***Битонная сортировка.***

*Битонная сортировка (*[*англ.*](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA)*Bitonic sorter)*— [параллельный алгоритм](https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%B0%D0%BB%D0%BB%D0%B5%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC) сортировки данных, метод для создания [сортировочной сети](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D1%8C_%D1%81%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B8). Разработан американским информатиком [Кеннетом Батчером](https://ru.wikipedia.org/w/index.php?title=%D0%91%D0%B0%D1%82%D1%87%D0%B5%D1%80,_%D0%9A%D0%B5%D0%BD%D0%BD%D0%B5%D1%82&action=edit&redlink=1) в 1968 году.

*Определение:*

битонная последовательность – последовательность элементов

в которой либо элементы до *k-го* образуют неубывающую последовательность, а после него – невозрастающую последовательность, либо приводится к такому виду с помощью циклического сдвига:

*Примеры битонных последовательностей:*

*1. Алгоритм сортировки битонной последовательности длины*

1. Если , то алгоритм завершается.

2. Разобьем данную последовательность на две:

где *m = n / 2.*

Заметим несколько фактов:

3. Таким образом, если отсортировать последовательности , ответом будет являться последовательность Отсортировать последовательности можно рекурсивно этим же алгоритмом.

*Асимптотика алгоритма 1:*

Будем считать, что алгоритм состоит из фаз, на каждой из которых сортируются части массива длины . Таким образом, на первой фазе выполняется сравнений. Далее массив делится на 2 равные части, в каждой из которых выполняется сравнений. И так далее, нетрудно видеть, что на каждой фазе, кроме последней, массивов вдвое больше, чем на предыдущей, а сравнений в каждом из них выполняется в 2 раза меньше. Значит всего выполняется операций. Значит асимптотика данного алгоритма равна .

*2. Алгоритм преобразования последовательности длины В битонную.*

1. Если , то алгоритм завершается, так как любая последовательность длины такой длины является битонной.

2. Разделим последовательность на две:

где *m = n / 2.*

Запустим этот алгоритм для построения битонных последовательностей из и

3. Отсортируем алгоритмом 1 последовательность по возрастанию и последовательность по убыванию.

4. Возвращаем объединение двух последовательностей:

Очевидно, что при любом результате сравнения последовательностьявляется битонной.

*Асимптотика алгоритма 2:*

Алгоритм рекурсивный, глубина рекурсии, очевидно, равна *.* На каждом уровне выполняется операций (см. асимптотику алгоритма 1). Итоговая асимптотика:

*3. Алгоритм битонной сортировки.*

1. Дополним массив минимальным количеством элементов, равных бесконечности, так, чтобы количество элементов в нем стало равно

2. Преобразуем нашу последовательность алгоритмом 2 в битонную.

3. Отсортируем весь массив алгоритмом 1.

4. Выбросим элементы, равные бесконечности, из конца массива.

*Асимптотика алгоритма битонной сортировки:*

Очевидно, первая и четвертая части выполняются за Для остальных асимптотика доказана: третья часть - вторая -

Асимптотика всего алгоритма:

*Пример:*

.