

CHƯƠNG 5: B-Tree

1

1

MỤC TIÊU

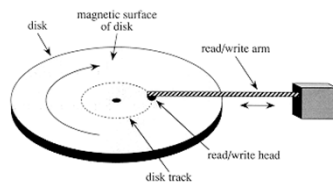
- 📖 B-Cây
- 📖 Thêm phần tử vào B-Cây
- 📖 Xóa phần tử khỏi B-Cây

2

2

Giới thiệu

- ❖ B-cây (B-Trees) xử lý tốt trên đĩa hoặc các thiết bị lưu trữ ngoài

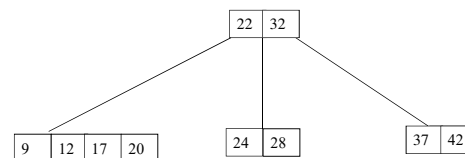


3

3

B-Cây

- ❖ Mỗi node có thể chứa n khóa, các khóa này sắp xếp theo thứ tự tăng dần
- ❖ Mỗi node trong có $n + 1$ con trỏ tới các node con
- ❖ Node lá không có node con và có cùng mức



4

4

B-Cây

- ❖ Một B-Cây bậc t ($t \geq 2$)
 - Mỗi node (trừ node gốc) có tối thiểu t khóa và có tối thiểu t con
 - Mỗi node có tối đa $2t$ khóa, vì vậy một node trong có tối đa $2t + 1$ con
 - Một node gọi là đầy khi chứa $2t$ khóa
- ❖ Ví dụ
 - B-tree đơn giản nhất có $t = 2$, mỗi node trong có hoặc 3, 4, 5 con

5

5

Thêm phần tử

- ❖ Thao tác thêm phần tử (Insert): thêm 1 khóa x vào cây
 - Thêm v vào 1 nút lá
 - Nếu nút lá đầy: tách nút lá ra làm đôi, và chuyển phần tử giữa lên nút cha

6

6

Thêm phần tử

- ❖ Thêm các phần tử sau vào B-cây bậc $t = 2$
 22; 42 12 32 17; 37 9 28 20 24; 7; 44 15 48 29 10 34; 40 26 47 27;
 - Thêm 22

22

 - Thêm 42

22	42
----	----
 - Thêm 12

12	22	42
----	----	----
 - Thêm 12

12	22	32	42
----	----	----	----

7

7

Thêm phần tử

- ❖ 22; 42 12 32 **17**; 37 9 28 20 24; 7; 44 15 48 29 10 34; 40 26 47 27;

12	22	32	42
----	----	----	----

 - Thêm 12
 - Thêm **17** thì trang này bị đầy (một trang B-cây cấp 2 chứa tối đa bốn phần tử).
 - Dãy các khóa là 12 17 22 32 42 nên trang này bị tách ra thành hai trang:
 - Một trang chứa khóa (12 17) và một trang chứa khóa (32 42), và cấp phát thêm trang mới chứa khóa 22 (trang cha).

8

8

Thêm phần tử

❖ 22; 42 12 32 **17**; 37 9 28 20 24; 7; 44 15
48 29 10 34; 40 26 47 27;
➤ Thêm **17**

9

9

Thêm phần tử

❖ 22; 42 12 32 17; **37 9 28 20 24**; 7; 44 1
5 48 29 10 34; 40 26 47 27;
➤ Thêm **37 9 28 20 24**

10

10

Thêm phần tử

❖ 22; 42 12 32 17; 37 9 28 20 24; 7; 44 15
48 29 10 34; 40 26 47 27;
❖ Kết quả thêm 24

11

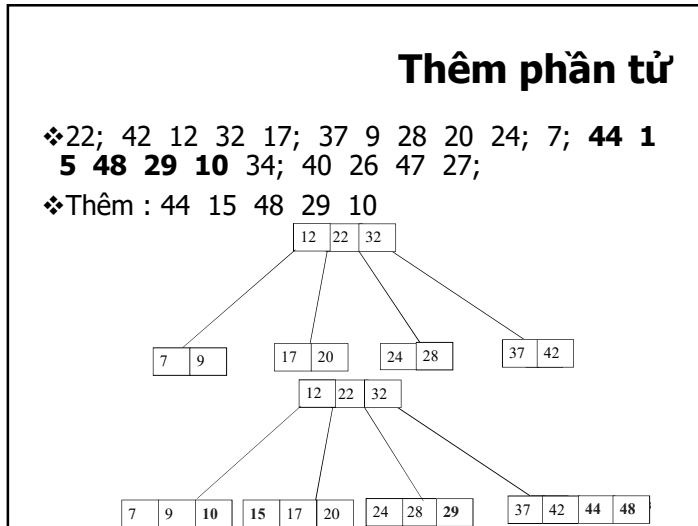
11

Thêm phần tử

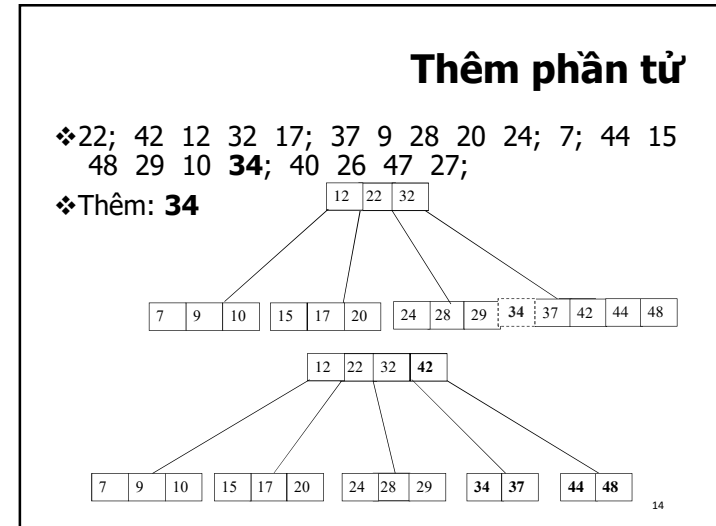
❖ 22; 42 12 32 17; 37 9 28 20 24; **7**; 44 15
48 29 10 34; 40 26 47 27;
❖ Thêm **7** -> **Phạm**

12

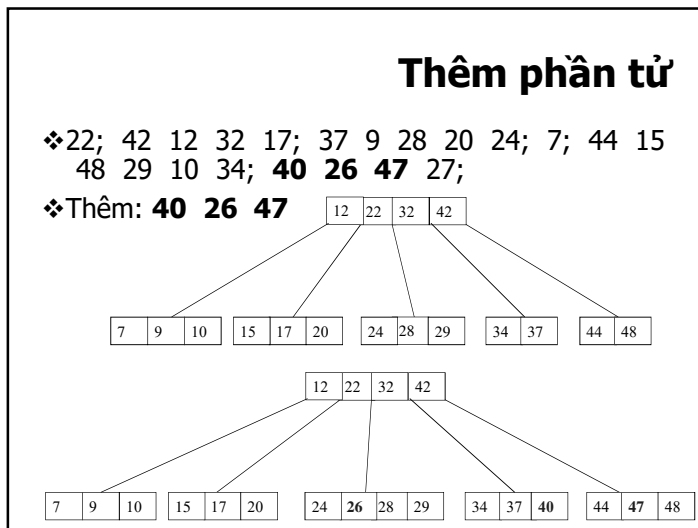
12



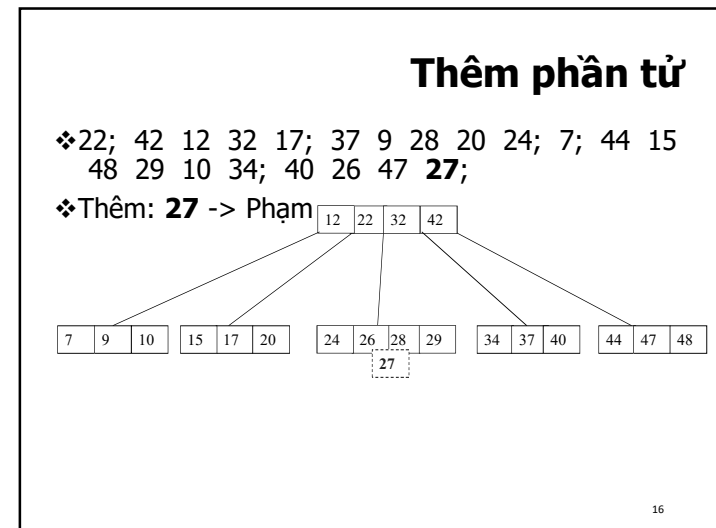
13



14



15

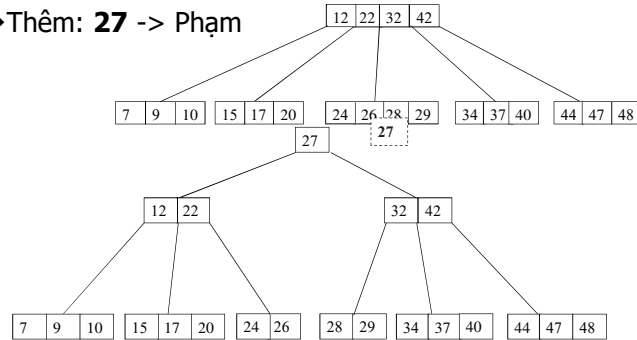


16

Thêm phần tử

❖ 22; 42 12 32 17; 37 9 28 20 24; 7; 44 15
48 29 10 34; 40 26 47 **27**;

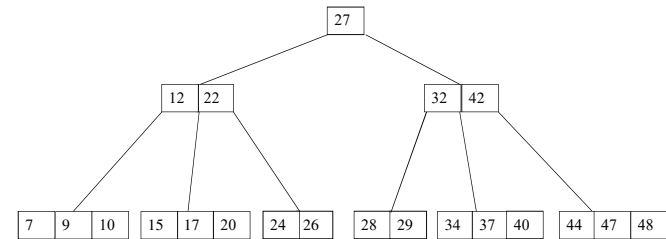
❖ Thêm: **27** -> Phạm



17

Xóa phần tử

❖ Xóa thứ tự các nút có khóa:
27 47 26; 40 34; 10 29 48 15 44; 7 24 20
28; 9 37 17; ở cây dưới

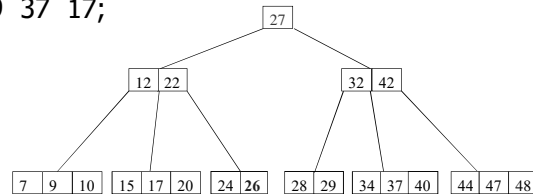


18

18

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 44; 7 24 20
28; 9 37 17;



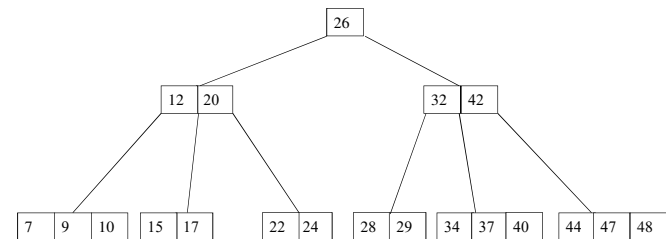
❖ Xóa 27 ta mang 26 lên thay thế, khi ấy nút 24, 26 chỉ còn lại 24 (phạm). Khi đó ta mượn trang trái 1 khóa đẩy lên cha, và lấy 22 của cha xuống

19

19

Xóa phần tử

❖ **27** 47 26; 40 34; 10 29 48 15 44; 7 24 20
28; 9 37 17;

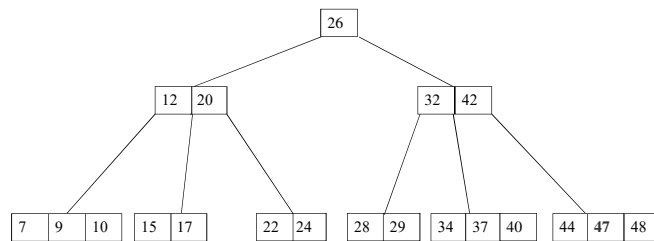


20

20

Xóa phần tử

❖ 27 47 26;40 34; 10 29 48 15 44; 7 24 20 28; 9 37 17;

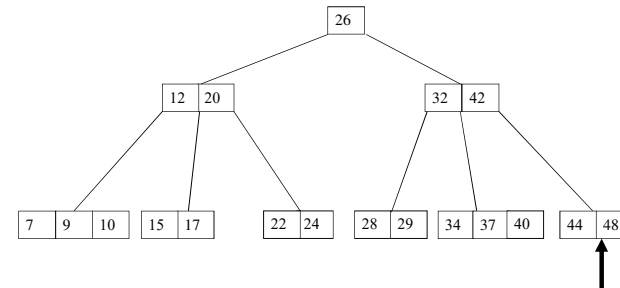


21

21

Xóa phần tử

❖ 27 47 26;40 34; 10 29 48 15 44; 7 24 20 28; 9 37 17;

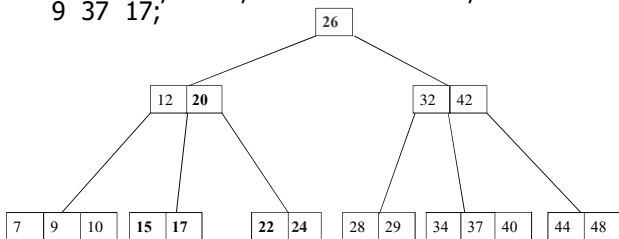


22

22

Xóa phần tử

❖ 27 47 **26**;40 34; 10 29 48 15 44; 7 24 20 28; 9 37 17;



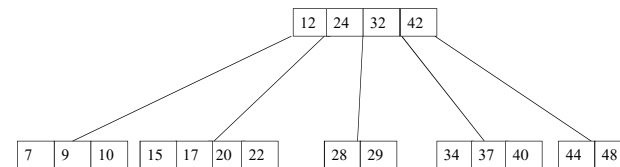
❖ Xóa 26, lấy 24 lên thay thế, lúc đó nút 22, 24 còn 1 khóa (phạm). Khi này ta nhập 15, 17, 20, 22 thành 1 nút. Và khi ấy cha chỉ còn 12 (phạm). Ta tiếp tục gộp 12, 26, 32, 42 lại

23

23

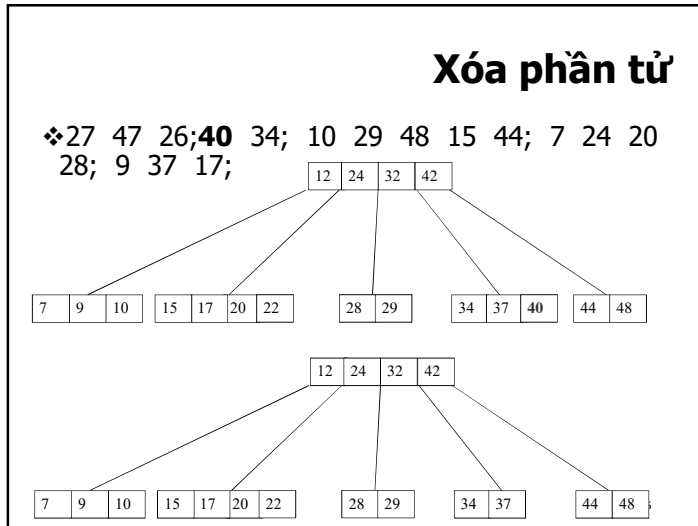
Xóa phần tử

❖ 27 47 **26**;40 34; 10 29 48 15 44; 7 24 20 28; 9 37 17;

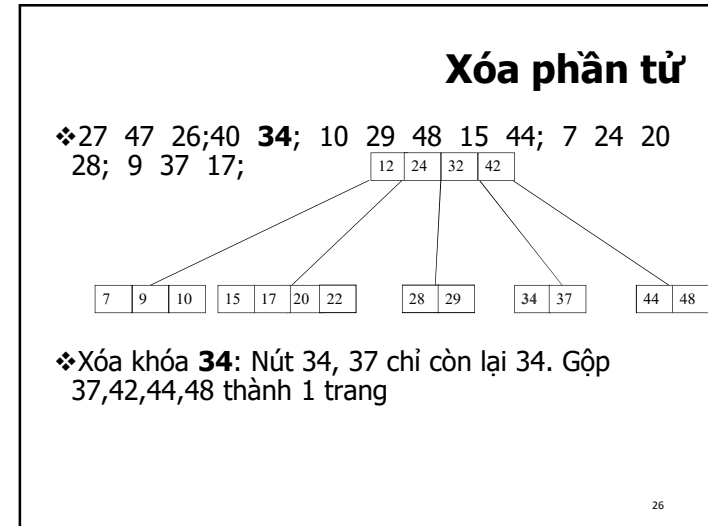


24

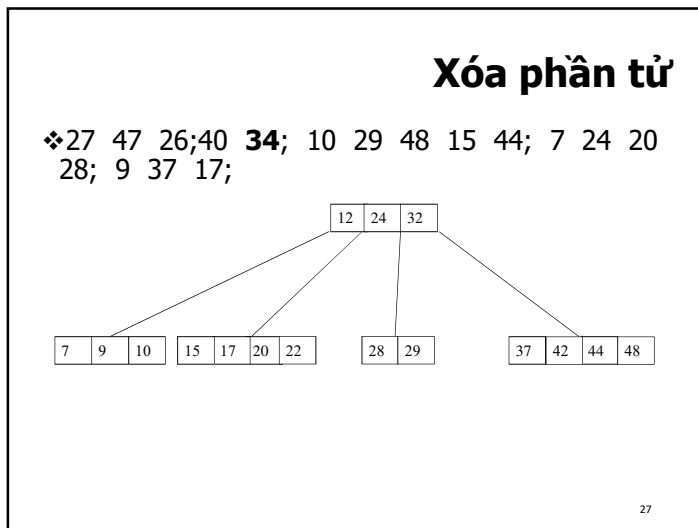
24



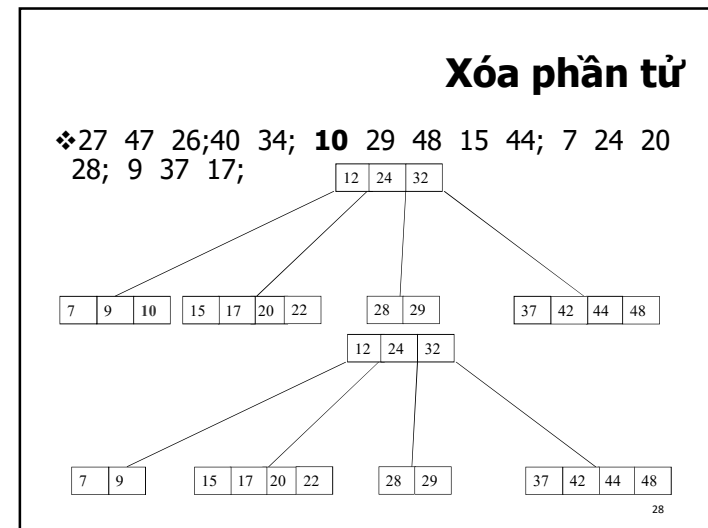
25



26



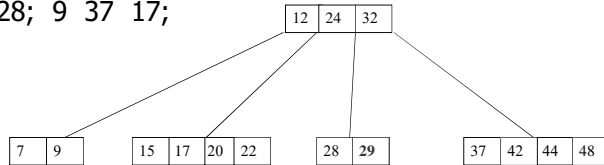
27



28

Xóa phần tử

❖ 27 47 26; 40 34; 10 **29** 48 15 44; 7 24 20
28; 9 37 17;

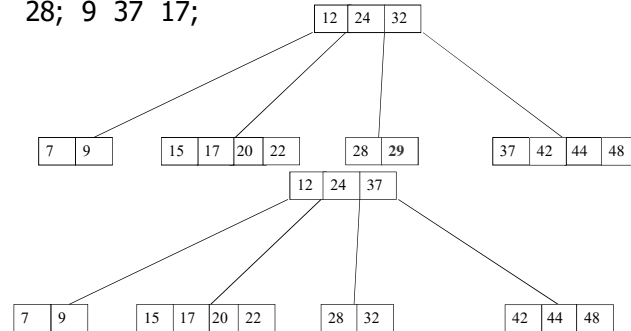


❖ Xóa 29, trang 28, 29 còn lại 28 (phạm), mượn 37 trang phải. Do đó 37 lên cha, 32 xuống thế 29

29

Xóa phần tử

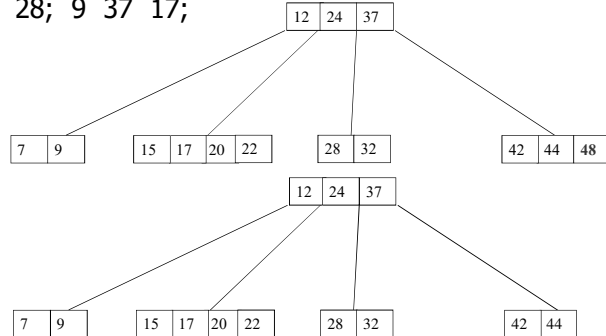
❖ 27 47 26; 40 34; 10 **29** 48 15 44; 7 24 20
28; 9 37 17;



30

Xóa phần tử

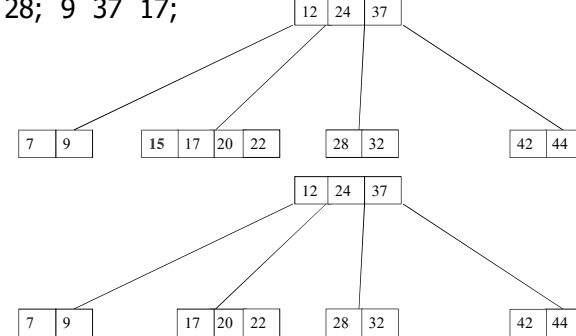
❖ 27 47 26; 40 34; 10 29 **48** 15 44; 7 24 20
28; 9 37 17;



31

Xóa phần tử

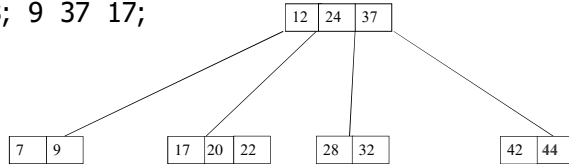
❖ 27 47 26; 40 34; 10 29 48 **15** 44; 7 24 20
28; 9 37 17;



32

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 **44**; 7 24 20
28; 9 37 17;

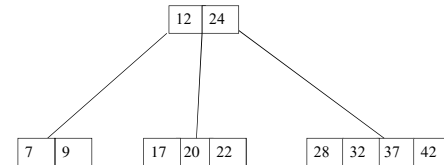
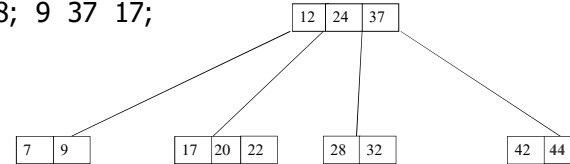


❖ Xóa 44 (phạm), gộp trang phải lại

33

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 **44**; 7 24 20
28; 9 37 17;



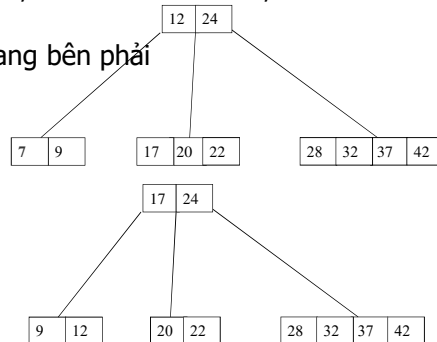
34

34

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 44; **7** 24 20
28; 9 37 17;

❖ Xóa 7, mượn trang bên phải

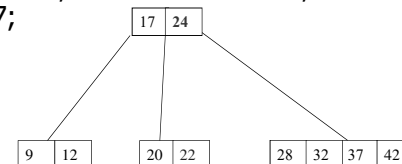


35

35

Xóa phần tử

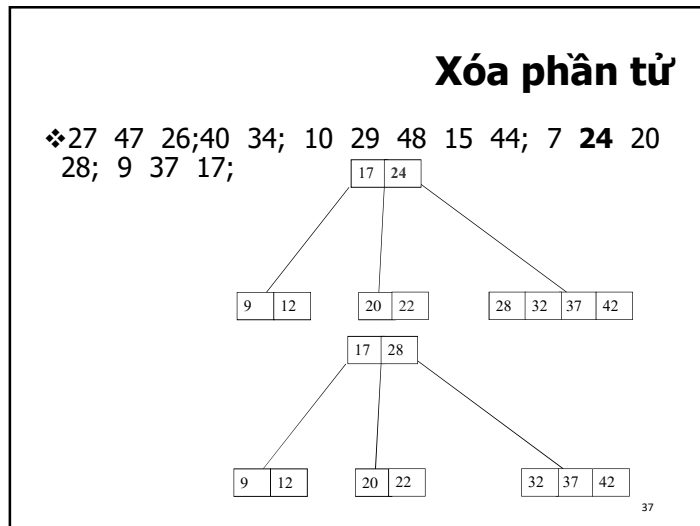
❖ 27 47 26; 40 34; 10 29 48 15 44; 7 **24** 20
28; 9 37 17;



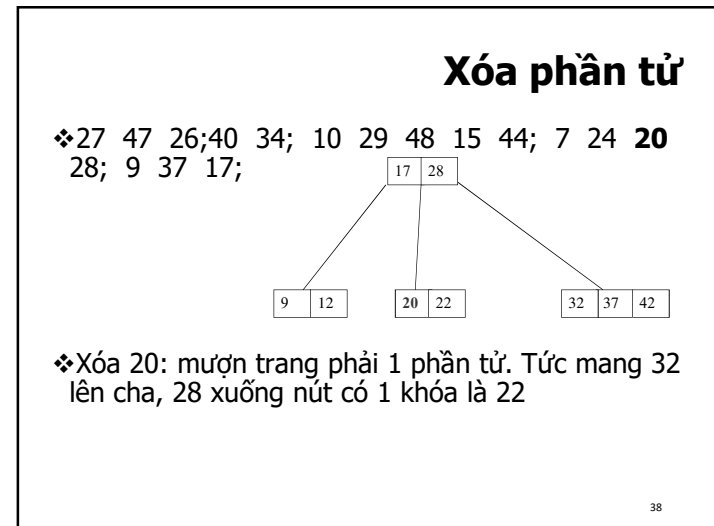
❖ Xóa 24, ta đem khóa 22 lên thay thế. khi đó trang 20, 22 chỉ còn 20 (phạm), Nên ta phải đem khóa 22 trở lại trang 20, 22. Mang 28 lên thêm

36

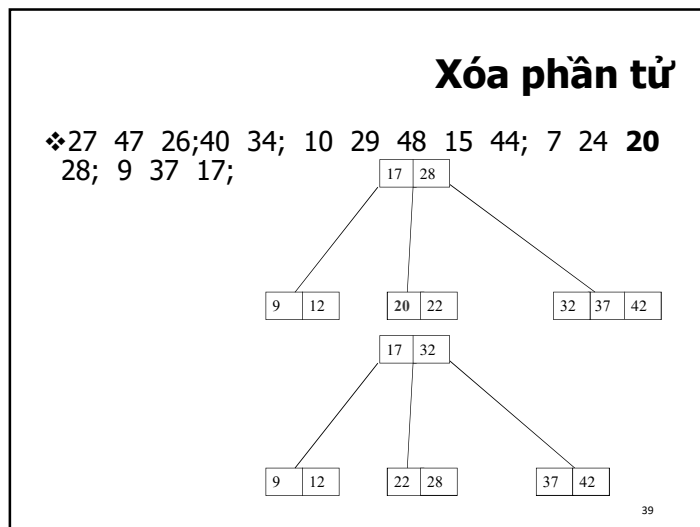
36



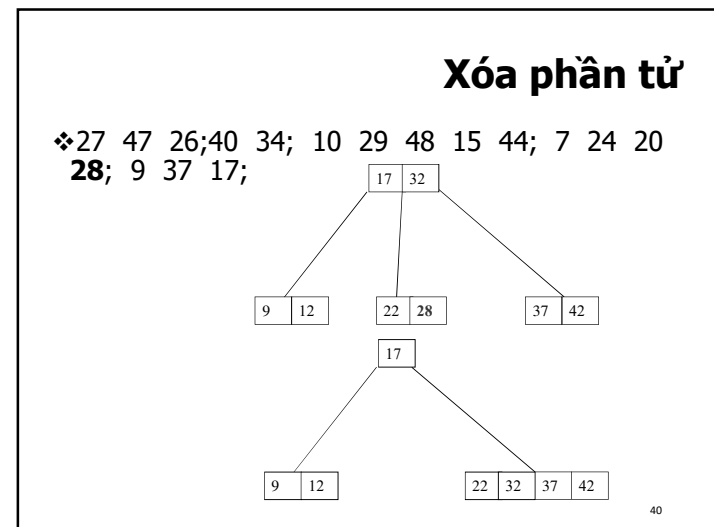
37



38



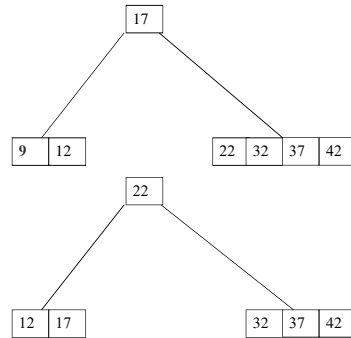
39



40

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 44; 7 24 20
28; **9** 37 17;

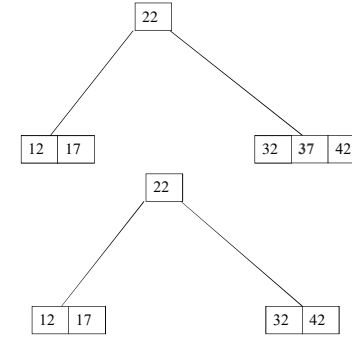


41

41

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 44; 7 24 20
28; 9 **37** 17;

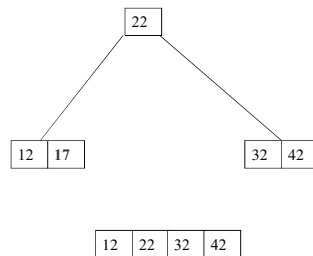


42

42

Xóa phần tử

❖ 27 47 26; 40 34; 10 29 48 15 44; 7 24 20
28; 9 37 **17**;



43

43

Bài tập

❖ Bài 1: hãy mô tả quá trình thêm các node sau và B-cây cấp 2: 10, 20, 60, 45, 05, 70, 15, 25, 50, 40, 35, 90, 80, 45, 55, 65, 95, 85, 06, 46, 69, 54, 32.

❖ Bài 2: Giả sử đã tạo B-cây cấp 2 ở bài 1, hãy xóa các khóa theo thứ tự khóa được thêm vào sau sẽ được xóa trước

44

44

Bài tập làm thêm

❖ Bài 3: quản lý B-cây cấp 2, yêu cầu:

- a. Khai báo cấu trúc B-cây cấp 2
- b. Viết thủ tục tìm một khóa trong B-cây trên
- c. Viết thủ tục thêm một khóa vào B-cây cấp 2 trên
- d. Viết thủ tục xóa một khóa trong B-cây cấp 2 trên

45

45

Tài liệu tham khảo

- [1] Truong, Le Xuan. Cấu trúc dữ liệu. Trường ĐH Mở Tp.HCM, 2015 (chapter 4).
- [2] Cormen, Thomas H., et al. *Introduction to algorithms*. MIT press, 2009 (III.11).
- [3] Drozdek, Adam. *Data Structures and algorithms in C++*. Cengage Learning, 2012 (chapter 10).

46

46