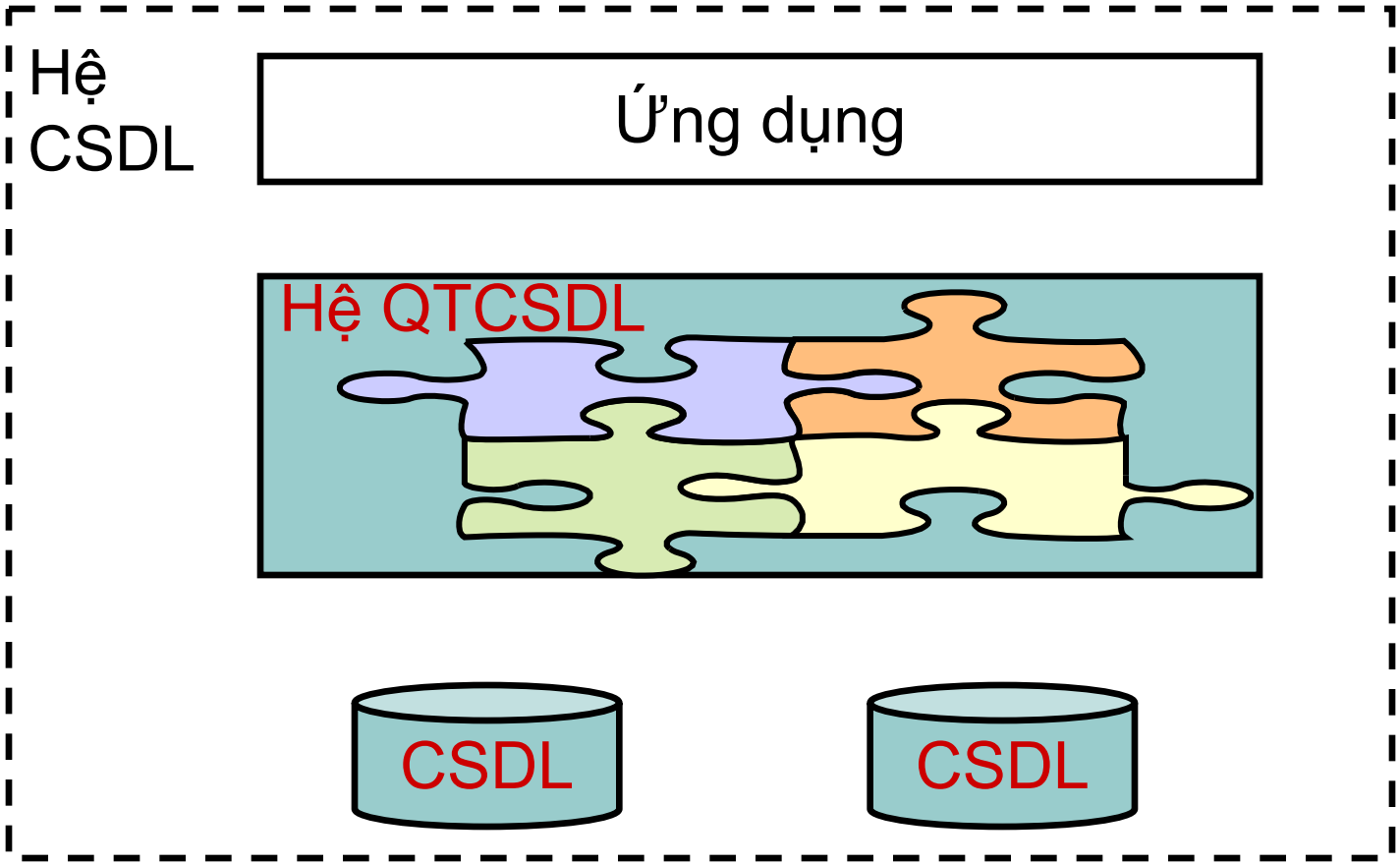
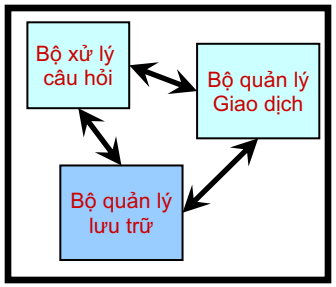
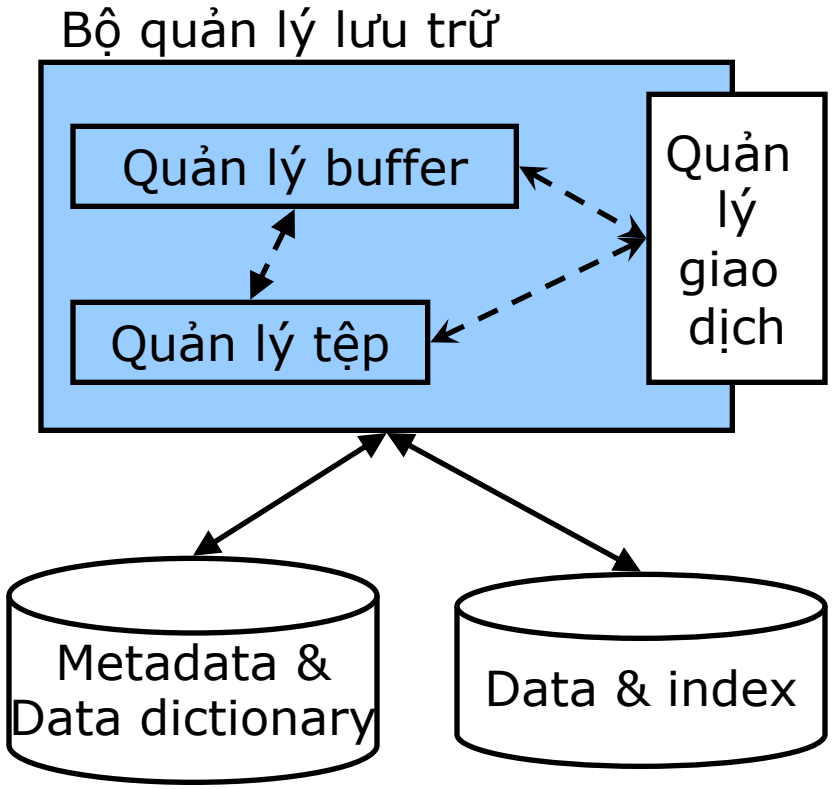
Tổ chức dữ liệu vật lý


Hệ
CSDL
Ứng dụng
Hệ QTCSDL
CSDL
CSDL



Quản lý lưu trữ

- Tổ chức tệp: sắp xếp các bản ghi trên thiết bị nhớ ngoài
  - RID (*record id):* xác định địa chỉ vật lý của các bản ghi
  - chỉ số: cấu trúc dữ liệu xác định sự tương ứng giữa RID của bản ghi và giá trị của trường (khoá)
- Vùng nhớ đệm: trung gian giữa thiết bị nhớ ngoài và bộ nhớ trong (có thể sử dụng cho cả DL và chỉ số)



Các thiết bị nhớ ngoài

- Đĩa từ, băng từ, ...
- Đĩa từ: được tổ chức thành từng block
  - Chí phí truy nhập đến các block bất kỳ là tương đương
  - Chí phí đọc nhiều block liền nhau < chí phí đọc các block đó theo thứ tự bất kỳ
- Băng từ:
  - chỉ có thể đọc được các block liền nhau
  - rẻ hơn đĩa từ nhưng chi phí truy nhập thương lớn hơn
- ...

Đĩa từ vs. bộ nhớ trong

- Tốc độ truy nhập bộ nhớ

  ms vs. ns (~1000 lần)
- Kích thước

  GB vs. 10x MB (~ 100 lần với cùng chi phí)
- Lưu trữ

  ổn định (kể cả khi mất điện) vs. tạm thời
- Phân chia block

  4KB vs. 1Byte

Tổ chức bộ nhớ ngoài

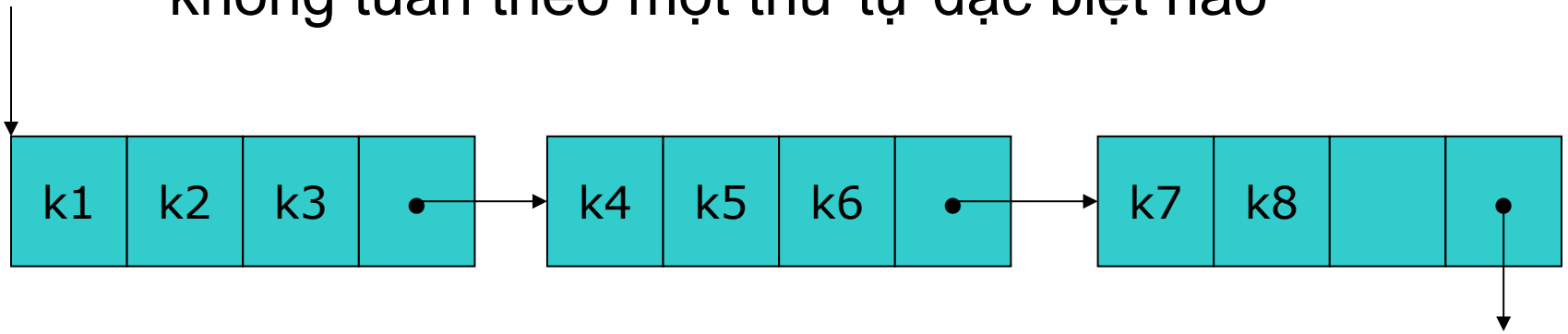
- Mục đích: giảm thiểu truy xuất đến dữ liệu không cần thiết trên thiết bị nhớ ngoài
- Các vấn đề cần quan tâm
  - Cấu trúc lưu trữ
  - Các phép toán (thêm, xoá, sửa, tìm kiếm)
- Mỗi tệp dữ liệu chiếm 1 hoặc nhiều khối
  Mỗi khối chứa 1 hoặc nhiều bản ghi

Nội dung

- ✓ Tổng quan về tổ chức bộ nhớ ngoài
- Tổ chức tệp đống
- Tổ chức tệp băm
- Tổ chức tệp chỉ dẫn
- Cây cân bằng

Tổ chức tệp đống (*Heap File*)

- Lưu trữ kế tiếp các bản ghi trong các khối không tuân theo một thứ tự đặc biệt nào

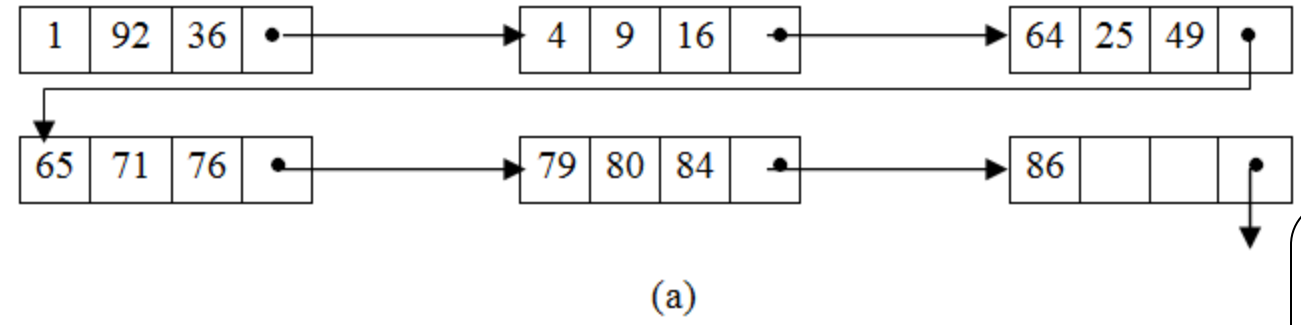


- Có các con trỏ trỏ tới tất cả các khối (block) của tệp và các con trỏ này được lưu trữ ở bộ nhớ trong.

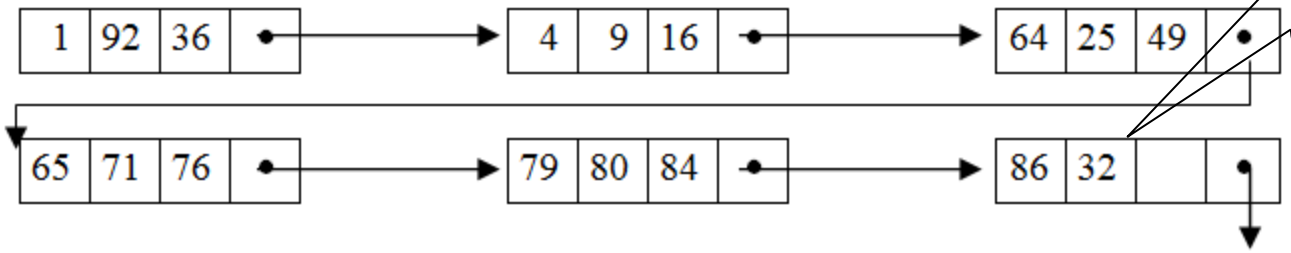
Các phép toán

- Tìm kiếm 1 bản ghi:
  - tìm kiếm một bản ghi có giá trị khóa cho trước => *quét toàn bộ tệp*
- Thêm 1 bản ghi:
  - thêm bản ghi mới vào sau bản ghi cuối cùng
- Xoá 1 bản ghi
  - Tìm kiếm + đánh dấu xóa → hệ thống cần tổ chức lại đĩa theo định kỳ
- Sửa đổi một bản ghi:
  - Tìm kiếm và sửa các trường

Ví dụ

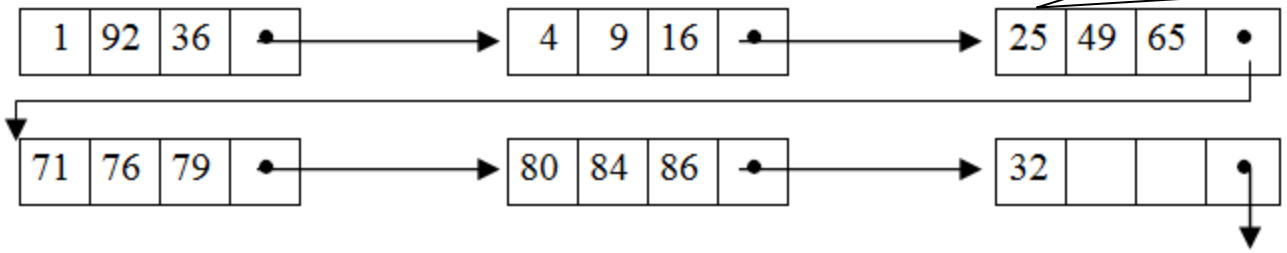


(a)



Thêm bản ghi có giá trị khóa là 32

(b)



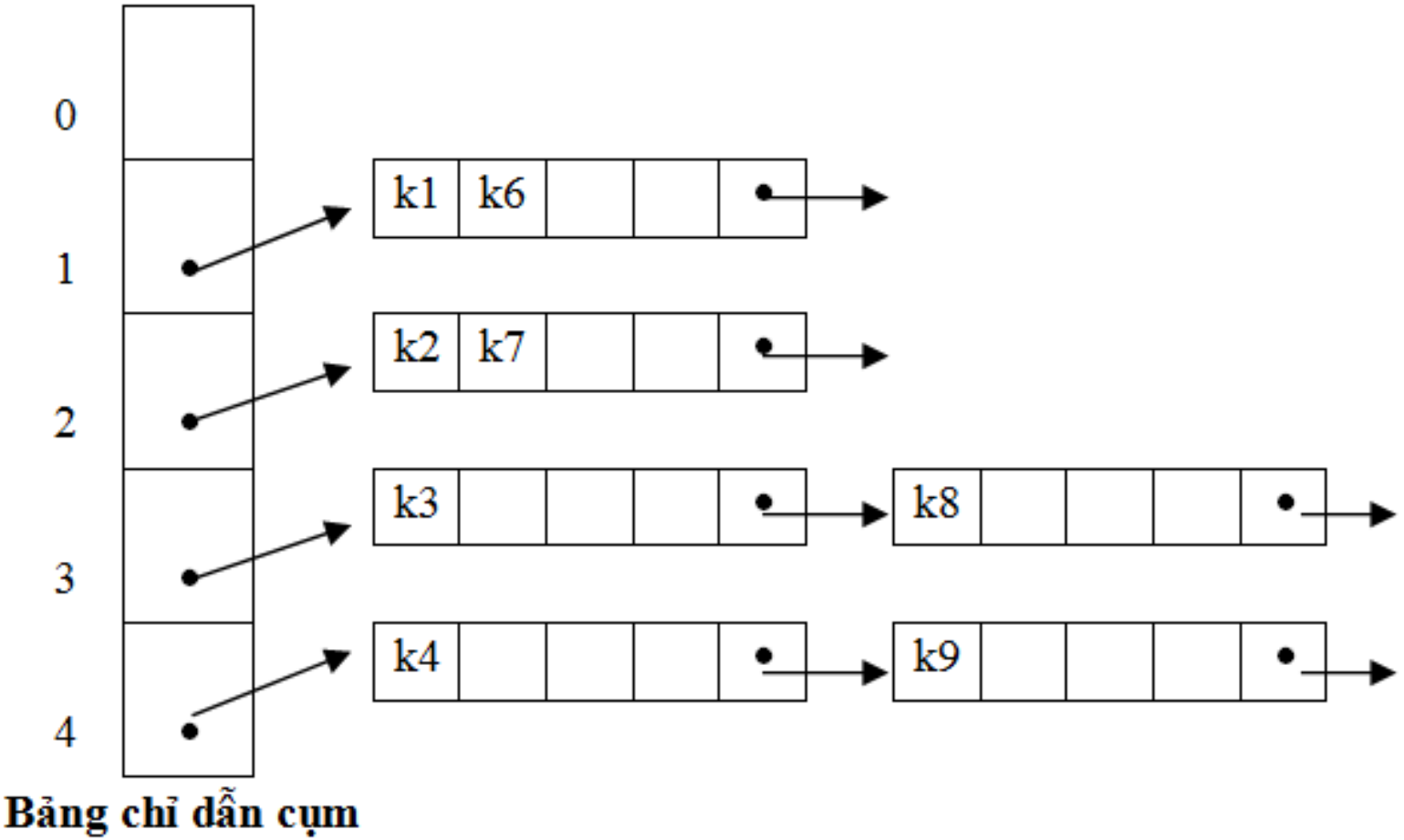
Xóa bản ghi có giá trị khóa là 64

(c)

Tổ chức tệp băm (*Hash File*)

- Tổ chức tệp dữ liệu
  - Phân chia các bản ghi vào các **cụm**
  - Mỗi cụm gồm một hoặc nhiều **khối**
  - Mỗi khối chứa số **lượng bản ghi cố định**
  - Tổ chức lữu trữ dữ liệu **trong mỗi cụm** áp dụng theo **tổ chức đống**
- Mục đích
  - Sử dụng chỉ số để hạn chế số lượng phép truy xuất đĩa bằng các phân nhóm các bản ghi (giả thiết n nhóm)
  - *Mapping* giá trị khoá với vị trí của (nhóm) bản ghi tương ứng

Tổ chức tệp băm (*Hash File*) …



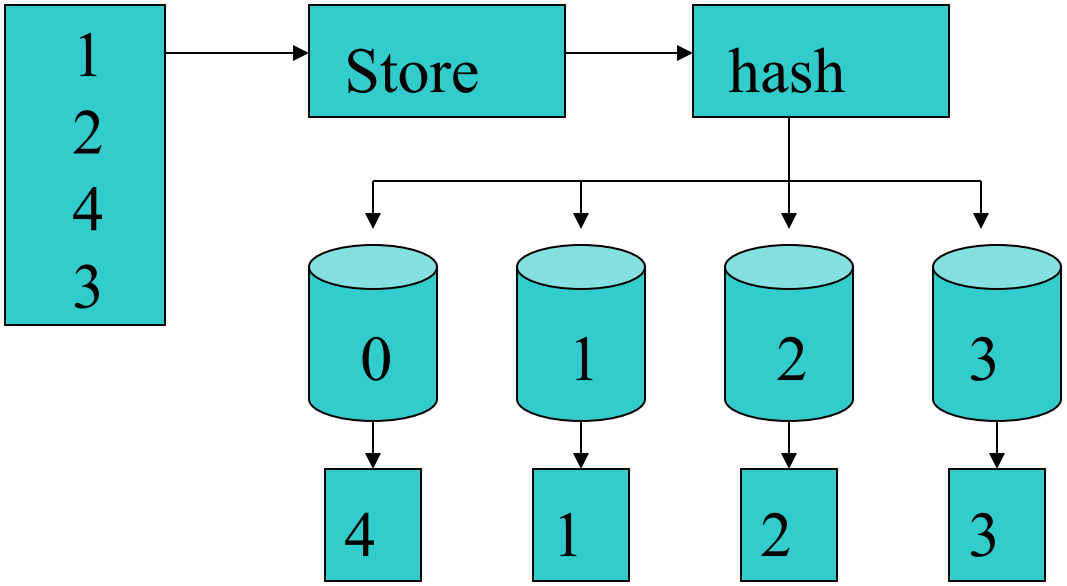
Tổ chức tệp băm (*Hash File*) …

- Dựa trên bảng băm (*hash table*)
  - Hàm băm (*hash function*)
  - Cụm (*bucket*)
- Hàm băm: h(x) nhận một giá trị trong đoạn [0,k-1], ví dụ: h(x)=x mod k

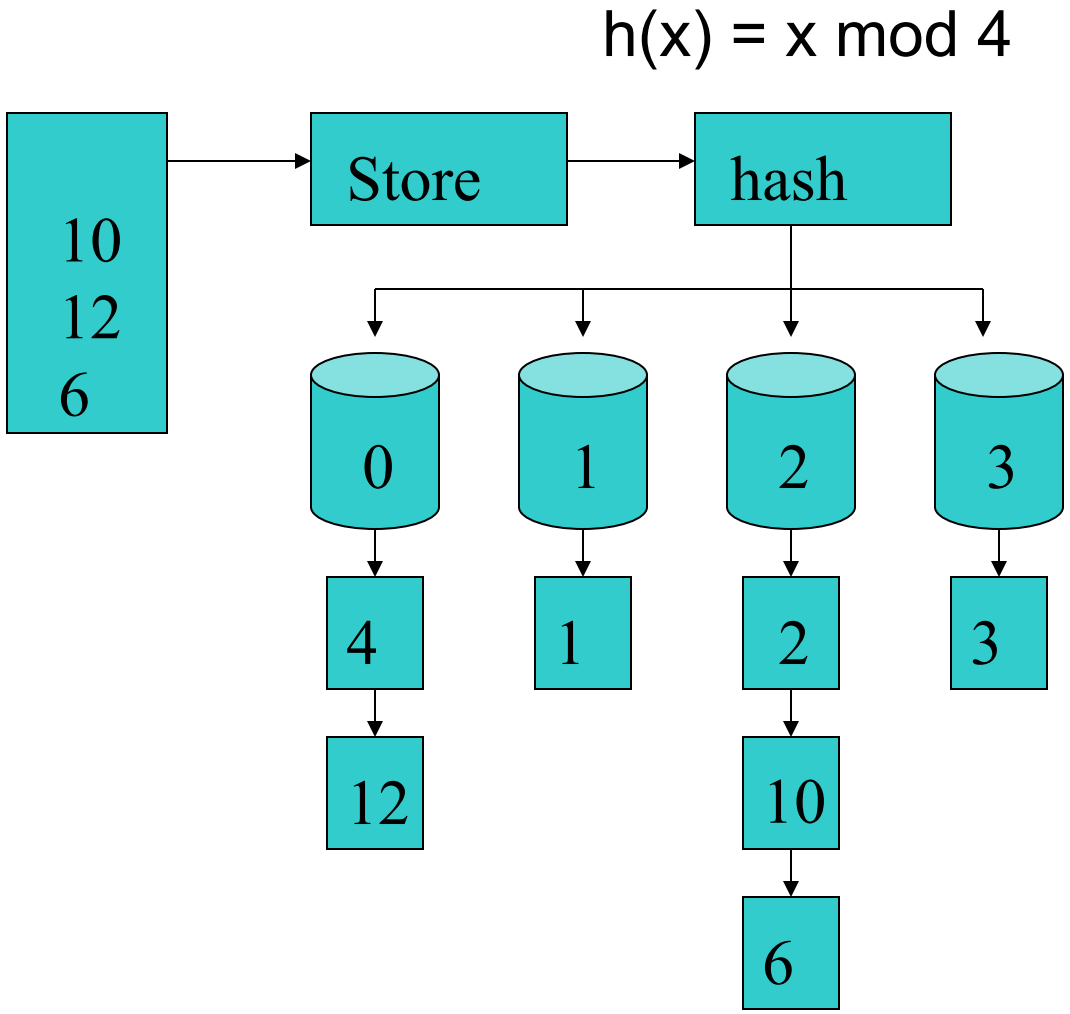
  → k cụm
- Tiêu chí chọn hàm băm: phân bố các bản ghi tương đối đồng đều theo các cụm

Ví dụ

h(x) = x mod 4



Ví dụ tiếp



Các phép toán

- Tìm kiếm 1 bản ghi có khóa x
  - tính h(x) sẽ được cụm chứa bản ghi,
  - sau đó tìm kiếm theo tổ chức đống trong cụm

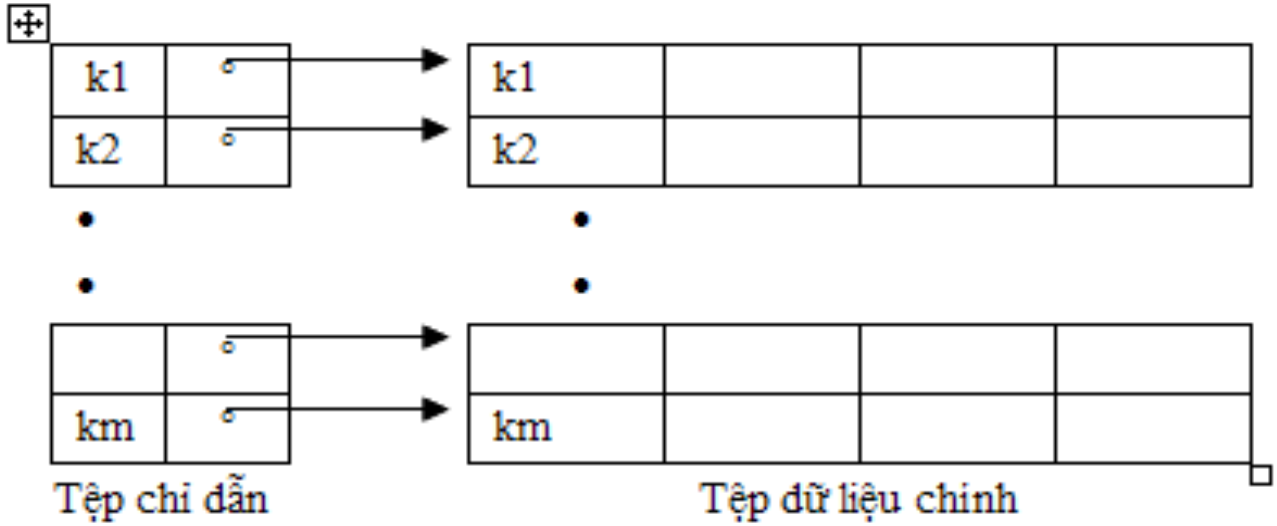
- Thêm 1 bản ghi có khóa x
  - Tìm kiếm
  - Đã tồn tại: bản ghi mới sai
  - Chưa tồn tại :
    - ghi vào khối đầu tiên còn chỗ trống trong cụm h(x)
    - Nếu không còn chỗ trống: thêm khối mới vào cuối cụm h(x), và ghi bản ghi mới vào khối này.

Các phép toán …

- Xoá 1 bản ghi có khóa x:
  - Tìm kiếm bản ghi có khóa x
  - Xóa bản ghi đó
  - Giải phóng khối nếu việc xóa bản ghi tạo ra khối trống
- Sửa đổi một bản ghi có khóa x:
  - Nếu sửa trên trường khóa: tìm kiếm → xóa , thêm mới bản ghi
  - Nếu sửa trên trường không khóa: tìm kiếm → cập nhật lại giá trị trên các trường

Tổ chức tệp chỉ dẫn (*Indexed File*)

- Tệp chỉ dẫn được xây dựng theo khoá được chọn trong các bản ghi
- Tệp chỉ dẫn bao gồm các cặp (k,d), trong đó k là giá trị của khoá của bản ghi đầu tiên, d là địa chỉ của khối (hay con trỏ khối)
- Giả sử tập dữ liệu chính có dữ liệu được sắp xếp theo khóa
- Tệp chỉ dẫn được sắp xếp theo giá trị của khoá



Tìm kiếm 1 bản ghi (trên tập chỉ dẫn hoặc trong khối)

- Tìm kiếm tuần tự
  - Duyệt tệp chỉ dẫn từ bản ghi đầu tiên đến khi tìm thấy bản ghi có khoá k cần tìm
  - Nhận xét
    - chậm đối với các tệp chỉ dẫn nói chung.
    - Thích hợp với các tệp chỉ dẫn nhỏ đủ để lưu ở bộ nhớ trong
- Tìm kiếm nhị phân
  - Chia đôi tệp chỉ dẫn đã sắp xếp để hạn chế số bản ghi cần duyệt
  - Tại mỗi lần chia hạn chế được ½ số bản ghi cần xem xét

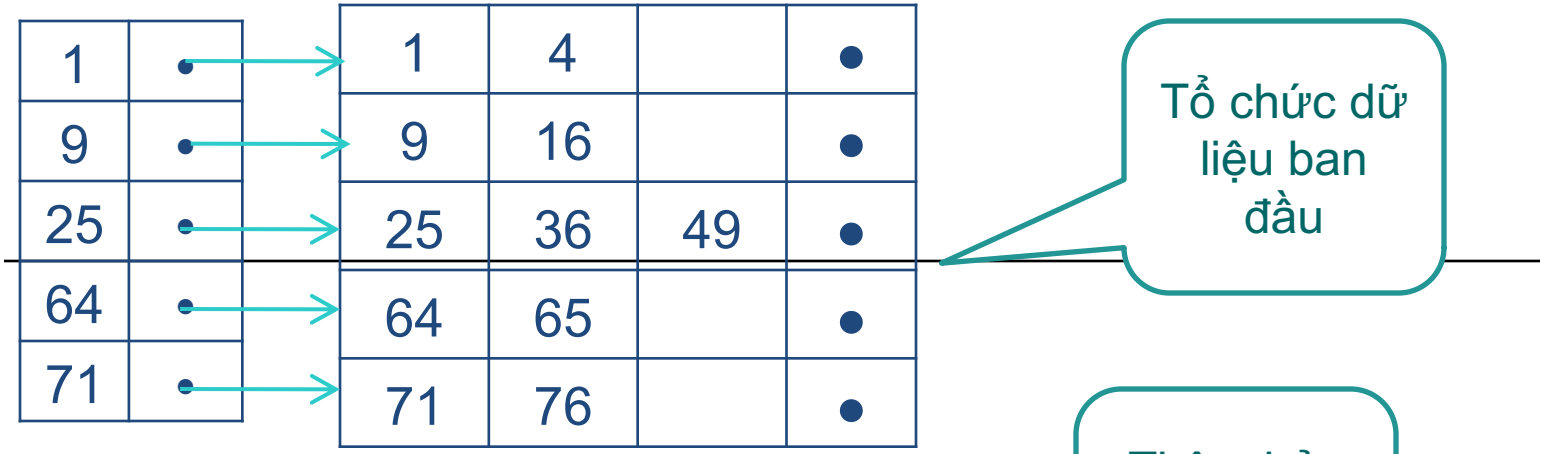
Các phép toán

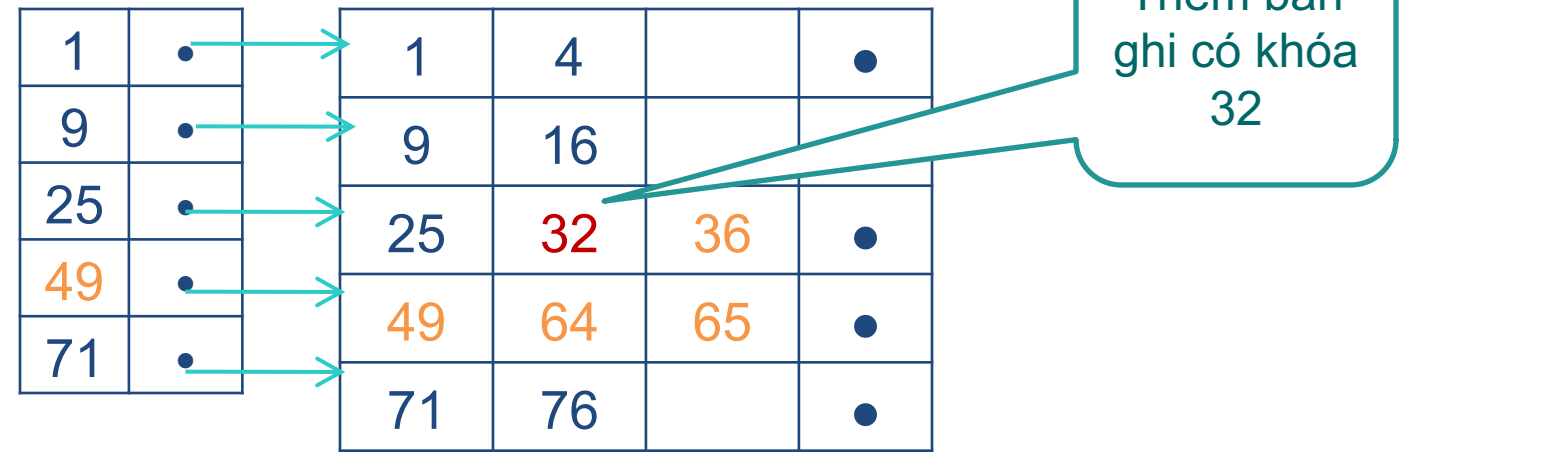
- Tìm kiếm 1 bản ghi:
  - Tìm kiếm trên tệp chỉ dẫn → khối chứa bản ghi
  - Tìm kiếm trên khối (tuần tự or nhị phân)
- Thêm 1 bản ghi có khóa K
  - Tìm trên tệp chỉ dẫn ra khối Bi sẽ chứa bản ghi đó
  - Nếu khối Bi còn chỗ trống: chèn bản ghi vào vị trí theo thứ tự sắp xếp của khóa →~ dịch các bản ghi khác trong Bi
    - nếu chèn vào vị trí đầu của Bi → cập nhật lại chỉ số trong file chỉ dẫn cho Bi)
    - Nếu chèn vào làm Bi hết chỗ → dịch bản ghi cuối của Bi sang đầu Bi+1 → cập nhật lại file chỉ dẫn cho Bi+1
  - Nếu khóa K lớn hơn tất cả các khóa khác và không còn Bi có chỗ trống → tạo khối mới & thêm 1 dòng vào file chỉ dẫn

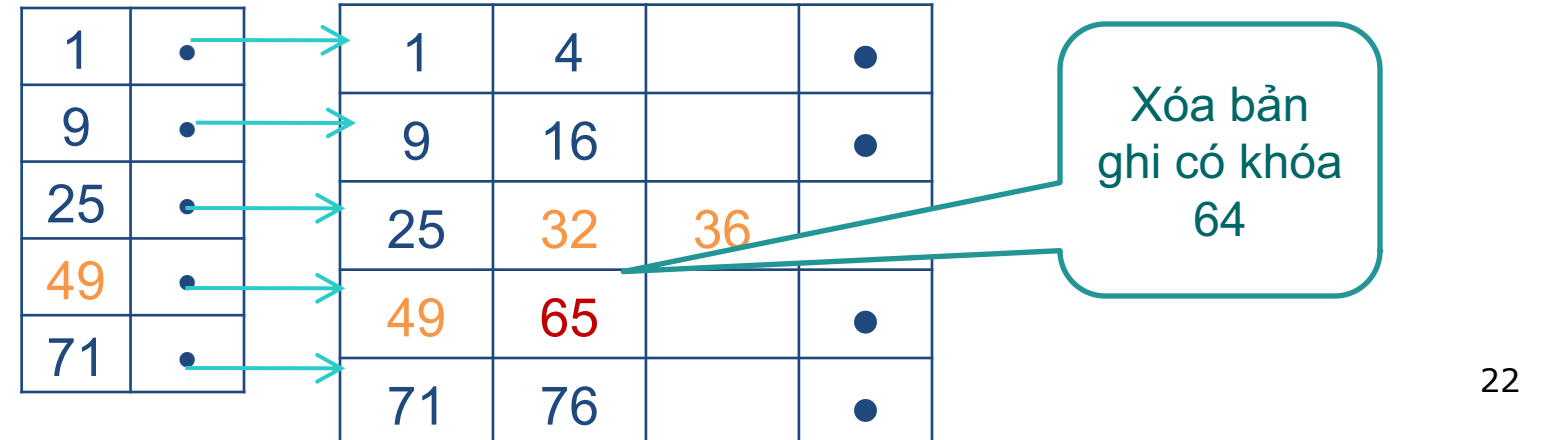
Các phép toán …

- Xoá 1 bản ghi:
  - Tương tự như khi thêm 1 bản ghi (dịch chuyển các bản ghi trong khối và update chỉ số file chỉ dẫn)
  - Nếu xóa tạo block rỗng thì xóa cả block

- Sửa đổi một bản ghi có khóa x
  - Tìm kiếm bản ghi cần sửa
  - Nếu trường cần sửa không tham gia vào khóa : cập nhật bản ghi
  - Nếu trường cần sửa tham gia vào khóa : xóa và thêm mới bản ghi


1
9
25
64
71
1 4
9 16
25 36 49
64 65
71 76
Tổ chức dữ liệu ban đầu


1
9
25
49
71
1 4
9 16
25 32 36
49 64 65
71 76
Thêm bản ghi có khóa 32

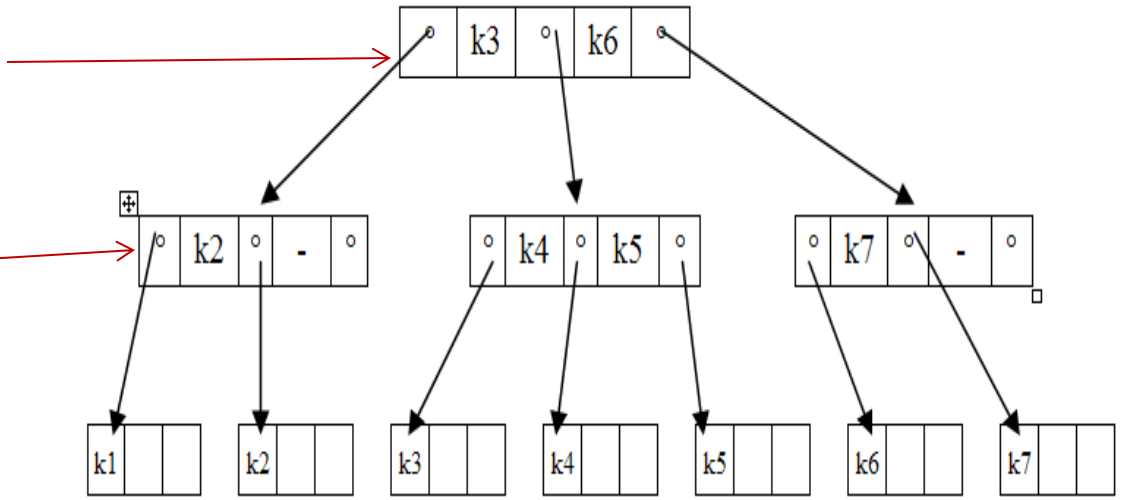

1
9
25
49
71
1 4
9 16
25 32 36
49 65
71 76
Xóa bản ghi có khóa 64

Cây cân bằng (*BalanceTree*)

- B-tree cân bằng được tổ chức theo cấp m, có các tính chất sau đây:
  - Gốc của cây hoặc là 1 nút lá hoặc ít nhất có 2 con
  - Mỗi nút (trừ nút gốc và nút lá) có từ ⌈m/2⌉ đến m con
  - Mỗi đường đi từ nút gốc đến bất kỳ nút lá nào đều có độ dài như nhau

1 nút lá
hoặc tối thiểu 2 con

từ ⌈m/2⌉ đến m con



Nhận xét

- Cấu trúc của mỗi nút trong B-tree

  (p0, k1, p1, k2,...,kn, pn)

  - pi (i=1..n) là con trỏ trỏ tới khối i của nút có ki là khoá đầu tiên của khối đó.
  - Các khoá k trong một nút được sắp xếp theo thứ tự tăng dần

- Mọi khoá trong cây con, trỏ bởi pi đều $<$ ki+1 (i = 0..n-1)
- Mọi khoá trong cây con, trỏ bởi pi đều $\geq$ ki (i = 1..n)

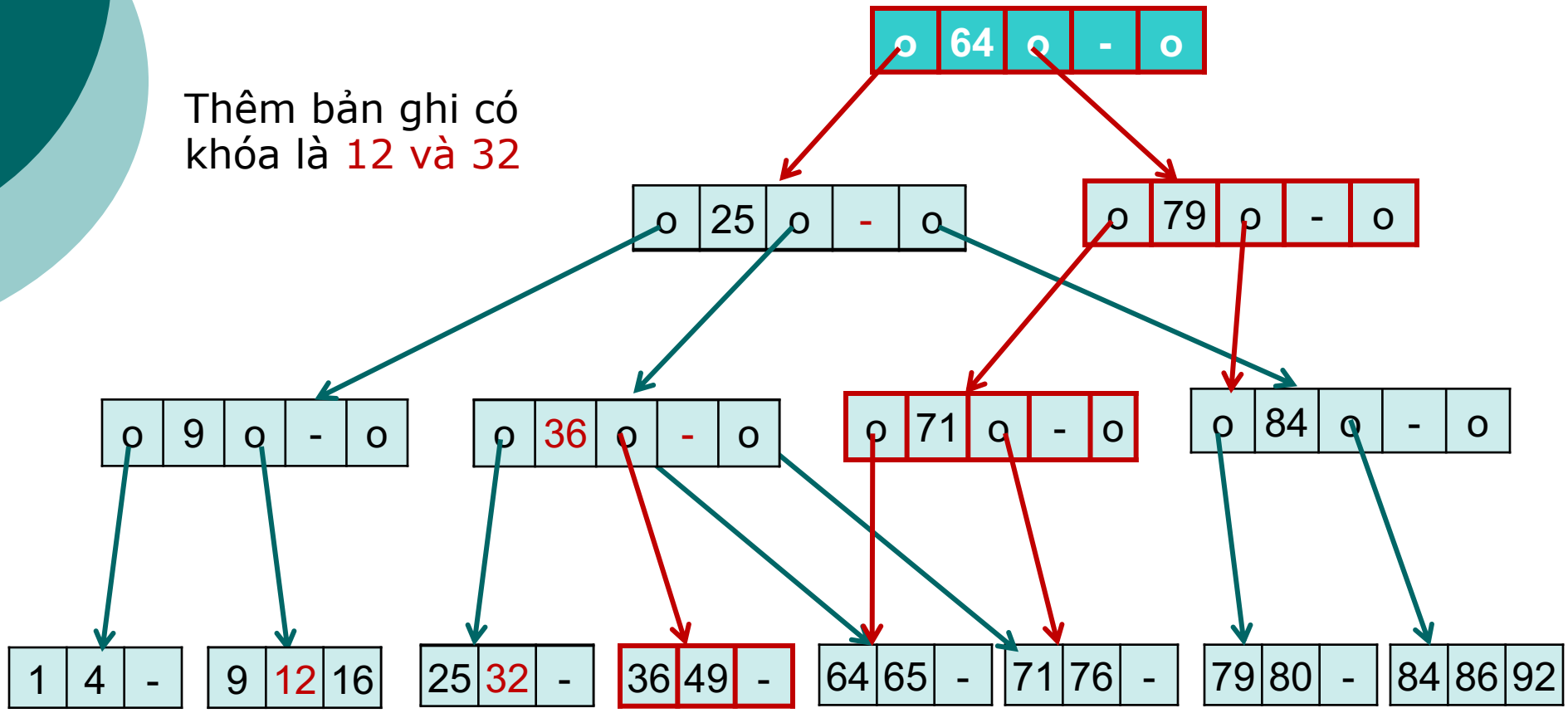
Các phép toán

- Tìm kiếm 1 bản ghi : duyệt từ nút gốc đến nút lá chứa bản ghi

- Thêm 1 bản ghi có khóa k:
  - Xác định vị trí chứa bản ghi, nút lá L
    - Nếu còn chỗ: thêm bình thường
    - Nếu hết chỗ → tạo nút lá mới L', chuyển nửa cuối DL của L sang L' và chèn bản ghi mới vào L hoặc L' tùy theo giá trị khóa k
  - ➔ có khả năng lan truyền đến nút gốc

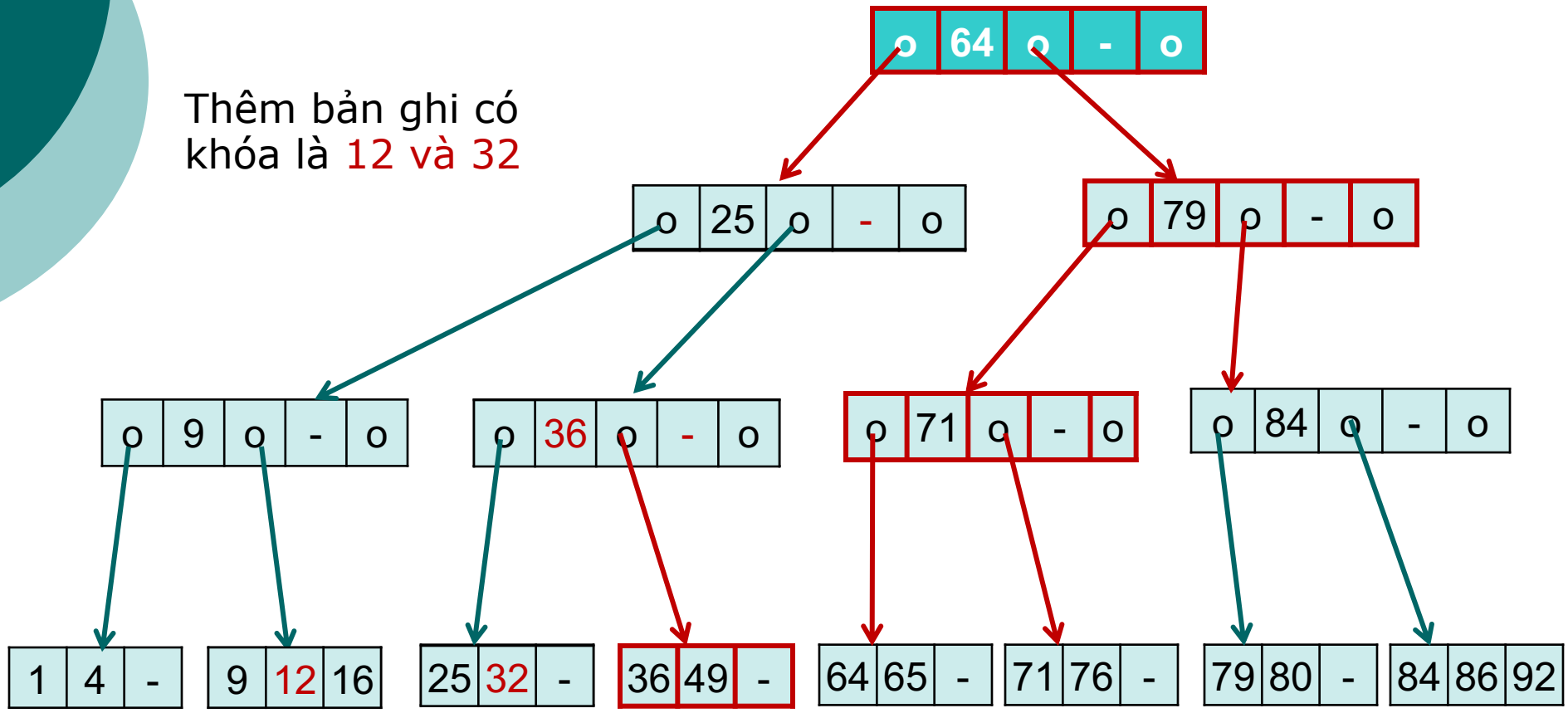
Các phép toán …

Thêm bản ghi có khóa là 12 và 32



Các phép toán …

Thêm bản ghi có khóa là 12 và 32



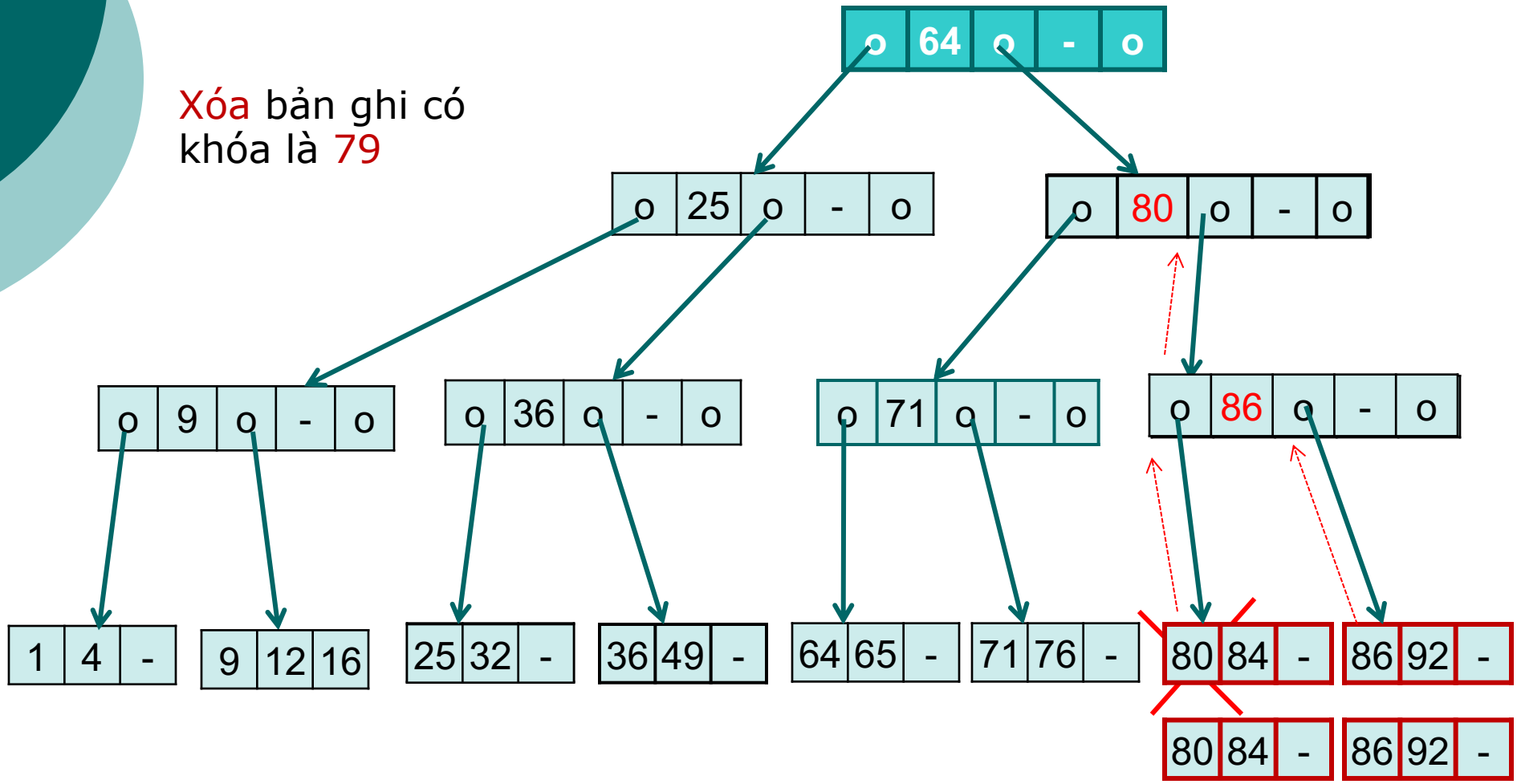
Các phép toán …

- Xoá 1 bản ghi
  - Tìm kiếm nút lá L chứa bản ghi
  - Loại bỏ bản ghi ra khỏi L
    - Nếu bản ghi bị loại là bản ghi đầu tiên của L → chỉnh khóa ở các nút trên cho tới gốc cây
    - Nếu việc xóa 1 bản ghi làm nút L có chứa số bản ghi ít hơn $\lceil m/2 \rceil$ → chỉnh lại các giá trị (pi, ki) ở các nút trên → có thể gộp 2 nút thành một / hai nút mới

  ➔ Có thể lan truyền đến tận gốc
- Sửa đổi một bản ghi:
  - Sửa các trường không tham gia khóa: cập nhật bình thường
  - Sửa trường có tham gia khóa → xóa và thêm mới

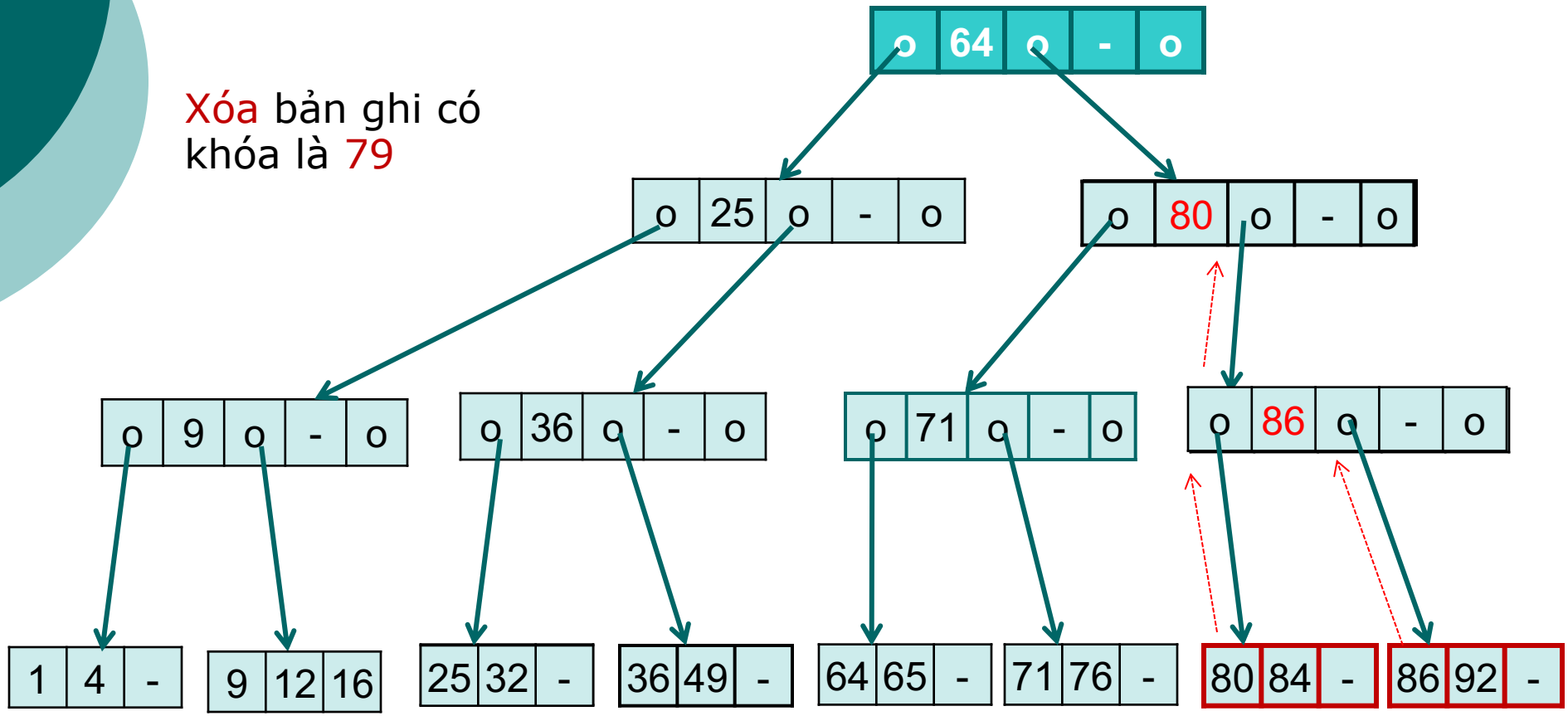
Các phép toán …

Xóa bản ghi có khóa là 79



Các phép toán …

Xóa bản ghi có khóa là 79



So sánh các cách tổ chức dữ liệu

- Tệp đống
  - thao tác đơn giản
  - tìm kiếm chậm
- Tệp băm
  - dựa trên 1 hàm băm, cho phép tìm thấy địa chỉ khoản mục dữ liệu một cách trực tiếp
  - hàm băm tốt? Phân bố các bản ghi đồng đều trong các cụm
- Tệp chỉ dẫn
  - được áp dụng phổ biến, với các ứng dụng yêu cầu cả xử lý tuần tự và truy nhập trực tiếp đến các bản ghi
  - hiệu năng sẽ giảm khi kích thước tệp tăng =>chỉ dẫn B-cây
- Cây cân bằng
  - phức tạp trong việc thêm, xóa, sửa dữ liệu

Kết luận

- Truy cập đến CSDL thường liên quan đến một phần nhỏ các bản ghi trong một tệp dữ liệu hay một vài trường (đặc biệt là các trường khoá) của các bản ghi dữ liệu.
  - Xác định các yêu cầu này cho phép thiết kế dữ liệu vật lý hiệu quả thông qua việc sử dụng các tổ chức lưu trữ đặc biệt

- Các cấu trúc chỉ dẫn được tạo lập trên khoá tìm kiếm để tăng hiệu quả của lưu trữ dữ liệu