KUPC2020 spring I: 偶奇ソート

原案, 作成: yamunaku

N が奇数のとき

次のようにして、2 回以下の操作によって、$P_N = N-1$ となるようにする。

- P_N が $N-1$ のときは、何も行わない。
- P_N が奇数のときは、P_N と $N-1$ を入れ替える。
- P_N が偶数のときは、P_N となんらかの奇数を入れ替えて P_N を奇数にし、P_N と $N-1$ を入れ替える。

そして今後 P_N のことは考えないことにし、N を $N-1$ に置き換えて、N が偶数のときと同様の操作をする。

N が偶数のとき

方針

$S =$ `"111...1"`, $k = 0$ として、P 全体に操作を 1 回行い、左側に偶数、右側に奇数を集める。

左側に偶数、右側に奇数が集まっている状態を維持しながら、偶数同士、奇数同士の位置関係を調節する。目標は、最後の 1 回の操作で昇順に並べられるようにすることである。

例えば、$N = 10$ なら、目標となる順列は、$Q = (0, 6, 2, 8, 4, 5, 1, 7, 3, 9)$ である。なぜなら、S を次の表のように定め、k を 1 とした操作を Q に対して行えば、 Q を昇順に並べ替えることができるからである。

Q	0	6	2	8	4	5	1	7	3	9
S	0	1	0	1	0	0	1	0	1	0

$N = 12$ なら、目標となる順列は、$Q = (0, 6, 2, 8, 4, 10, 1, 7, 3, 9, 5, 11)$ である。S を次の表のように定め、k を 1 とした操作を Q に対して行えば、 Q を昇順に並べ替えることができる。

Q	0	6	2	8	4	10	1	7	3	9	5	11
S	0	1	0	1	0	1	1	0	1	0	1	0

0 以上 $N/2-1$ 以下の奇数の個数と、$N/2$ 以上 $N-1$ 以下の偶数の個数は等しいため、どのような偶数 N についても、必ずこのような順列 Q が存在する。一般には $N/2$ の偶奇で場合分けをして、目標となる順列を求めることができる。また、実装上では、昇順にならんだ順列に対して、左側の奇数を右側に、右側の偶数を左側に移動させるような操作をシミュレートして Q を求めることもできる。

両側で基数ソートを行う

P を目標となる順列 Q に並び替えるために、Q に現れる各数に対して、左側は左から 0 から $N/2-1$ までの番号をつけ、右側は右から 0 から $N/2-1$ までの番号を付ける。例えば、$N=10$ なら、次のように番号をつける。

Q	0	6	2	8	4	5	1	7	3	9
番号	0	1	2	3	4	4	3	2	1	0

順列の左側ではこの番号が昇順となるように数を並び替え、右側ではこの番号が降順となるように数を並び替えたい。そこで、左右を同時に基数ソートすることを考える。基数ソートは次のようにして行える。

i を 0 から 12 まで動かしながら、以下の操作を行う。

1. $S=$ (番号の i ビット目が 1 である数は `1` 、 0 である数は `0` とした文字列) とし、$k=1$ として操作を 1 回行う。
2. $S=$ `"111...1"`, $k=0$ として、P 全体に操作を 1 回行い、左側に偶数、右側に奇数を集める。

1 の操作では、番号の i ビット目が 1 である偶数が右側に移動し、番号の i ビット目が 1 である奇数が左側に移動する[*1]。2 の操作では、右側に移動した偶数が左側に戻り、移動しなかった偶数のすぐ右に移動する。また、左側に移動した奇数が右側に戻り、移動しなかった奇数のすぐ左に移動する。これを繰り返して基数ソートを左右で同時に行うことができる。

基数ソートにおいて、i を 12 まで動かしたときにソートが完了しているような配列の長さの上限は、$2^{13}=8192$ である。偶数および奇数の個数は $N/2 \leq 7500$ であるため、i を 12 まで動かすだけで P を Q と等しくなるように並び替えることができる。

まとめ

解法をまとめると、

1. N が奇数なら、$P_N = N-1$ とする。 (操作 2 回)
2. 偶数を左側に、奇数を右側に集める (操作 1 回)
3. 左右で同時に基数ソートを行い、あと 1 回で昇順に並べられるような順列にする。 (操作 13×2 回)
4. 操作をして昇順に並べ替える。 (操作 1 回)

合計 30 回の操作によって順列 P を昇順に並び替えることができる。

[*1] 番号の i ビット目が 1 である偶数の個数と、番号の i ビット目が 1 である奇数の個数が等しくなるように番号をつけたから、偶数は必ず右側に、奇数は必ず左側に移動する。