

External Sorting: Example of multiway external sorting

Ta1: 17, 3, 29, 56, 24, 18, 4, 9, 10, 6, 45, 36, 11, 43

Assume that we have three tapes ($k = 3$) and the memory can hold three records.

A. Main memory sort

The first three records are read into memory, sorted and written on Tb1, the second three records are read into memory, sorted and stored on Tb2, finally the third three records are read into memory, sorted and stored on Tb3.

Now we have one run on each of the three tapes:

Tb1: 3, 17, 29

Tb2: 18, 24, 56

Tb3: 4, 9, 10

The next portion of three records is sorted into main memory and stored as the second run on Tb1:

Tb1: 3, 17, 29, 6, 36, 45

The next portion, which is also the last one, is sorted and stored onto Tb2:

Tb2: 18, 24, 56, 11, 43

Nothing is stored on Tb3.

Thus, after the main memory sort, our tapes look like this:

Tb1: 3, 17, 29, | 6, 36, 45,

Tb2: 18, 24, 56, | 11, 43

Tb3: 4, 9, 10

B. Merging

B.1. Merging runs of length M to obtain runs of length $k \cdot M$

In our example we merge runs of length 3 and the resulting runs would be of length 9.

- a. We build a heap tree in main memory out of the first records in each tape. These records are: 3, 18, and 4.
- b. We take the smallest of them - 3, using the **deleteMin** operation, and store it on tape Ta1.

The record '3' belonged to Tb1, so we read the next record from Tb1 - 17, and insert it into the heap. Now the heap contains 18, 4, and 17.

- c. The next **deleteMin** operation will output 4, and it will be stored on Ta1.

The record '4' comes from Tb3, so we read the next record '9' from Tb3

and insert it into the heap.
Now the heap contains 18, 17 and 9.

d. Proceeding in this way, the **first three runs** will be stored in sorted order on Ta1.

Ta1: 3, 4, 9, 10, 17, 18, 24, 29, 56

Now it is time to build a heap of the **second three runs**.
(In fact they are only two, and the run on Tb2 is not complete.)

The resulting sorted run on Ta2 will be:

Ta2: 6, 11, 36, 43, 45

This finishes the first pass.

B.2. Building runs of length $k \cdot k \cdot M$

We have now only two tapes: Ta1 and Ta2.

- We build a heap of the first elements of the two tapes - 3 and 6, and output the smallest element '3' to tape Tb1.
- Then we read the next record from the tape where the record '3' belonged - Ta1, and insert it into the heap.
- Now the heap contains 6 and 4, and using the **deleteMin** operation the smallest record - 4 is output to tape Tb1.

Proceeding in this way, the entire file will be sorted on tape Tb1.

Tb1: 3, 4, 6, 9, 10, 11, 17, 18, 24, 29, 36, 43, 45, 56

The number of passes for the multiway merging is $\log_k(N/M)$.

In the example this is $\lceil \log_3(14/3) \rceil + 1 = 2$.

[Back to External Sort](#)