Performance of plain B-trees:

$O(\frac{\lg^2 N}{B} + \log_B N)$ amortized memory transfers/update

$O(\log_B N)$/search — $O(\lceil K/B \rceil)$ for traversal of K.

Indirection: cluster $\lg N$ consecutive elements

Search: top cost same — in $O(\log_B N)$
bottom — can scan in $O(\frac{\lg N}{B})$
$\Rightarrow O(\log_B N)$

Update: top cost — $O(\frac{\lg N}{B} + \frac{1}{\lg B})$ am.
bottom cost — can scan in $O(\frac{\lg N}{B})$
$\Rightarrow O(\frac{\lg N}{B})$ amortized + search cost.

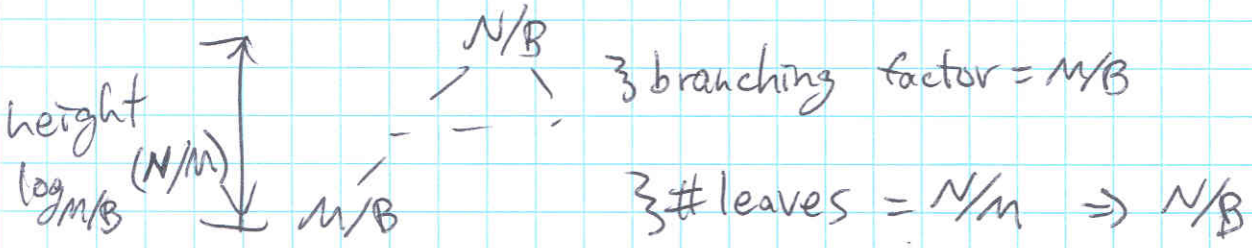
Can B-trees sort?

- N inserts + searches
- traversal of N

$O(N \log_B N)$ memory transfers

(M/B)-way merge sort (external memory)

$$T(N) = (M/B)\, T(N/(M/B)) + O(N/B)$$
$$T(M) = O(M/B)$$



$\Rightarrow$ total cost $= O(N/B \log_{M/B} N/M)$.

TOPS FORM 33041
LITHO IN U.S.A.