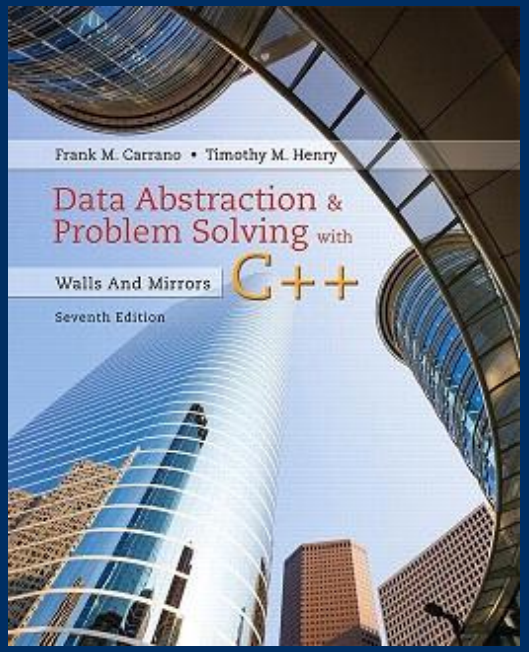
Processing Data in External Storage

CS 302 - Data Structures

M. Abdullah Canbaz



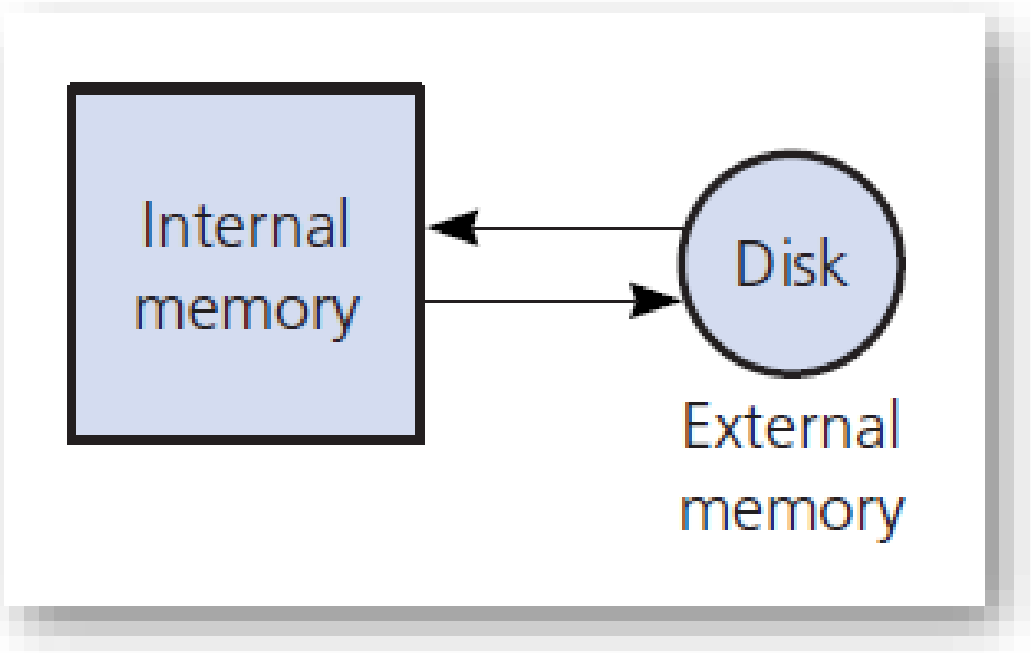
Modified from authors' slides

Look at External Storage

- External storage
  - Used when program reads/writes data to/from a C++ file
- Generally there is more external storage than internal memory
- Direct access files essential for external data collections

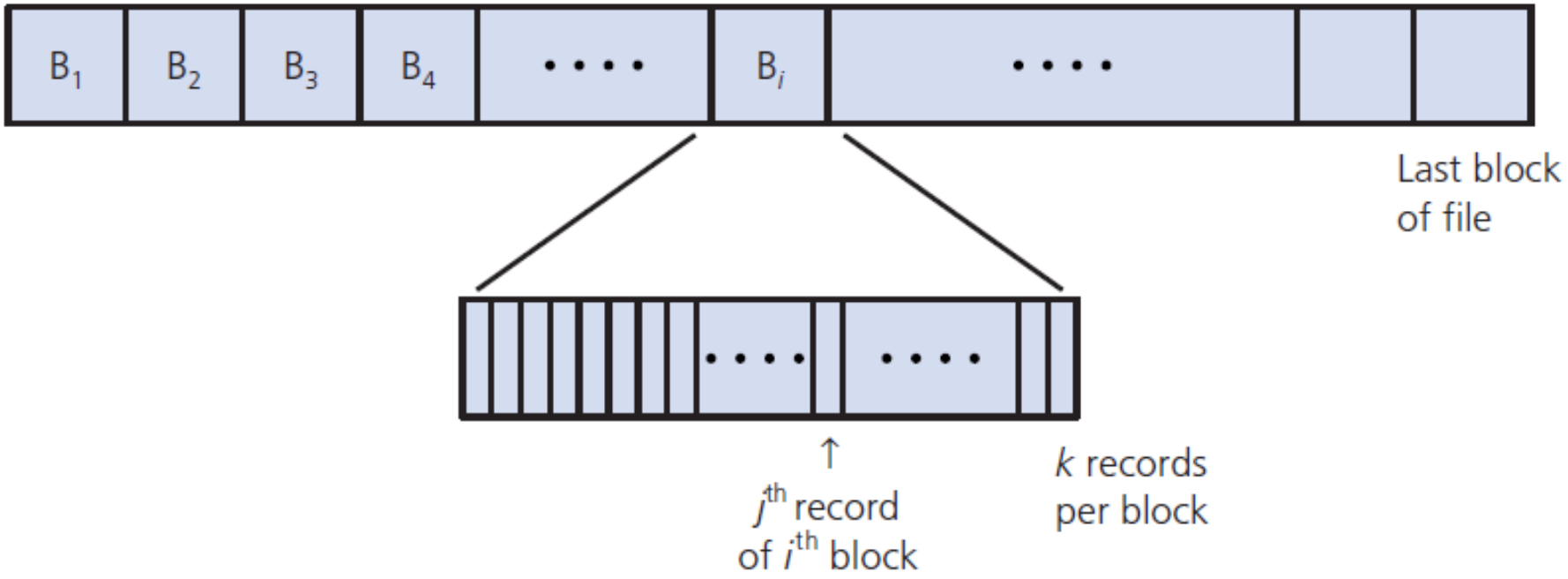
Look at External Storage

- Internal and external memory



Look at External Storage

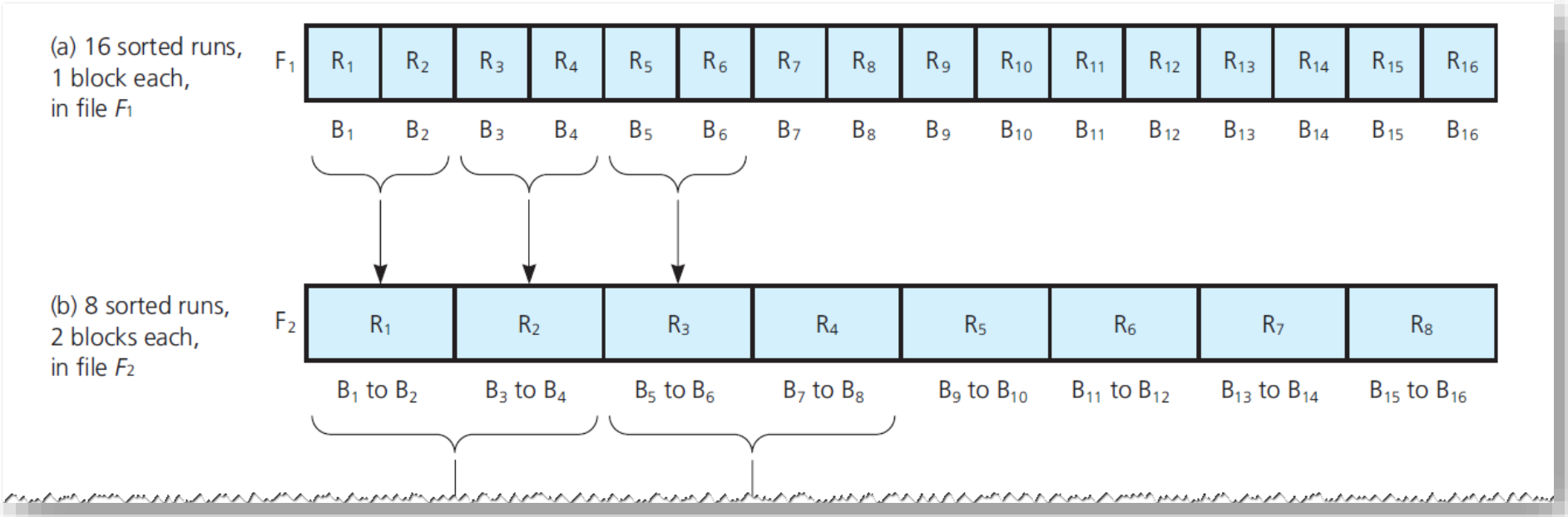
- A file partitioned into blocks of records



Look at External Storage

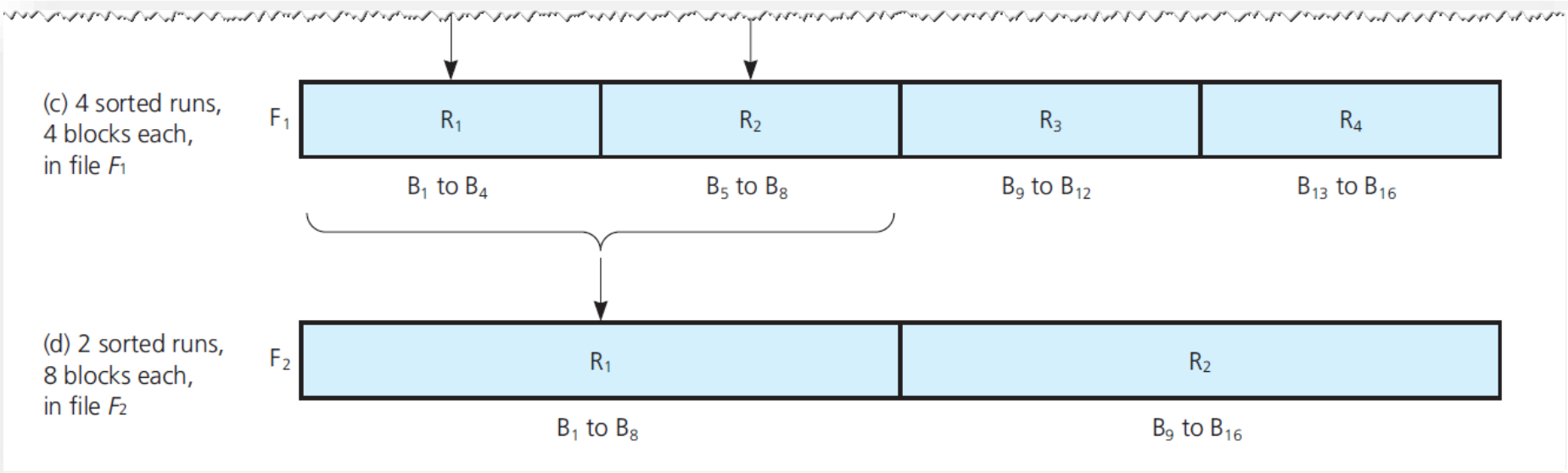
- Direct access I/O
  - Involves blocks instead of records
- Buffer stores data (blocks) temporarily
- Record updated within block (in buffer)
- Work to minimize block I/O
  - Takes more time for disk access

Working with External Data



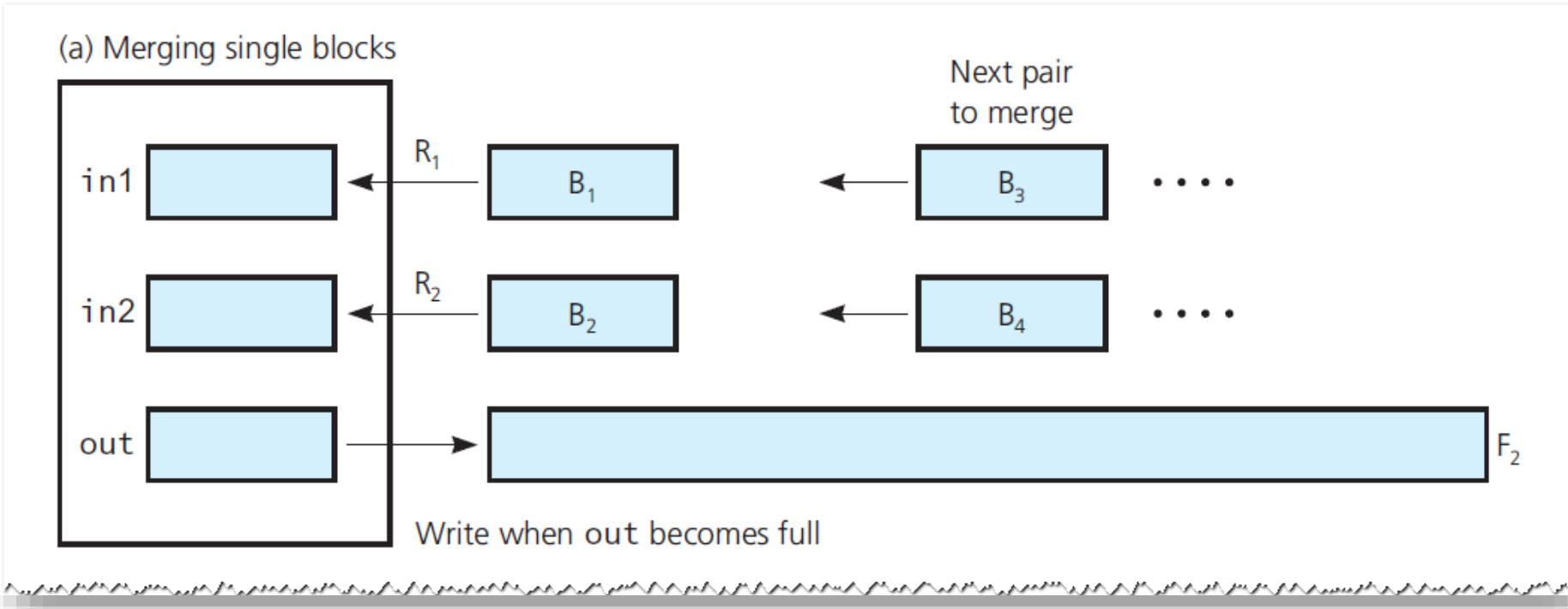
- Sorting a block of an external file F by merging the results of internal sorts and using two external work files F_1 and F_2

Working with External Data



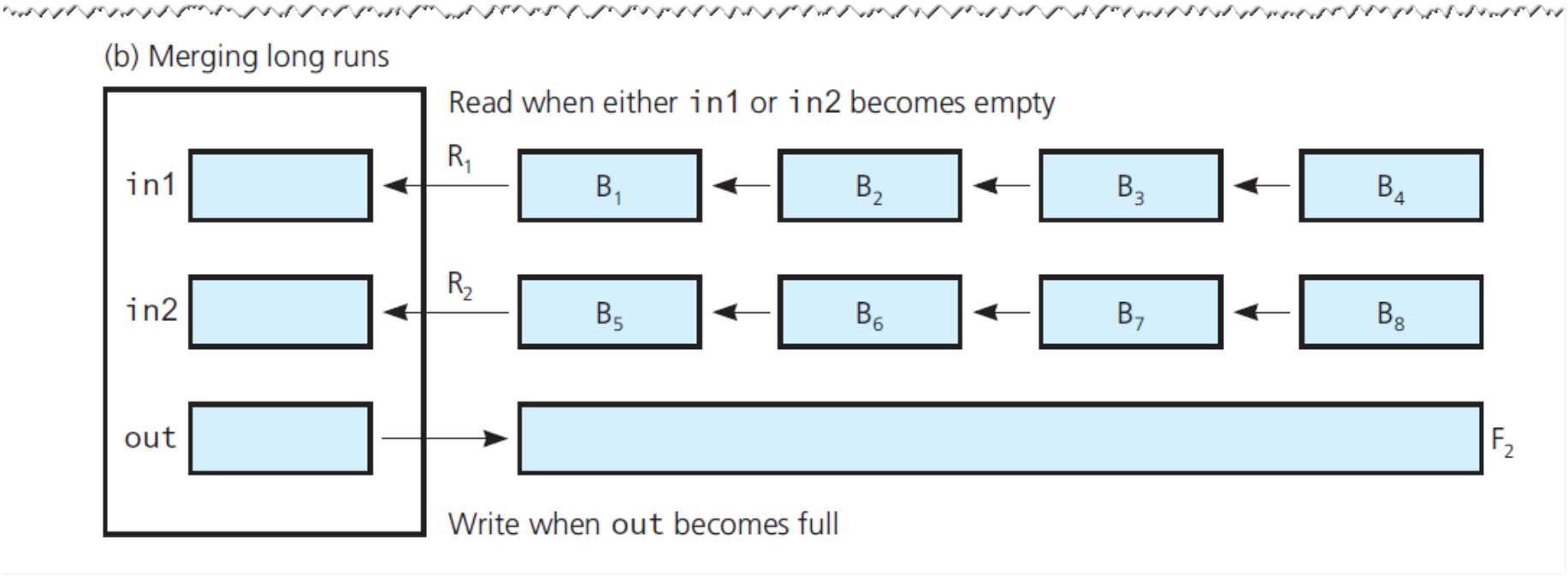
- Sorting a block of an external file F by merging the results of internal sorts and using two external work files F_1 and F_2

Working with External Data



- Phase 2 of an external sort: Merging sorted runs

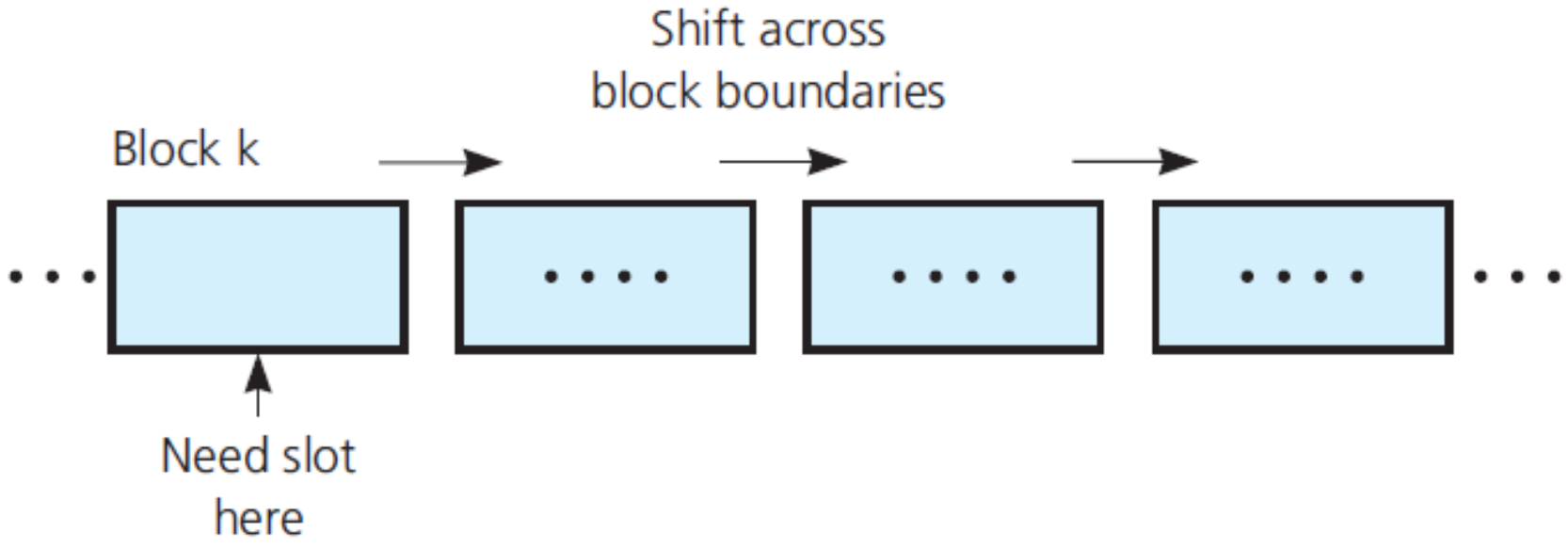
Working with External Data



- Phase 2 of an external sort: Merging sorted runs

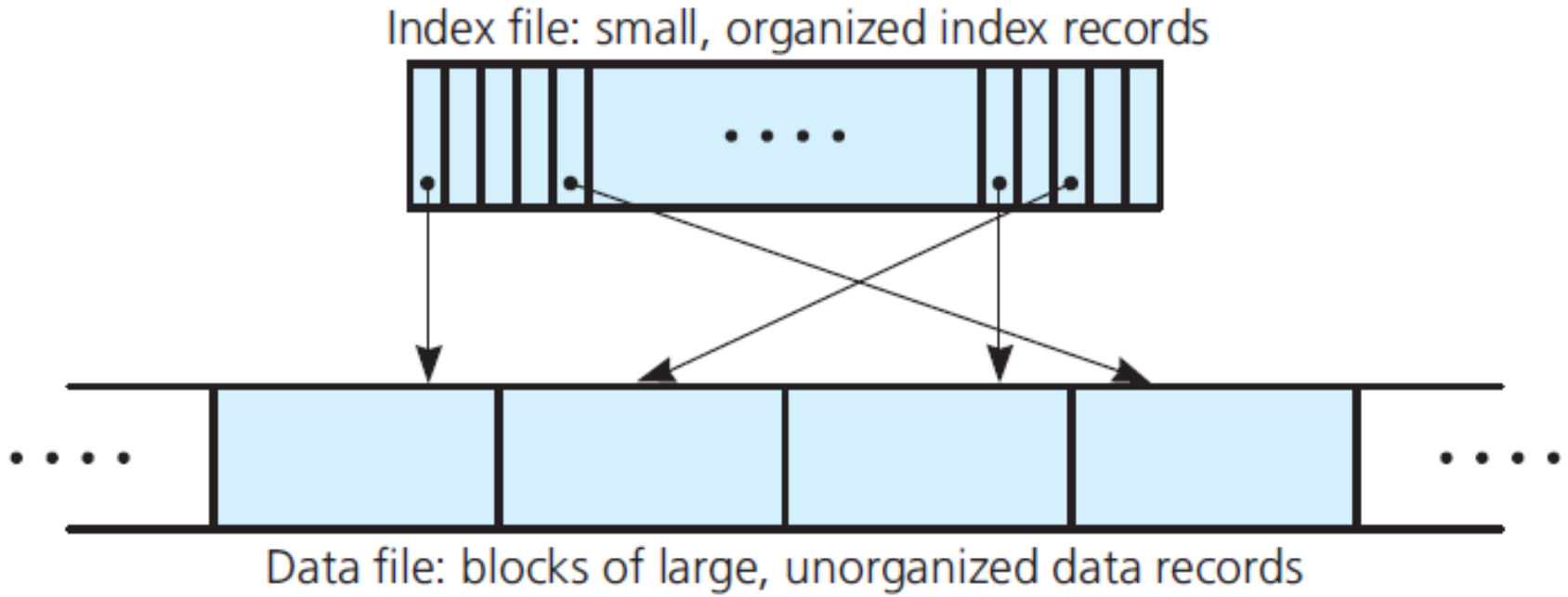
Basic Data Management Operations

- Shifting across block boundaries



Indexing an External File

- A data file with an index



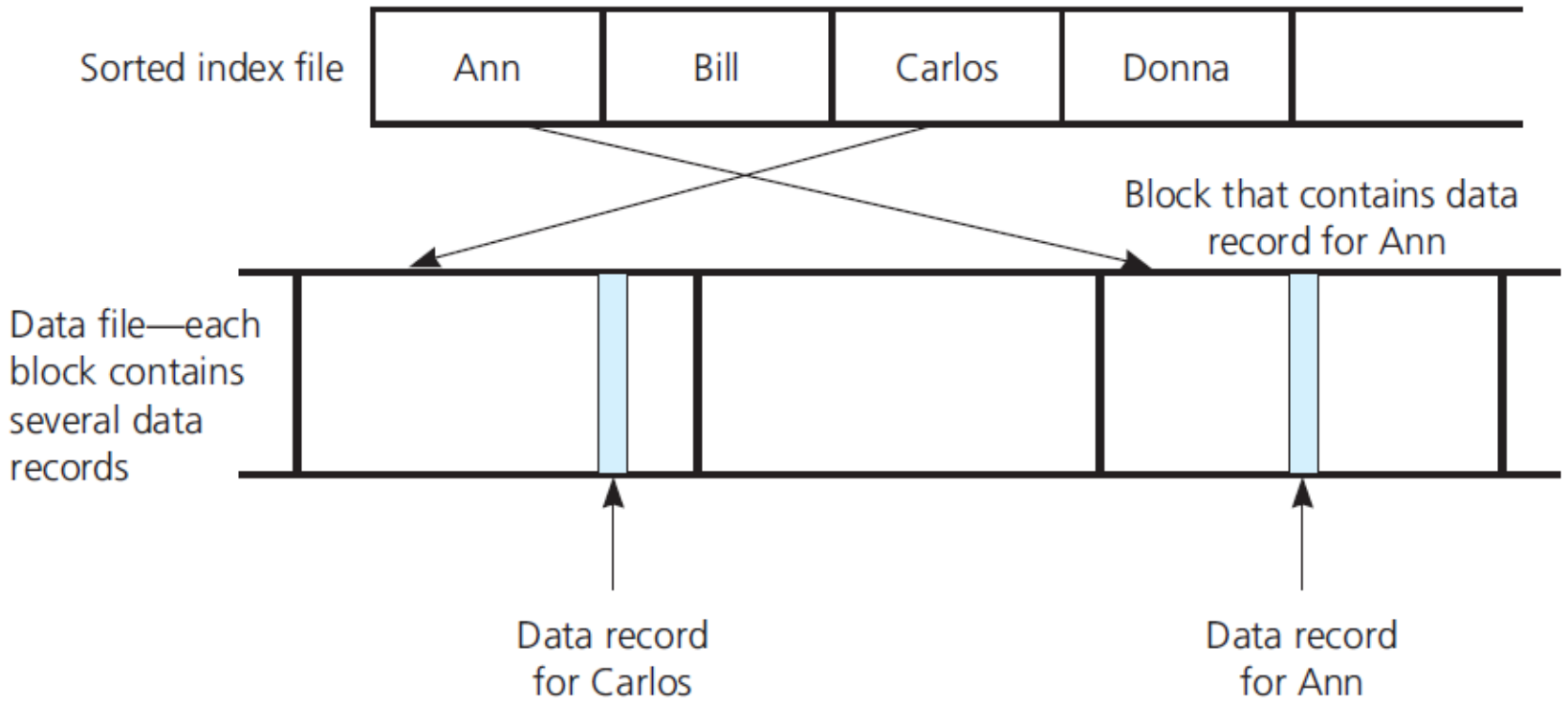
Indexing an External File

Advantages of an index file

- Index record smaller than a data record
- Data file need not be kept in any particular order
- Possible to maintain several indexes simultaneously
- Index file reduces number of block accesses
- Shift index records, not data records

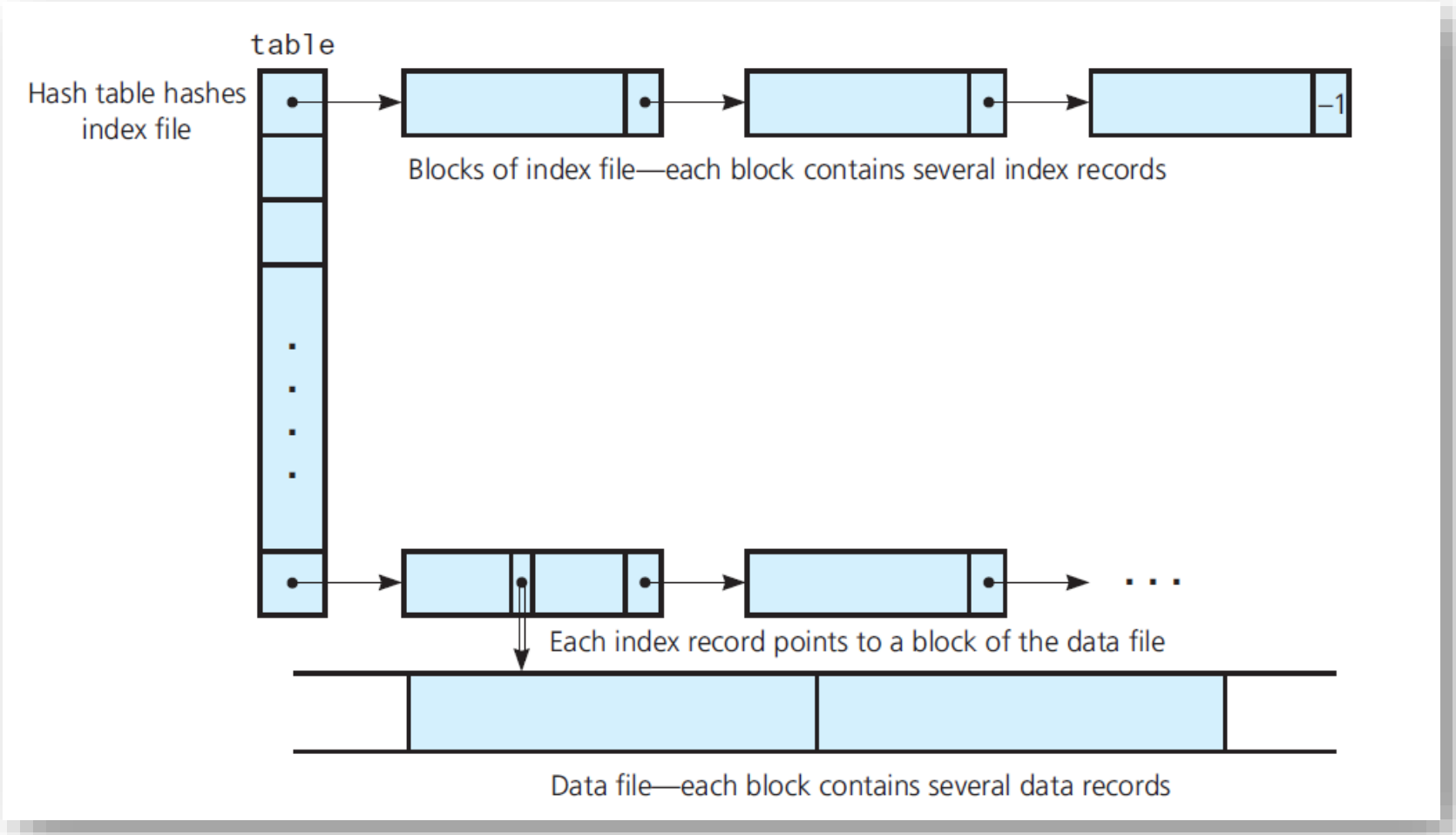
Indexing an External File

- A data file with a sorted index file



External Hashing

- A hashed index file



External Hashing

- A single block with a pointer

External Hashing

Addition when an index file uses external hashing

1. Add data record into data file.
2. Add corresponding index record into index file

$$i = h(searchKey)$$

External Hashing

Removal when an index file uses external hashing

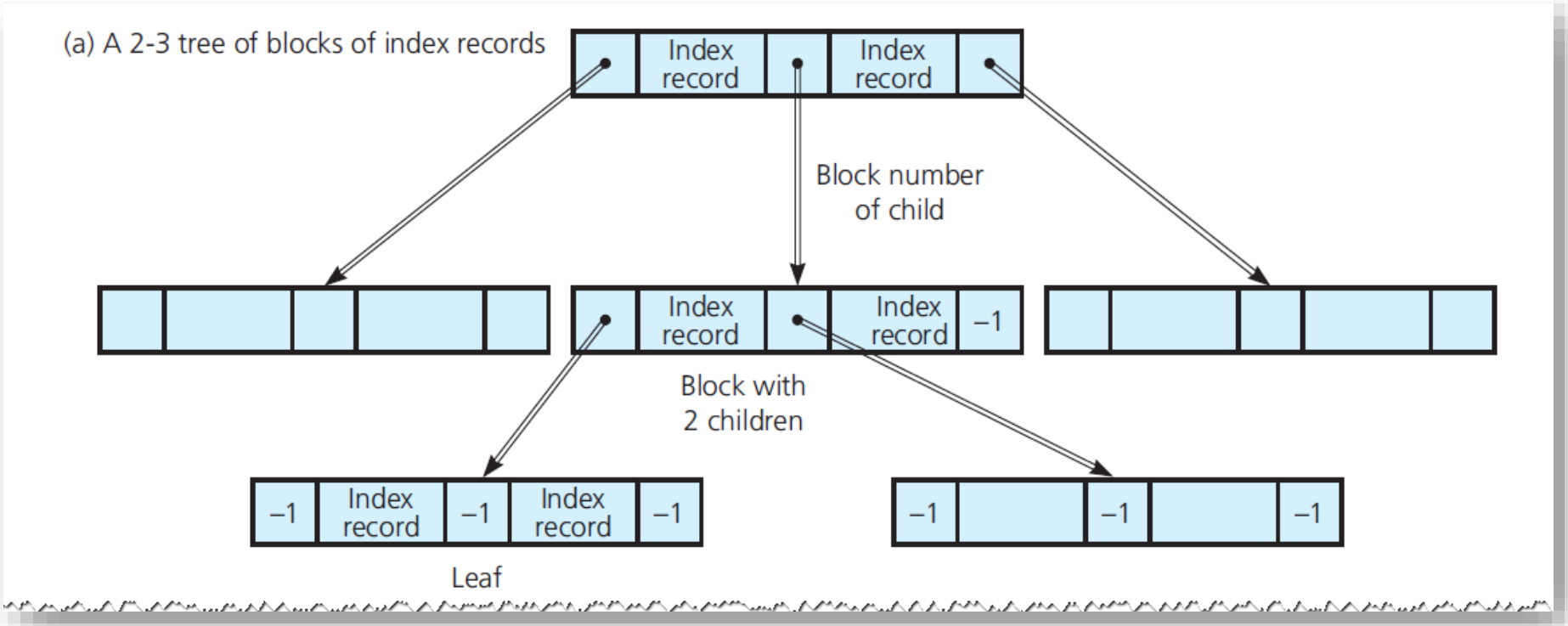
1. Search index file for corresponding index record

$$i = h(searchKey)$$

2. Remove data record from data file

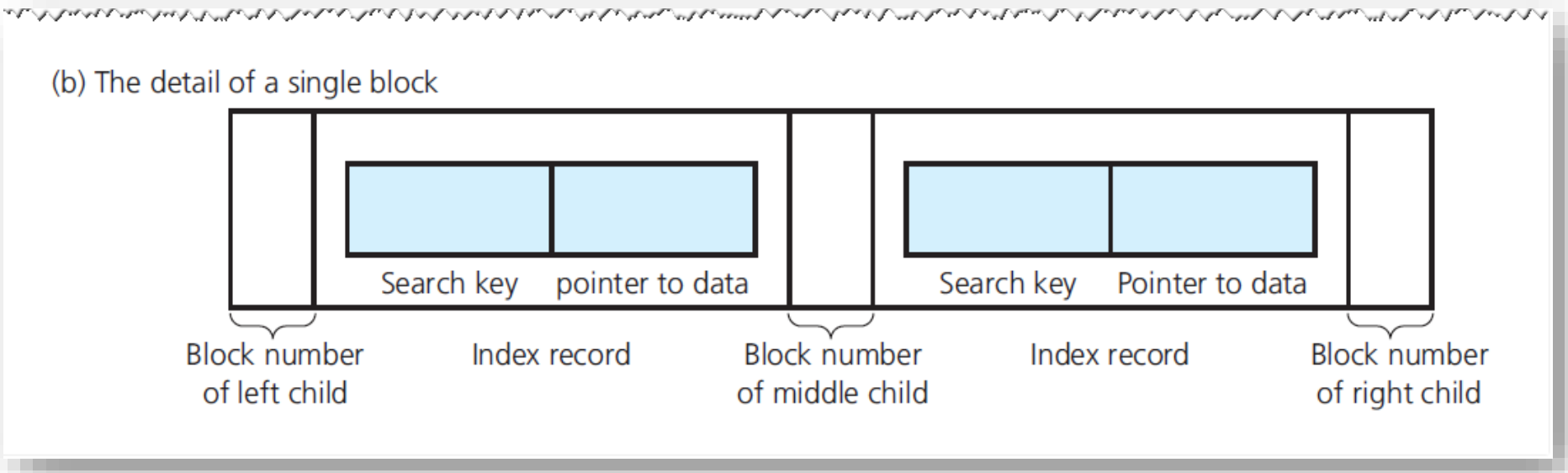
B-Trees

- An index file organized as a 2-3 tree



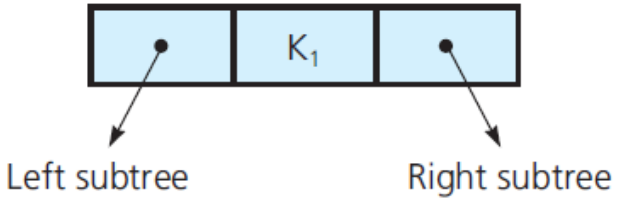
B-Trees

- An index file organized as a 2-3 tree

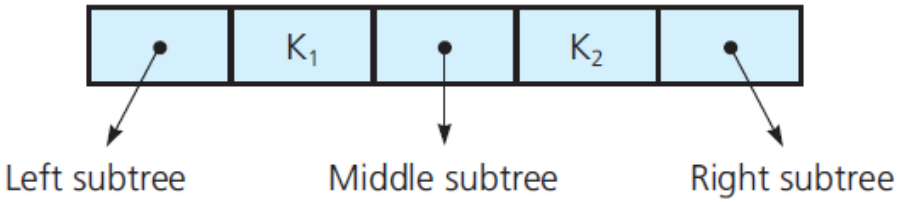


B-Trees

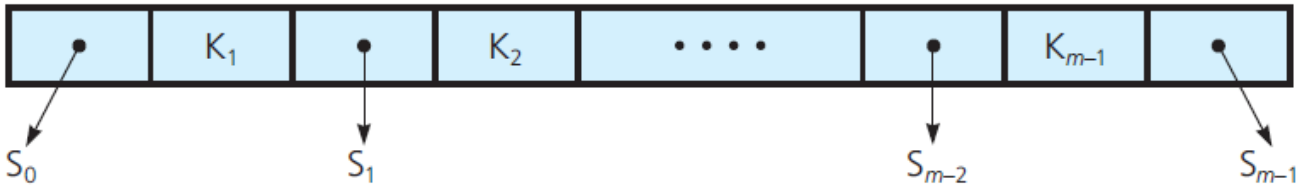
(a) A node with two children has one search key



(b) A node with three children has two search keys

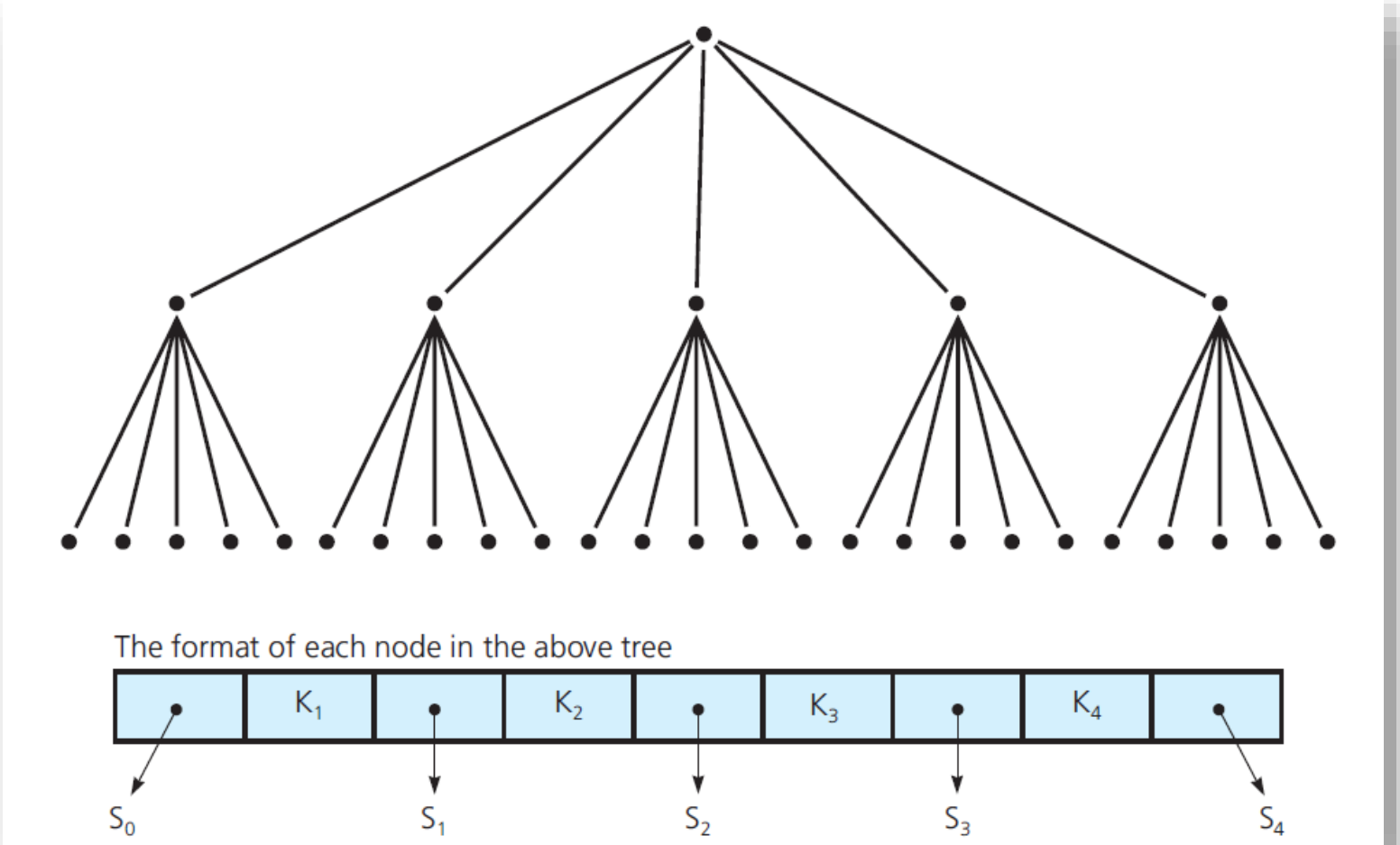


(c) A node with m children has $m - 1$ search keys



- Nodes with two, three, and m children and their search keys

B-Trees



- A full tree whose internal nodes have five children

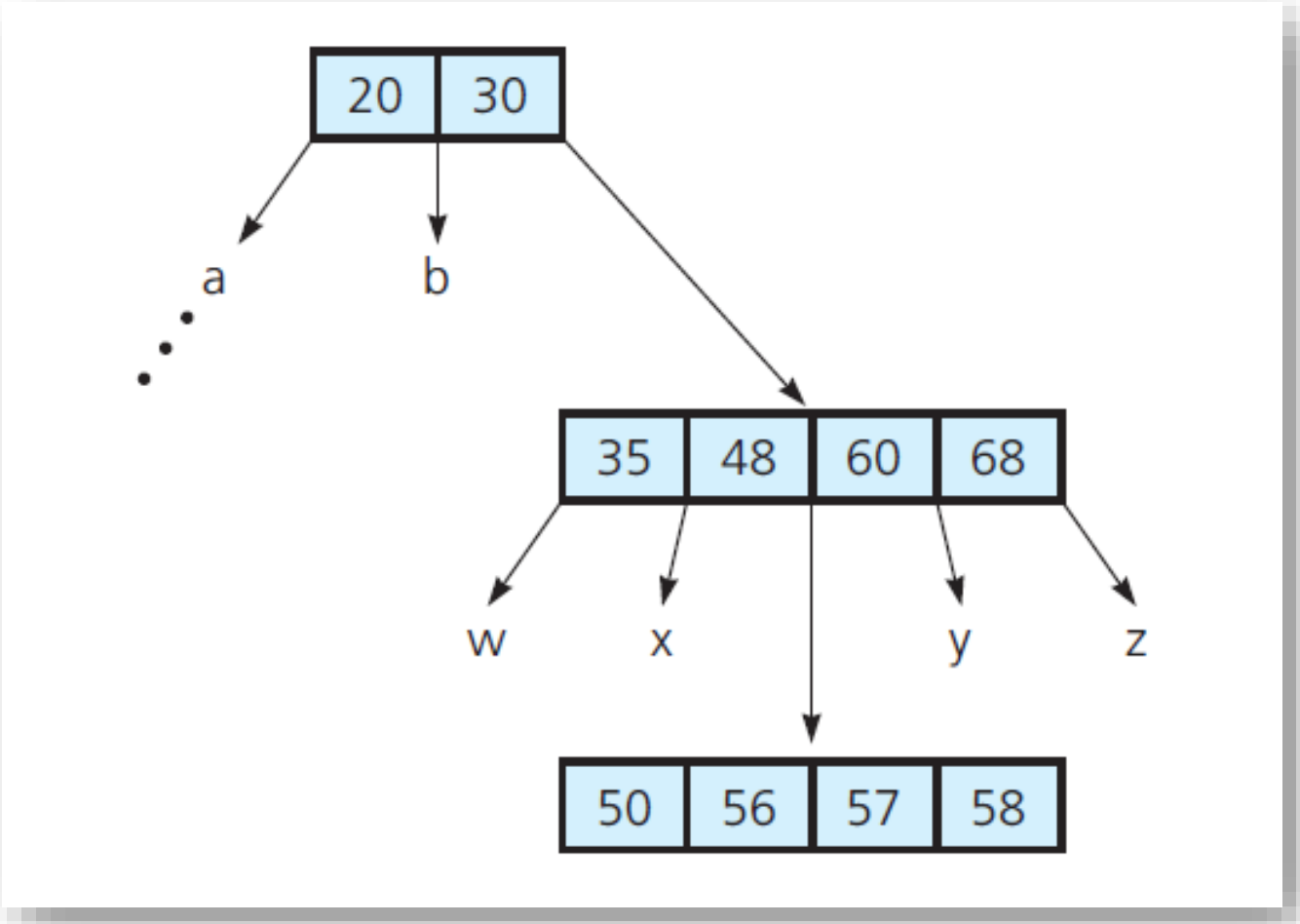
B-Trees

Adding a record to a B-tree

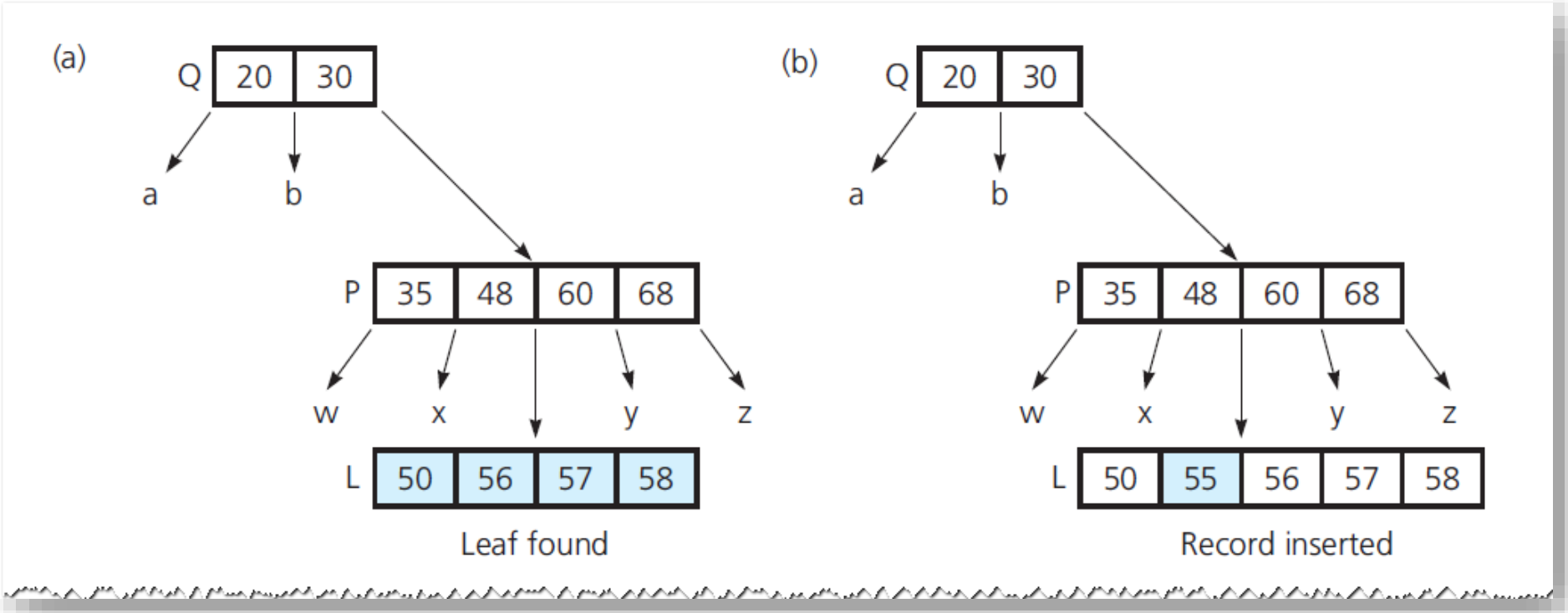
1. Add data record to data file
2. Add a corresponding index record to index file

B-Trees

- A B-tree of degree 5

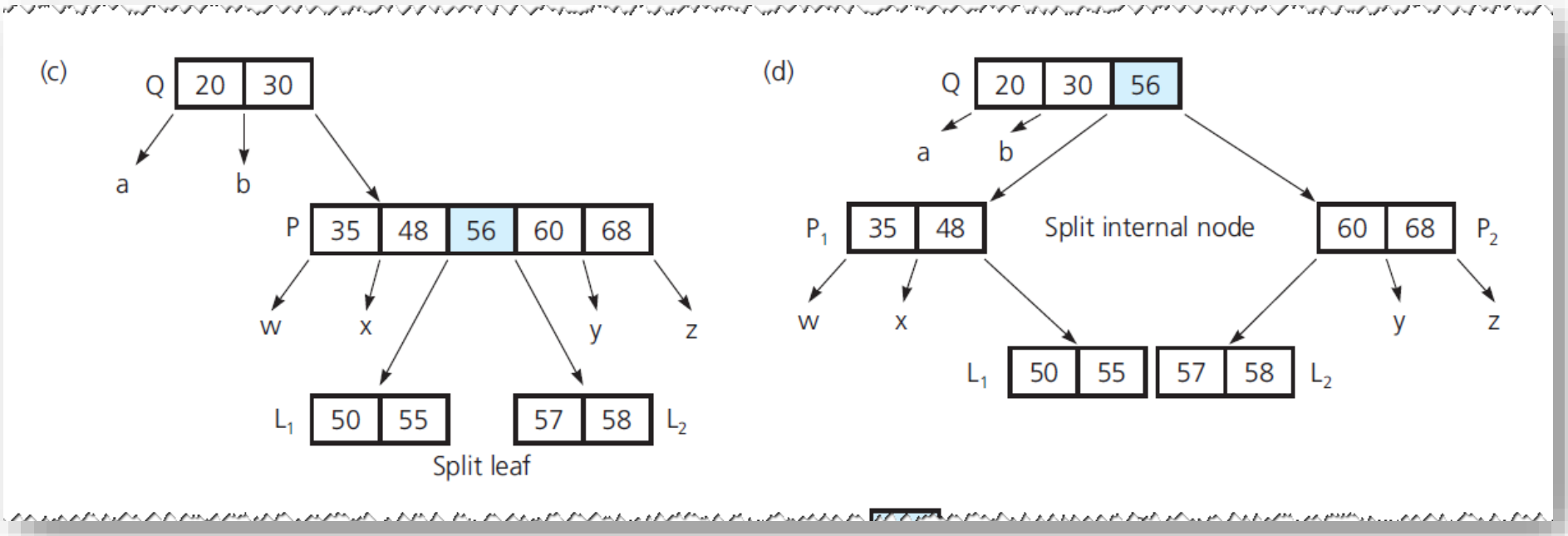


B-Trees



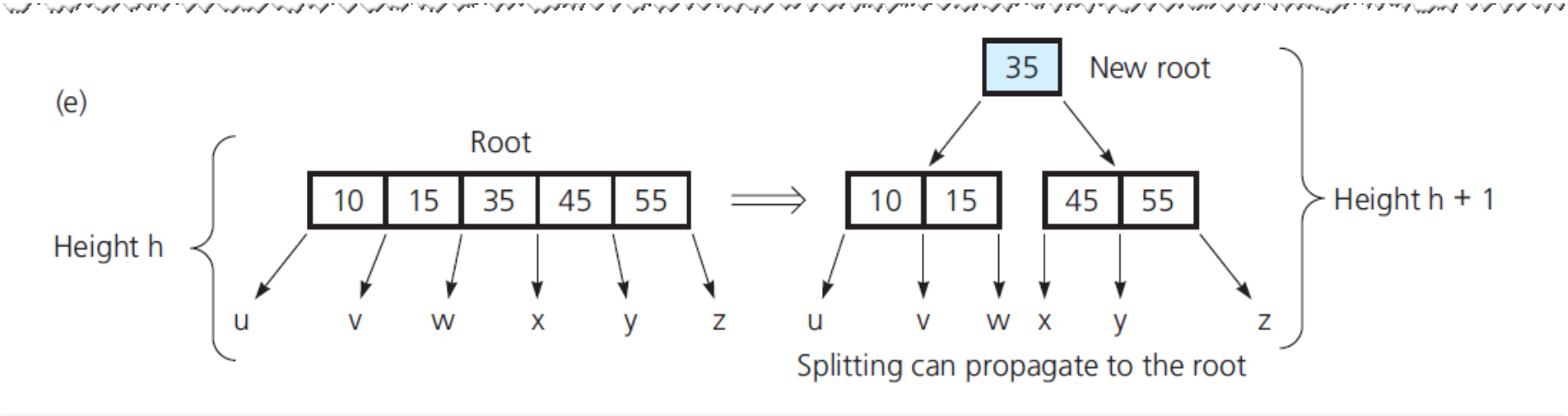
- (a through d) The steps for adding 55 to a B-tree; (e) splitting the root

B-Trees



- (a through d) The steps for adding 55 to a B-tree; (e) splitting the root

B-Trees



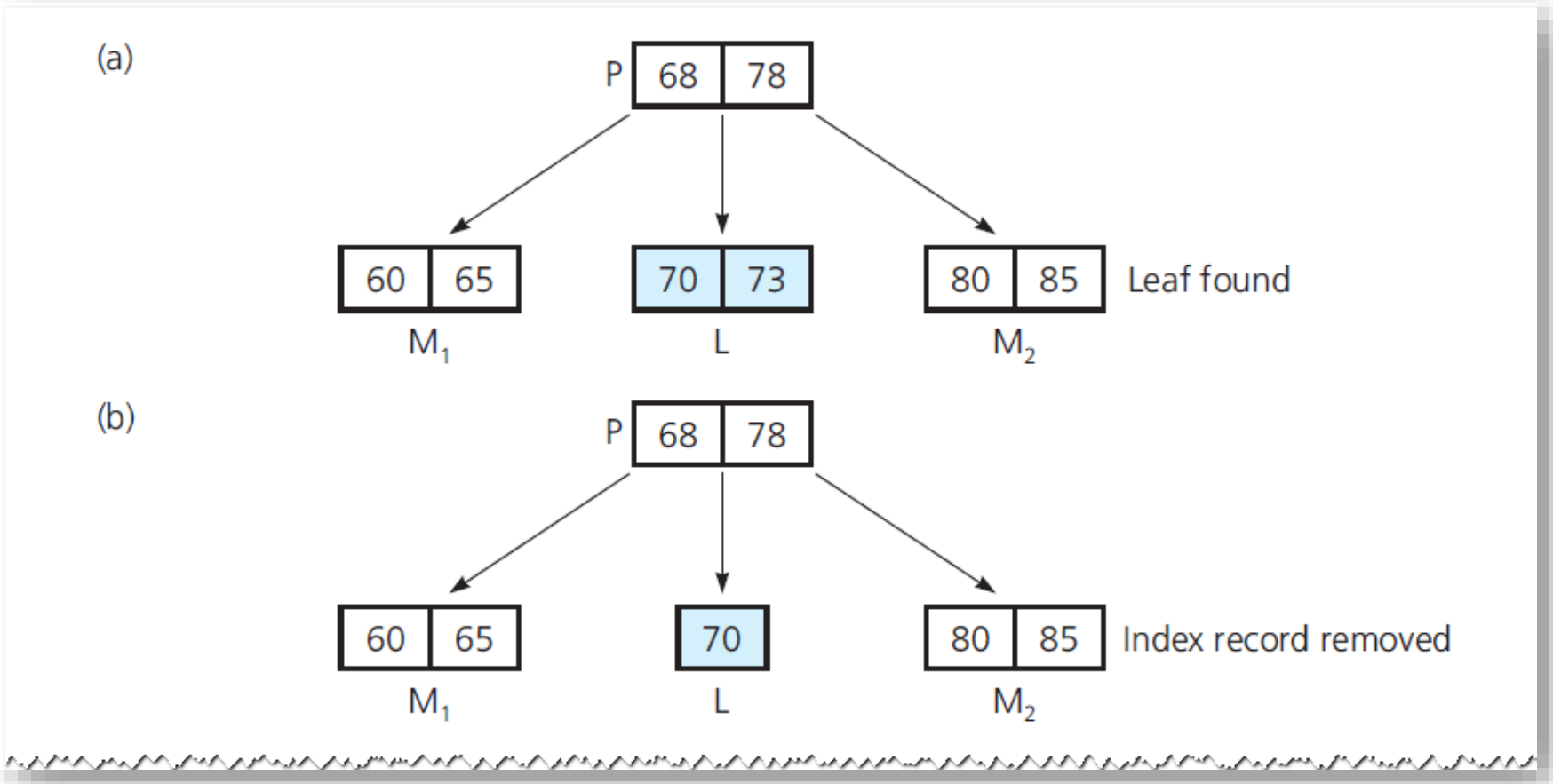
- (a through d) The steps for adding 55 to a B-tree; (e) splitting the root

B-Trees

Removing a record from a B-tree

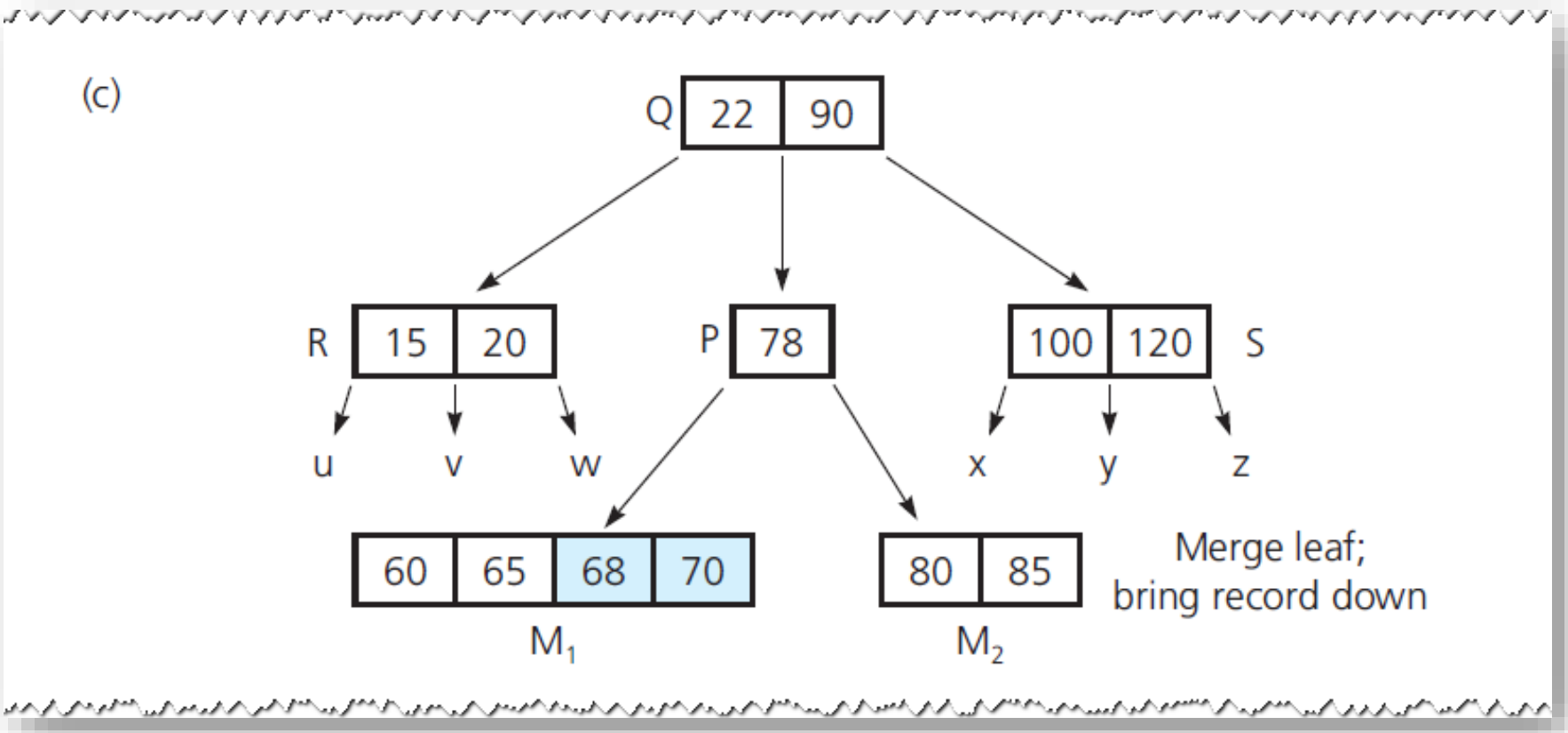
1. Locate index record in index file
2. Remove data record from data file

B-Trees



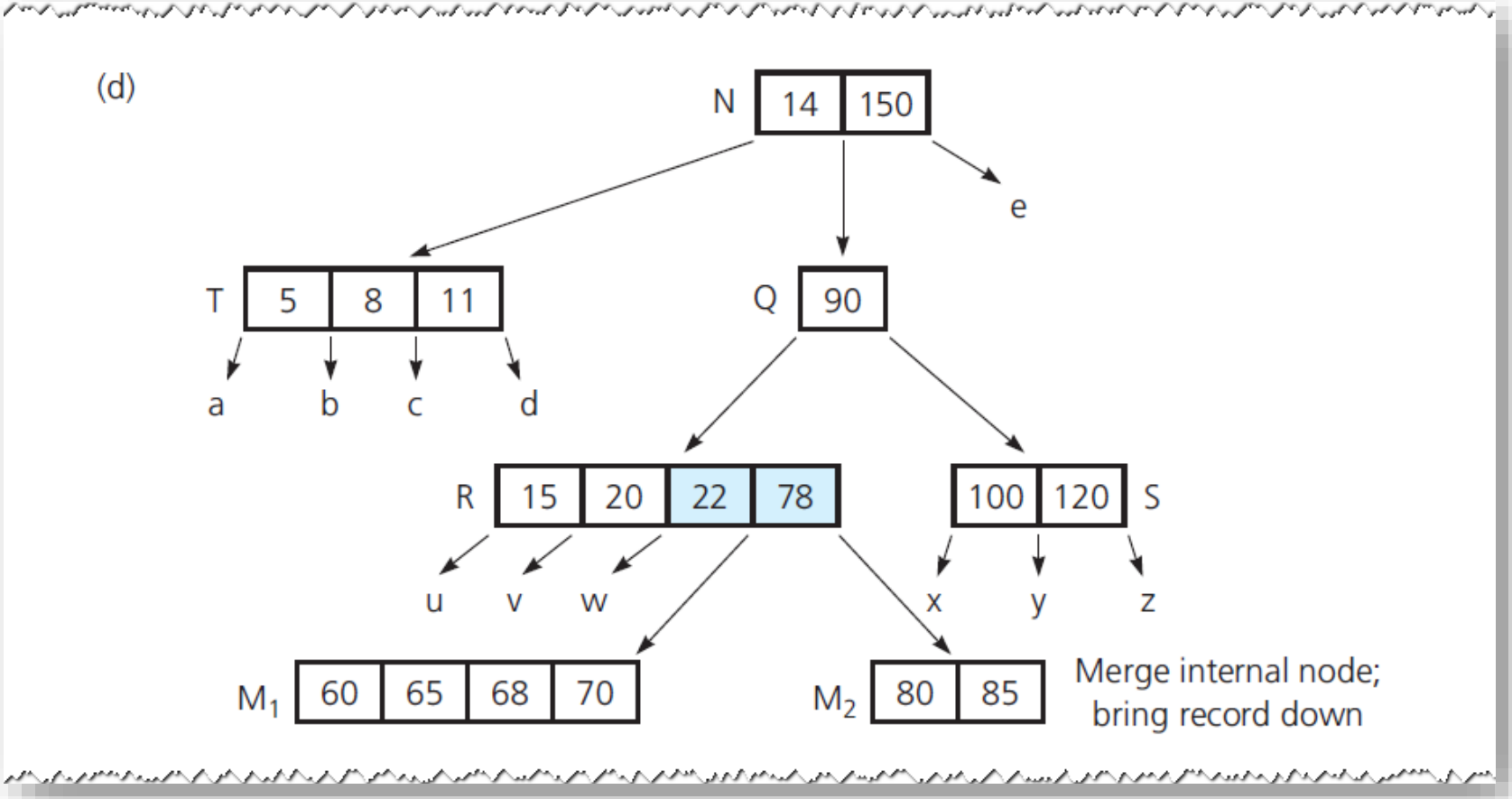
- (a through e) The steps for removing 73 ;
  (f) removing the root

B-Trees



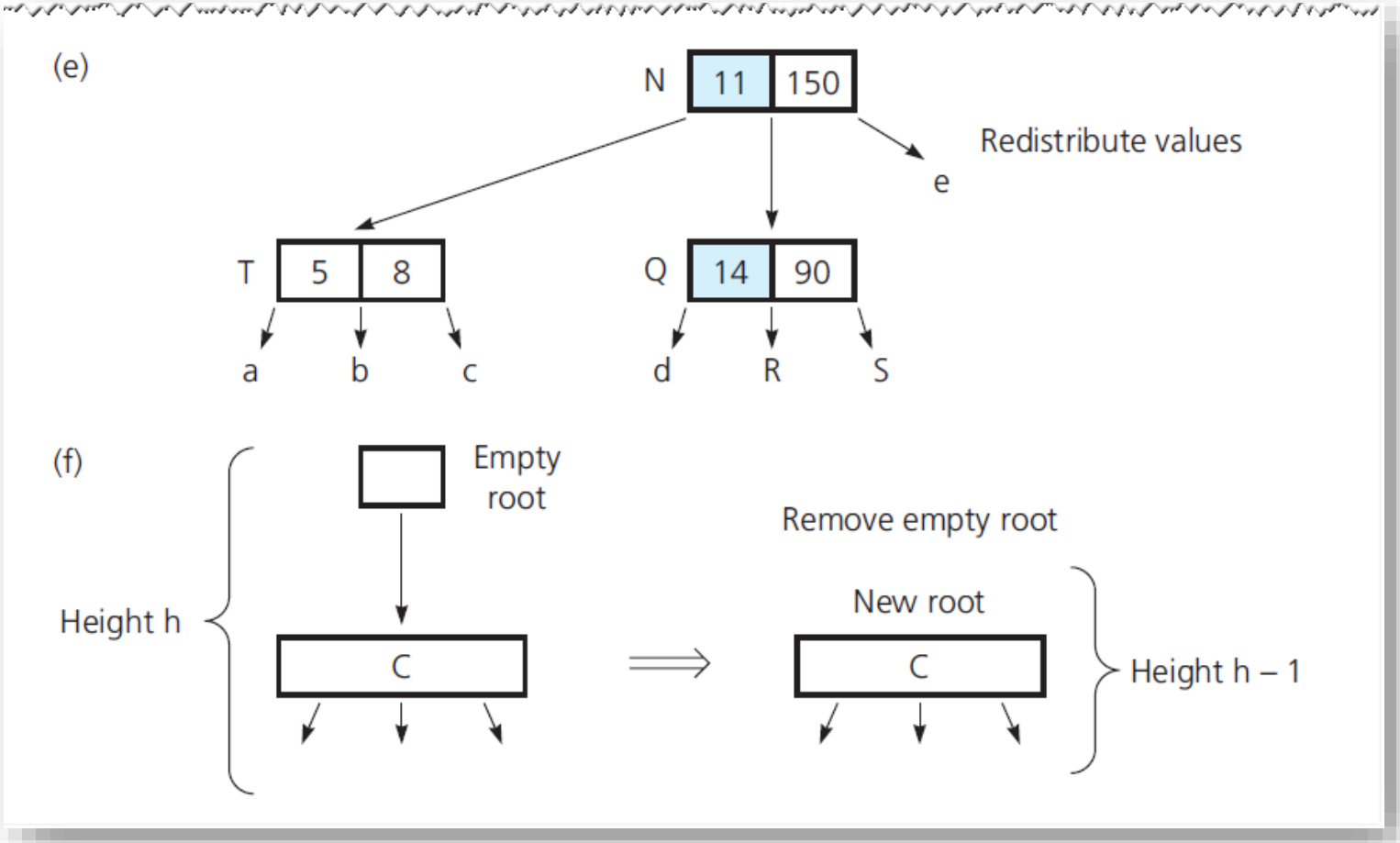
- (a through e) The steps for removing 73 ;
  (f) removing the root

B-Trees



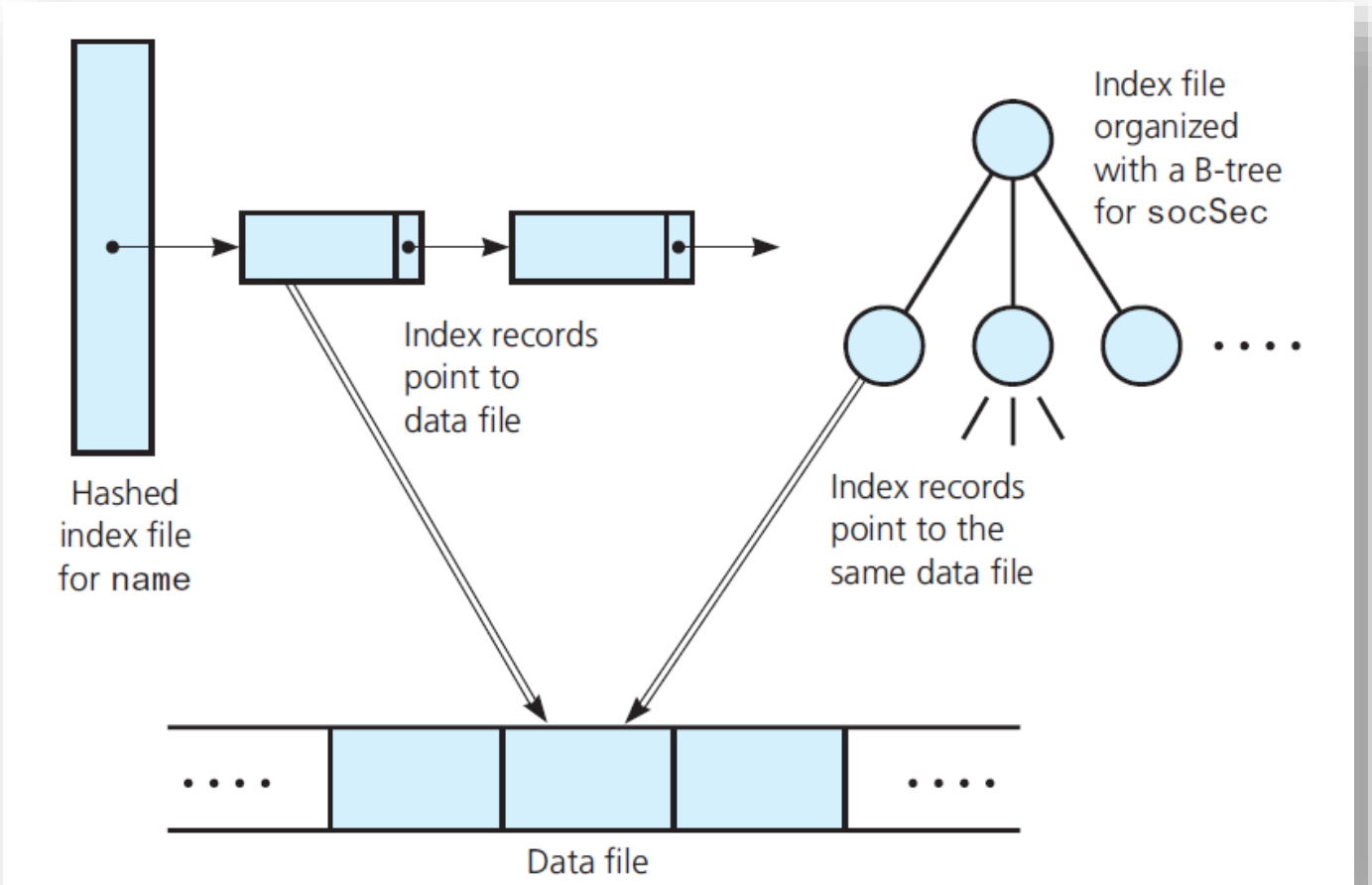
- (a through e) The steps for removing 73 ; (f) removing the root

B-Trees



- (a through e) The steps for removing 73 ; (f) removing the root

Multiple Indexing



- Multiple index files

Multiple Indexing

A removal by name must update both indexes

1. Search name index file for jones and remove index record.
2. Remove appropriate data record from data file, noting socSec value ssn of this record.
3. Search socSec index file for ssn and remove this index record.


HAPPY FINALS WEEK
AND MAY THE GRADES BE EVER IN YOUR FAVOR!
© Lionsgate
quickmeme.com