



実際のコンピュータでは

順に並んでいると情報を探すのが簡単になります。もし電話帳や辞書、本の索引が読みの順に並んでいなかったら、生活はかなり不便になるでしょう。支出額のような数字の並びが整列されていれば、極端な値は最初か最後にあるので見つけるのが簡単になります。同じ値は並ぶので、重複を調べることも簡単です。

コンピュータはたくさんのデータを整列する必要があるため、コンピュータ科学者は効率のよい整列方法を見つけようとしてきました。挿入ソート、選択ソート、バブルソートなどの遅い方法が役立つこともありますが、普通はクイックソートのような高速なものが使われます。

クイックソートでは再帰の考え方が使われています。この考え方では、並びを小さな並びに分けて、それぞれに対して同じように作業を進めます。この方法は「分けて作業する（分割して統治せよ）」と呼ばれます。並びは、作業するために適した大きさになるまで、繰り返し分割されます。クイックソートでは、それぞれの並びは要素が1個になるまで分解されます！この考え方は複雑に見えますが、他の方法よりずっと高速です。



解答とヒント

- ① いちばん小さな値を見つける最もよい方法は、すべての要素を見て、それまでのいちばん小さな値を覚えておくことです。そのために、2つの値を比較して、小さなほうの値を覚えます。次に、もう1つと比較します。そしてすべての要素を調べます。
- ② 天秤で3個の重さを比べましょう。3回の比較が必要ですが、子どもが推移則に気づいたときは2回で済むことがあります。(AがBより小さくBがCより小さいなら、AはCより小さいことがわかります)

上級者向け

選択ソートで比較する回数を楽に計算する方法を紹介します。

2個の要素から小さいほうを見つけるために、1回の比較が必要です。3個では2回、4個では3回が必要です。8個の要素を考えると、最初の1個を見つけるために7回の比較を行い、次の1個のために6回の比較を行い、次に5回の比較、というように、次の比較が必要です。

$$7 + 6 + 5 + 4 + 3 + 2 + 1 = 28 \text{ 回の比較。}$$

n 個の要素を整列するためには、 $1 + 2 + 3 + 4 + \dots + n - 1$ 回の比較が必要です。これらの和は、順序を変えることで簡単に計算できます。

たとえば $1 + 2 + 3 + \dots + 20$ の和は、次のように数の組み合わせを作れます。

$$\begin{aligned} & (1 + 20) + (2 + 19) + (3 + 18) + (4 + 17) + (5 + 16) + \\ & (6 + 15) + (7 + 14) + (8 + 13) + (9 + 12) + (10 + 11) \\ & = 21 \times 10 \\ & = 210 \end{aligned}$$

一般に、和は次のように計算できます。 $1 + 2 + 3 + 4 \dots + n - 1 = n(n - 1)/2$