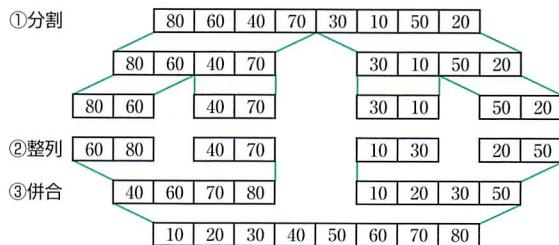


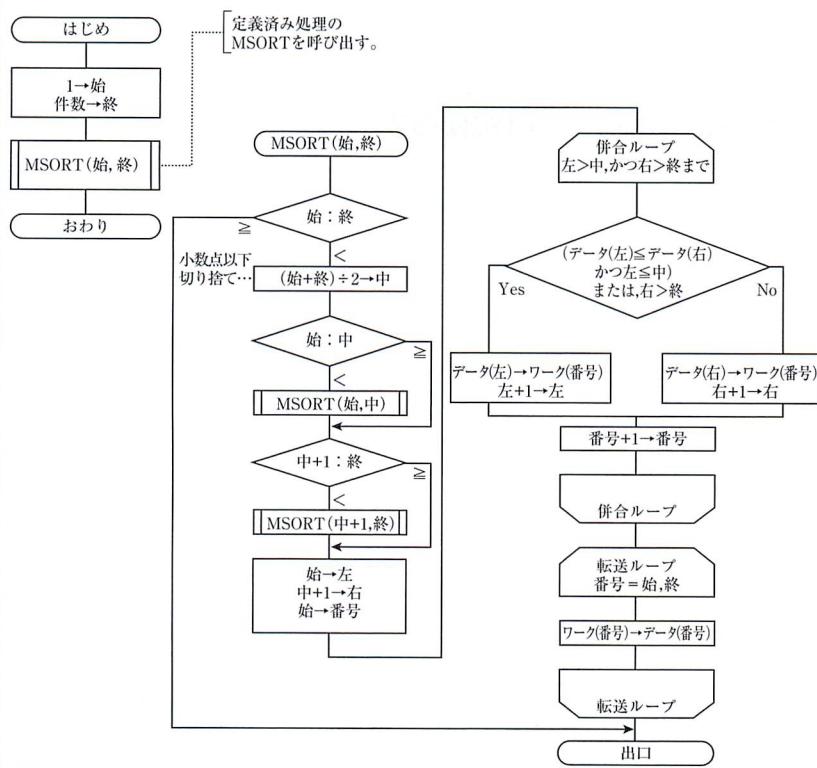
# 21 マージソート

## 1 アルゴリズムの概要

- データ列を複数のデータ列に分割し、個々のデータ列を整列し、併合する。



## 2 流れ図



### 3 整列の様子

件数=8

データ [80 60 40 70 30 10 50 20]

	始	中	終	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
入口 (1, 8)											
入口 (1, 4)											
入口 (1, 2)	1	1	2	60	80	40	70	30	10	50	20
出口 (1, 2)	3	3	4	60	80	40	70	30	10	50	20
入口 (3, 4)	1	2	4	40	60	70	80	30	10	50	20
出口 (3, 4)											
出口 (1, 4)											
入口 (5, 8)											
入口 (5, 6)	5	5	6	40	60	70	80	10	30	50	20
出口 (5, 6)	7	7	8	40	60	70	80	10	30	20	50
入口 (7, 8)	5	6	8	40	60	70	80	10	20	30	50
出口 (7, 8)	1	4	8	10	20	30	40	50	60	70	80
出口 (5, 8)											
出口 (1, 8)											

前半を併合

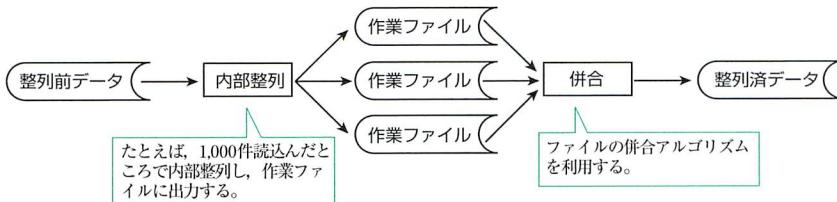
後半を併合

全体を併合

### 1 データ列を分割し、整列後併合するマージソート

マージソートは、データ列の分割、整列、併合を、再帰処理を用いて繰り返します。ここでは、主記憶装置上で行う内部整列の流れ図を示しました。しかし、作業用の領域（ここでは、「ワーク」）が必要になるため、内部整列ではありません。

マージソートは、外部整列に向いているアルゴリズムです。主記憶には一度に読み込めないほど大容量のデータが磁気テープなどのファイル上に存在するような場合に用います。実際には、2個まで分割するのではなく、内部整列できる個数まで分割し、その単位で内部整列し、併合することが多いようです。



**Memo** 232ページに例題があります。また、再帰を用いないマージソートも出題されています。