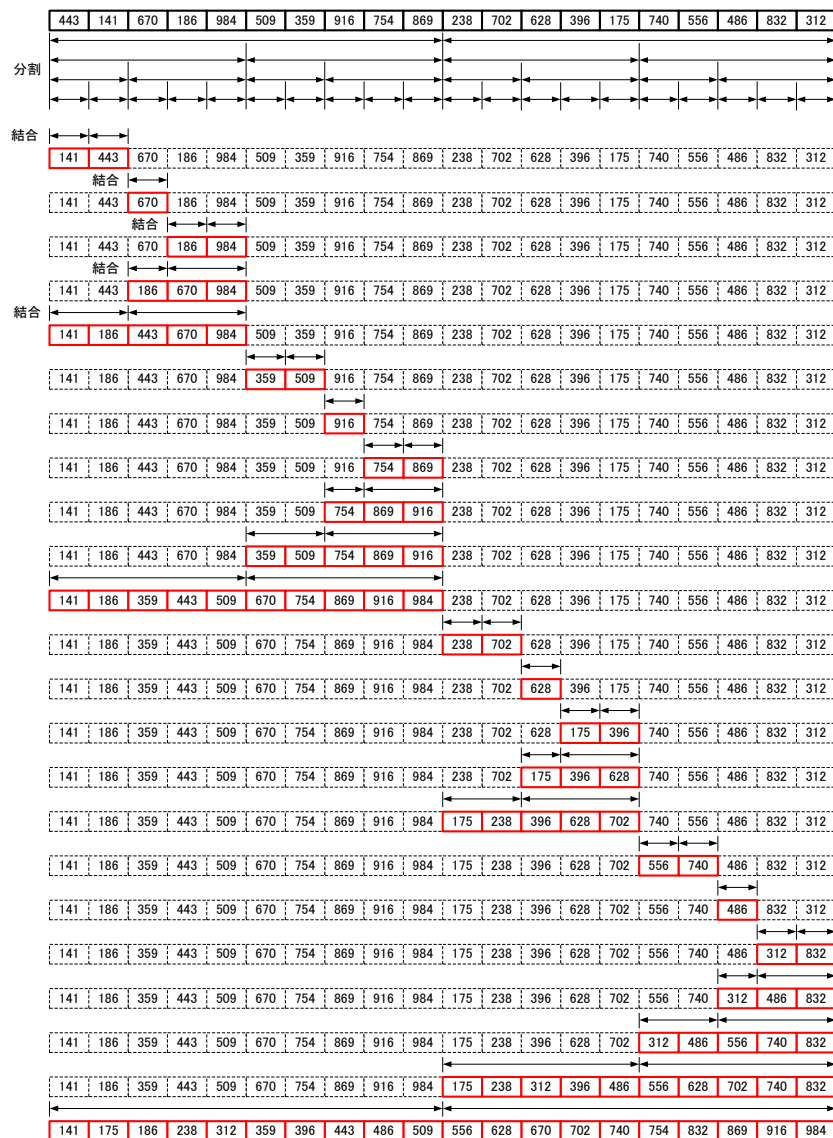


図 2: マージソートの様子 .

5 コームソート

5.1 方法

バブルソートに似た方法であるが，比較する対象が異なる．

- バブルソートは，いつでもとなり同士と比較し，大小関係が異なれば交換する．

- コームソートは、はじめは遠くのものと比較し、だんだん比較の間隔を狭くする。最後には、バブルソートと同じようになり同士と比較をする。

バブルソートの最大の欠点は、ソートが遅いことである。小さな値は左に一つずつしか、移動できないためである。なぜならば、いつもとなり同士を比較するため、交換はとなりとしかできないからである。

この問題の解決を図るために、最初大きな間隔 (ギャップ) で比較して、大きなステップで移動させるのがコームソートである。そして、だんだん間隔を狭くして、ソートを完了させるのである。

5.2 プログラムテクニック

グローバル関数 教科書の List 1-6 では、ソートする配列はグローバル変数となっており、どの関数からでもアクセスできるようになっている。ここで使われている `CombSort()` 関数と `main()` 関数の外側で、配列 `sort[N]` は宣言されている。このように、関数の外側で宣言された変数はグローバル変数と呼ばれ、どの関数からでもアクセスできる。

返値無し関数 `CombSort()` は、値を返さない関数である。この場合、返値の型として、`void` と宣言をする。`void` とは、“空の”という意味である。

引数無し関数 `CombSort()` は、引数のない関数である。この場合は、引数の型として、`void` と宣言する。

5.3 コームソートの様子

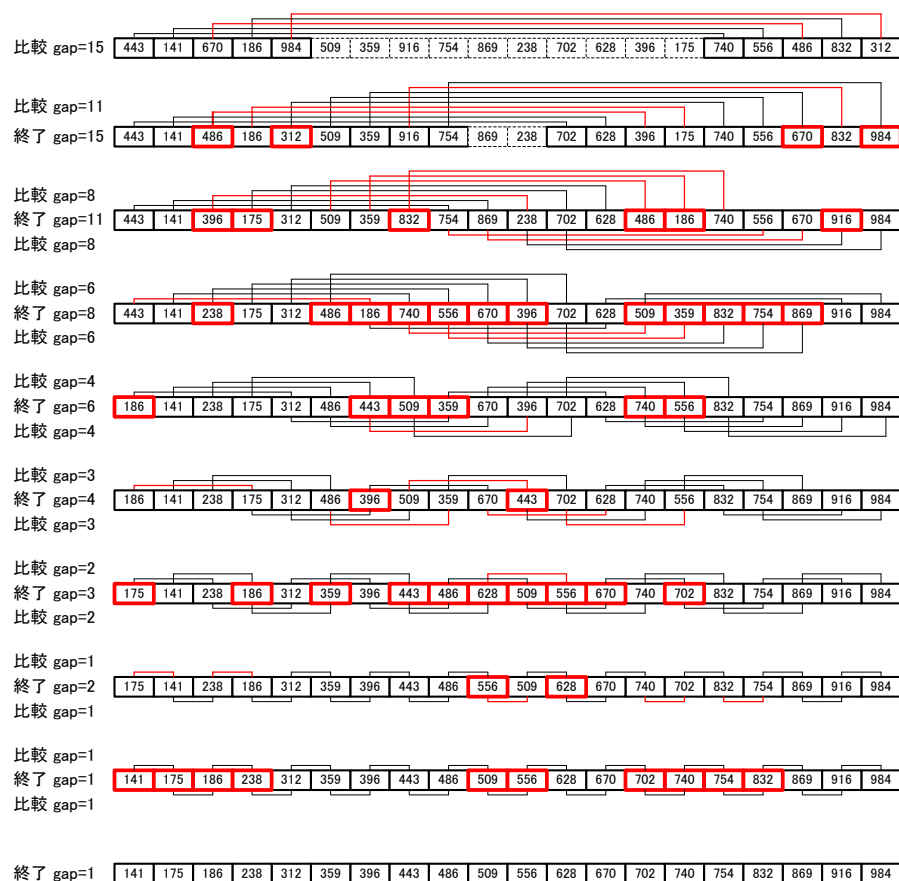


図 3: コームソートソートの様子.

6 単純挿入ソート

これは、夏休みの課題のプログラムである。

7 2分挿入ソート

教科書に沿って説明する。

8 課題

8.1 課題内容

8.1.1 ソート

以下の数列のソートに関する問題である．これをそれぞれにソートで並び替えを行った場合，その様子を
示せ．この講義ノートの「ソートの様子」で示したように記述すれば良い．ただし，様子は手書きで書く
こと．

752 778 608 239 956 244 535 840 629 353 784 199 945 847 637 413 686 172 462 223

ただし，ソートする数列は出席番号により異なる．

- 学籍番号の下1桁が1～9の場合，その数から10個の数列をソートする．たとえば，下一桁が3の場合
は，

608 239 956 244 535 840 629 353 784 199

である．

- 学籍番号の下1桁が0の場合，その10番目から10個の以下の数列をソートする．

353 784 199 945 847 637 413 686 172 462

[問1] マージソートで昇順の場合

[問2] コームソートで昇順の場合

8.1.2 プログラムの変更

教科書のマージソートのプログラム (List 1-5)，コームソートのプログラム (List 1-6) は，いずれの場合
も昇順に並び替えるようになっている．降順 (大きい順) に並び替えするようなプログラムに変更したい．以
下の問いに答えよ．

[問3] マージソートのプログラム (List 1-5) のどこをどのように変更するか?

[問4] コームソートのプログラム (List 1-6) のどこをどのように変更するか?

8.2 レポート 提出要領

提出方法は、次の通りとする。

期限 11 月 7 日 (月) AM 8:50
用紙 A4
提出場所 山本研究室の入口のポスト
表紙 表紙を 1 枚つけて、以下の項目を分かりやすく記述すること。
授業科目名「情報工学」
課題名「課題 ソート (その 3)」
2E 学籍番号 氏名
提出日
内容 2 ページ以降に問いに対する答えを分かりやすく記述すること。