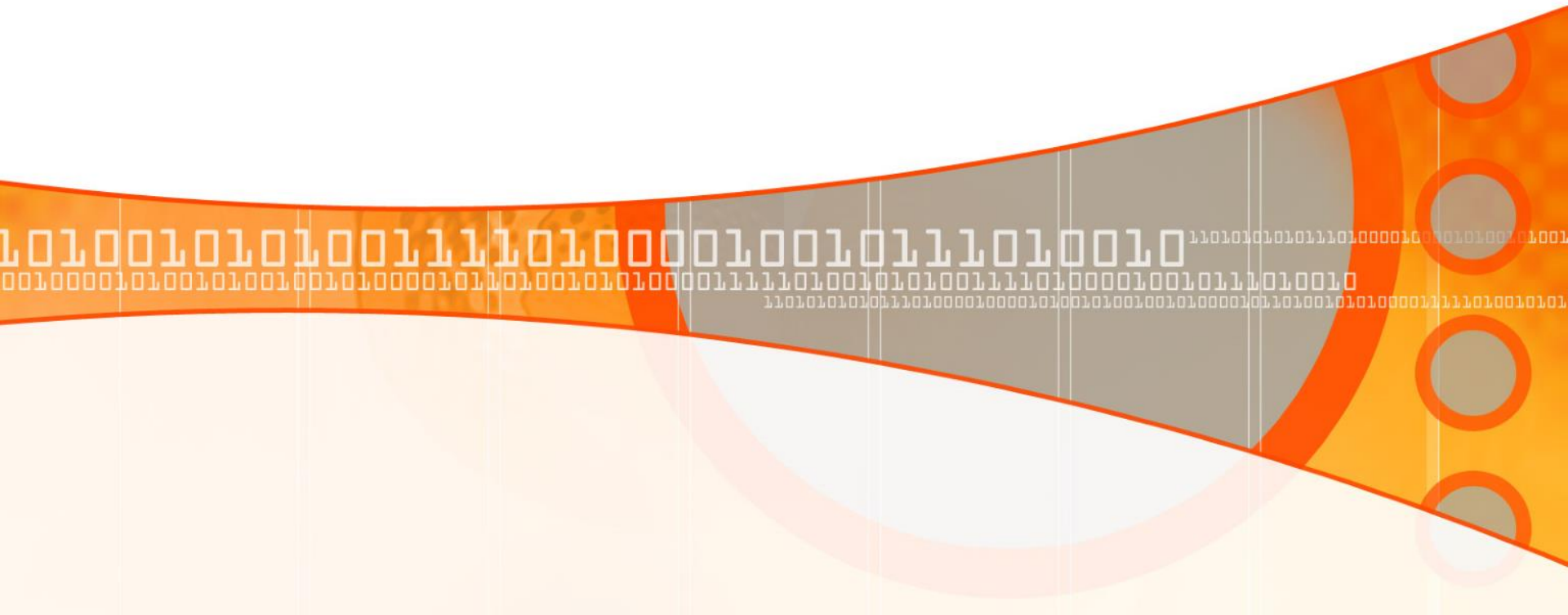


# STRUCTURI DE DATE

## Arbori B



# ARBORI B

**Sisteme de Gestiune a Bazelor de Date  
Relaționale (SGBDR): operatie importanta  
regasirea rapida a datelor – indecsi.**

**Indexul: colecție de perechi <valoare cheie,  
adresa articol> pentru a facilita accesul la o  
colecție de articole.**

**Structura de date foarte des folosită pentru  
implementarea indecșilor este arborele de  
căutare.**

# ARBORI B

Articolele memorate:

- oricât de complexe;
- conțin un câmp numit cheie pentru identificare.

Arborele de căutare bazat pe ordinea cheilor:  
**relație de ordine** totală pe  $C$  (mulțimea cheilor posibile ce vor trebui regăsite)

# ARBORI B

**Arbori de căutare bazați pe ordinea cheilor:**

- **arbori binari de căutare: o singură cheie asociată fiecărui nod;**
- **arbori multicăi de căutare: mai multe chei asociate fiecărui nod.**

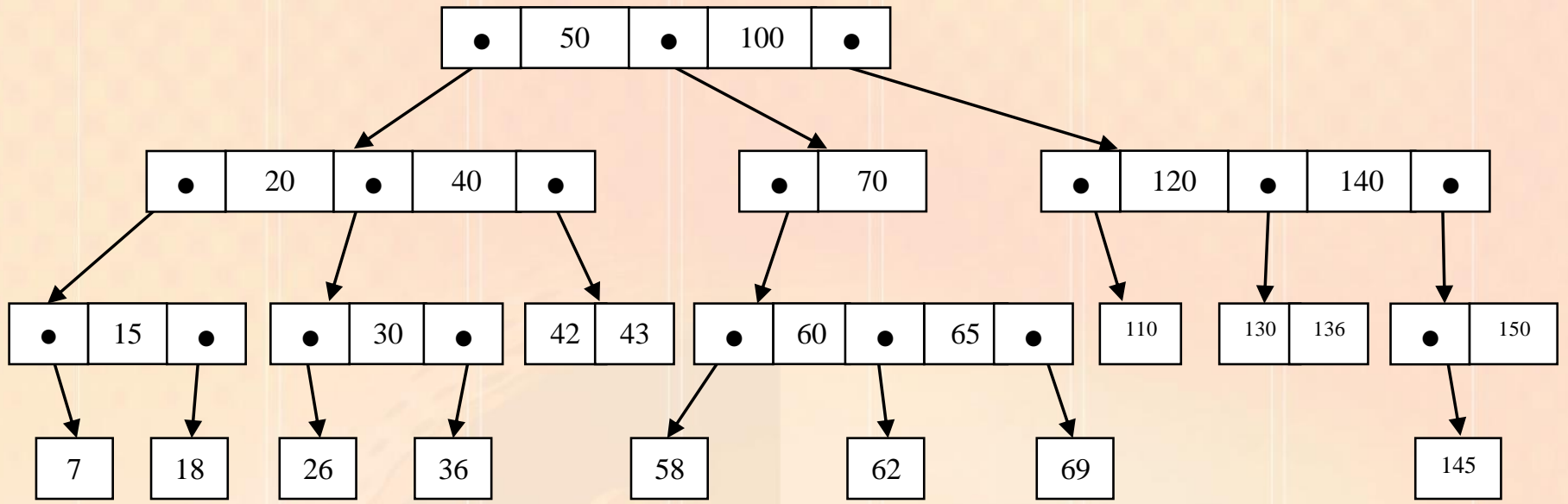
**Performanțele unui index: factorul de ramificare a arborelui de căutare folosit.**

# ARBORI B

## Arborii multicăi de căutare:

- generalizare a arborilor binari de căutare;
- nod oarecare: număr de **m** chei, ordonate strict crescător, ramificare în **m+1** subarbori;
- m diferă de la nod la nod;
- m între anumite limite pentru folosirea eficientă a mediului de stocare;
- m chei atașate unui nod: o pagină

# ARBORI B



Arbore multicăi de căutare de ordin 3



# ARBORI B

Arbore multicăi de căutare – **proprietăți:**

- Structura nod :

n	P <sub>0</sub>	K <sub>0</sub>	P <sub>1</sub>	K <sub>1</sub>	P <sub>2</sub>	...	P <sub>n-1</sub>	K <sub>n-1</sub>	P <sub>n</sub>
---	----------------	----------------	----------------	----------------	----------------	-----	------------------	------------------	----------------

- P<sub>0</sub>, P<sub>1</sub>, ..., P<sub>n</sub> pointeri către subarbori
- K<sub>0</sub>, K<sub>1</sub>, ..., K<sub>n-1</sub> valorile cheilor.

Număr de ramificații – restricția  $n \leq m - 1$ .

# ARBORI B

Arbore multicăi de căutare – **proprietăți**  
(continuare):

- Valorile cheilor într-un nod: ordine crescătoare;
- Valorile de chei din nodurile subarborelui  $P_i$  sunt mai mici decât valoarea cheii  $K_i$   
 $i = 1, 2, \dots, n-1$ ;



# ARBORI B

Arbore multicăi de căutare – **proprietăți**  
(continuare):

- Valorile de chei din nodurile subarborelui  $P_n$  sunt mai mari decât valoarea de cheie  $K_{n-1}$
- Subarborii  $P_i$ , sunt de asemenea arbori multicăi de căutare.

# ARBORI B

Arbore B de ordin  $m$  – proprietati:

- Arbore multicăi de căutare;
- Toate nodurile frunză sunt pe același nivel (arborele este echilibrat);
- Rădăcina: cel puțin doi descendenți, dacă nu este frunză (ramificare timpurie);
- Pagina conține cel puțin  $\lceil \frac{m}{2} \rceil$  chei; excepție rădăcina: mai puține chei, dacă este frunză (un nod este cel puțin 50% plin);

# ARBORI B

Arbore B de ordin **m** – proprietati  
(continuare):

- Un nod este frunză sau are  $n + 1$  descendenți, unde  $n$  este numărul de chei,  
 $\left\lceil \frac{m}{2} \right\rceil \leq n \leq m-1$ ;
- Pagina conține cel mult  $m-1$  chei; un nod poate avea maxim  $m$  descendenți.

# ARBORI B

## Operatii de baza:

### 1. Cautarea:

- Comparatie cheie cautata cu cheile nodului curent;
- Nodul de start: radacina
- Situatii de continuare a cautarii:

$c_i < x < c_{i+1}$  căutare în nodul  $P_i$ ;

$c_n < x$  căutare în  $P_n$ ;

$x < c_0$  căutare în  $P_0$ .

# ARBORI B

## Operatii de baza (continuare):

- **Lungimea maximă a drumului de căutare:  
înălțimea arborelui**



# ARBORI B

**Operatii de baza (continuare):**

**2. Inserarea a unei chei în arbore B:**

- **precedată de operația de căutare;**
- **cheie găsită în arbore: abandon operatie inserare;**
- **cheia nu a fost găsită: căutare terminata într-un nod frunză, unde se insereaza noua cheie;**



# ARBORI B

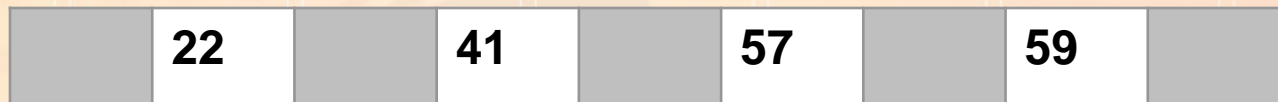
## Operatii de baza (continuare):

- **Situatii de inserare:**
  - nodul are mai puțin de  $m-1$  chei: inserare fără modificarea structurii arborelui B;
  - nodul are numărul maxim de  $m-1$  chei: “fisionare” nod; rezulta două noduri care se vor găsi pe același nivel și o cheie mediană care nu se va mai găsi în nici unul din cele două noduri.

# ARBORI B

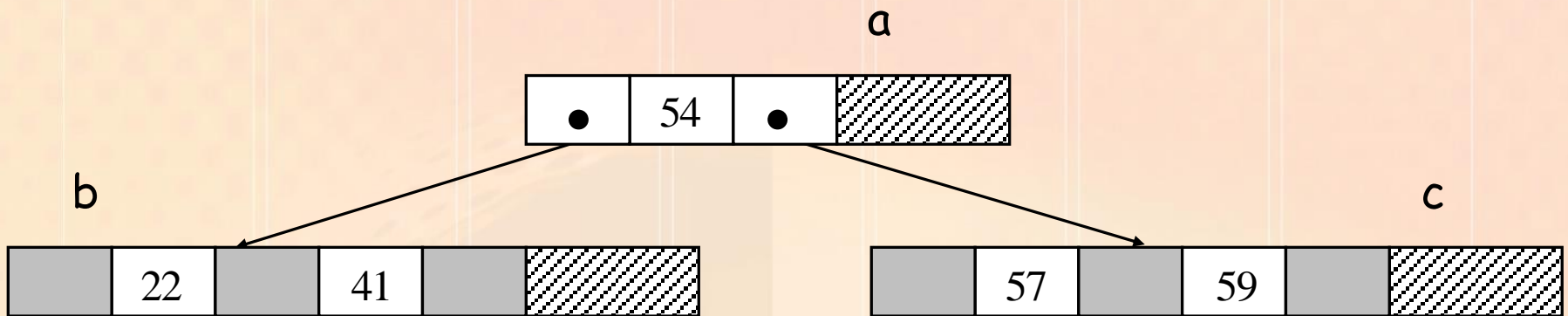
**Arbore B de ordin 5:**

- Numărul maxim de chei dintr-un nod este 4;
- Inserarea valorilor de cheie 22, 57, 41, 59.



# ARBORI B

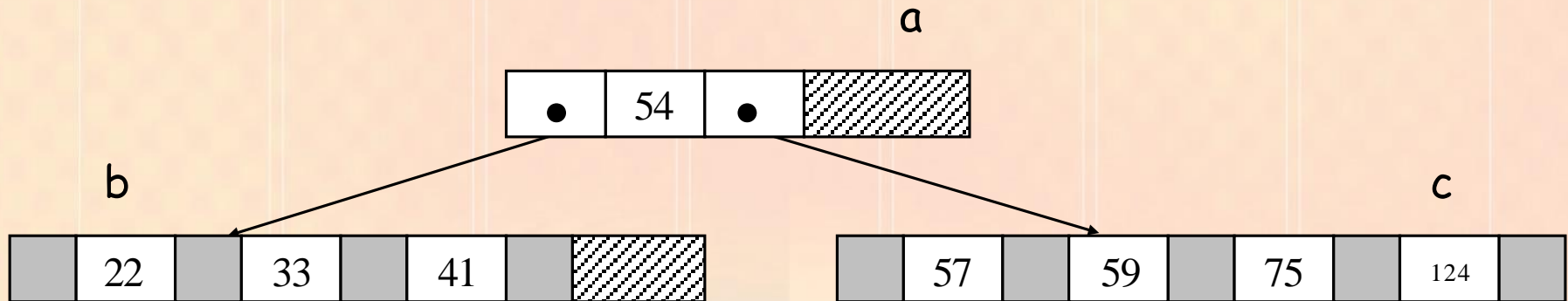
În urma inserării cheii 54, nodul rădăcină va conține prea multe chei, așa că el va “fisiona”



Formarea noii rădăcini a arborelui

# ARBORI B

Inserarea cheilor 33, 75, 124.



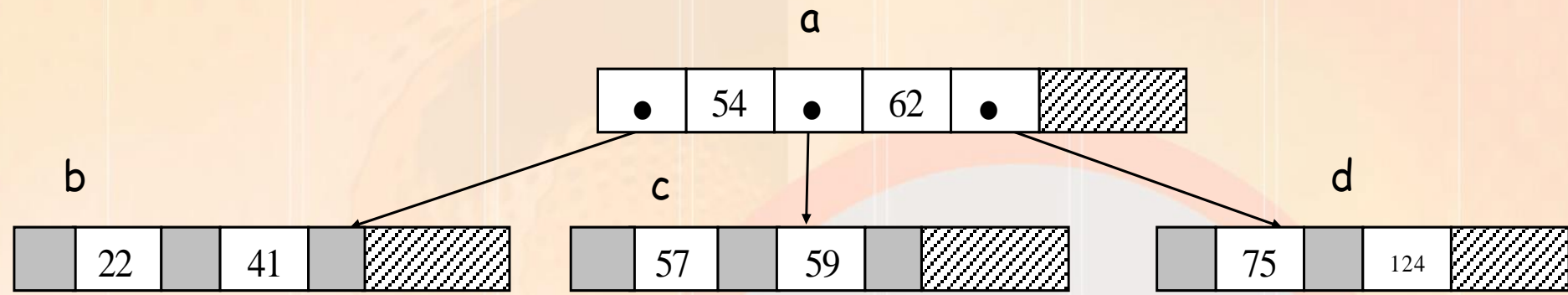
Structura arborelui după inserarea cheilor 33, 75, 124

# ARBORI B

## Inserarea cheii 62: divizarea nodului c

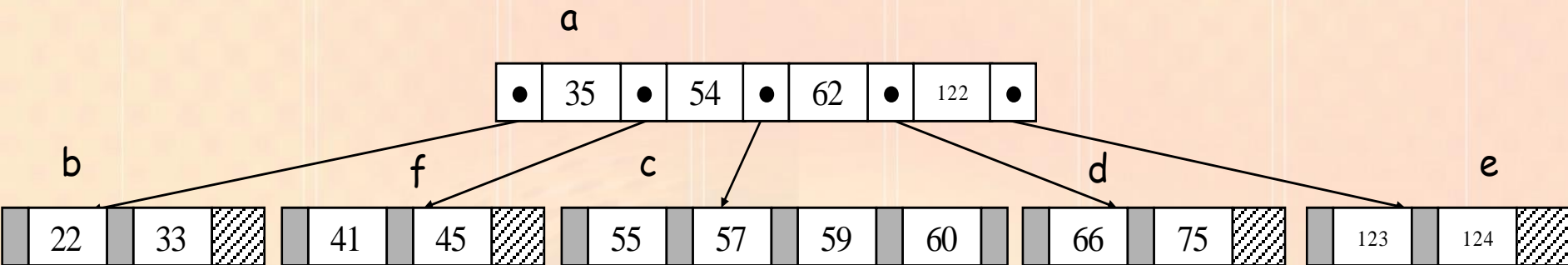


Cheia 62 promoveaza în nodul rădăcină



# ARBORI B

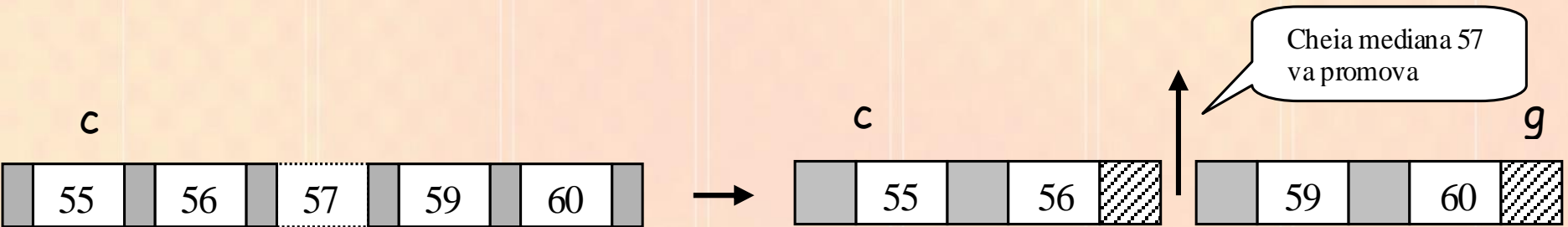
Inserarea cheilor 33, 122, 123, 55, 60, 45, 66, 35





# ARBORI B

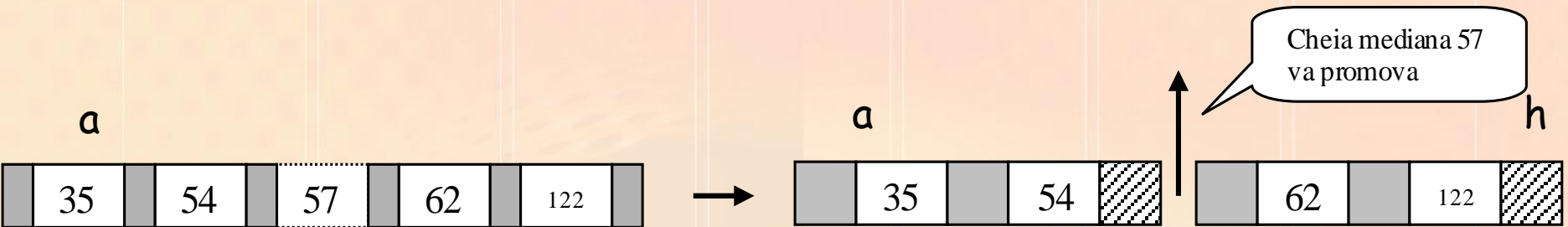
## Inserarea cheii 56: nodul c.



Fisionarea nodului c

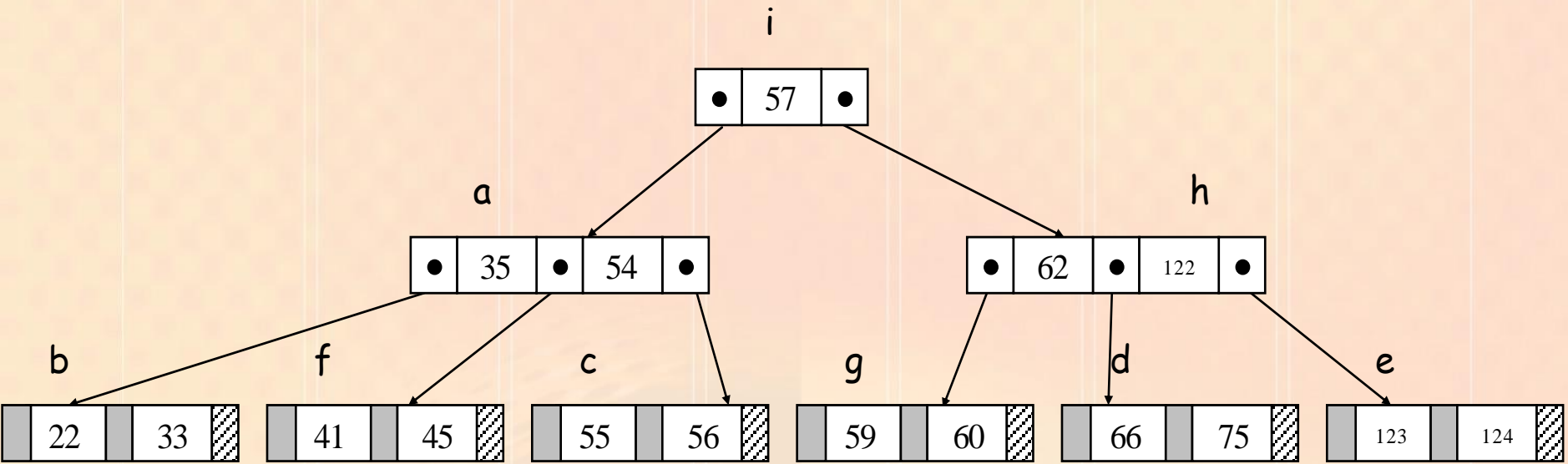
# ARBORI B

Nodul părinte **a** este plin și nu poate primi noua cheie 57. Algoritmul de fisurare este aplicat din nou, pentru nodul **a**.



Fisurarea nodului a

# ARBORI B



**Noua structura a arborelui B**

# ARBORI B

Algoritmul de inserare a unei chei într-un arbore B:

- inserează noua valoare de cheie în nodul frunză corespunzător;
- `nodul_curent = nodul_frunza`;
- `while(nodul_curent este OVERFLOW)`:
  - divide `nodul_curent` în două noduri aflate pe același nivel și promovează cheia mediană în nodul părinte pentru `nodul_curent`;
  - `nodul_curent = nodul_părinte` pentru `nodul_curent`.

# ARBORI B

**Cel mai rău caz: aplicarea algoritmului de fisionare pe întreaga înălțime a arborelui, fisionându-se  $h-1$  noduri ( $h$  este înălțimea arborelui înainte de inserare).**

# ARBORI B

## Operatii de baza (continuare):

### 3. Stergerea unei chei dintr-un arbore B:

- **Simpla** dacă valoarea de cheie ștersă se află într-un nod frunză;
- **Complexa** dacă valoarea de cheie ștersă nu se află într-un nod frunză: ștergere logica, fiind înlocuită cu o alta, vecină în inordine, care va fi ștersă fizic.



# ARBORI B

## Stergerea unei chei dintr-un arbore B (continuare):

- Situatii de stergere:
  - nodul conține mai mult de  $\lceil \frac{m}{2} \rceil$  chei: ștergerea nu ridică probleme;
  - nodul are numărul minim de chei  $\lceil \frac{m}{2} \rceil$ : după ștergere numărul de chei din nod va fi insuficient; se împrumută o cheie din nodul vecin cu cel puțin  $\lceil \frac{m}{2} \rceil$  chei (**partajare**)

# ARBORI B

(continuare)

**Dacă nu se poate face o partajare (nodurile vecine au numărul minim de chei): două noduri vecine vor fuziona, împrumutându-se o cheie și din nodul părinte.**

**Partajarea sau fuzionarea trebuie eventual repetate și pentru nivelurile superioare.**

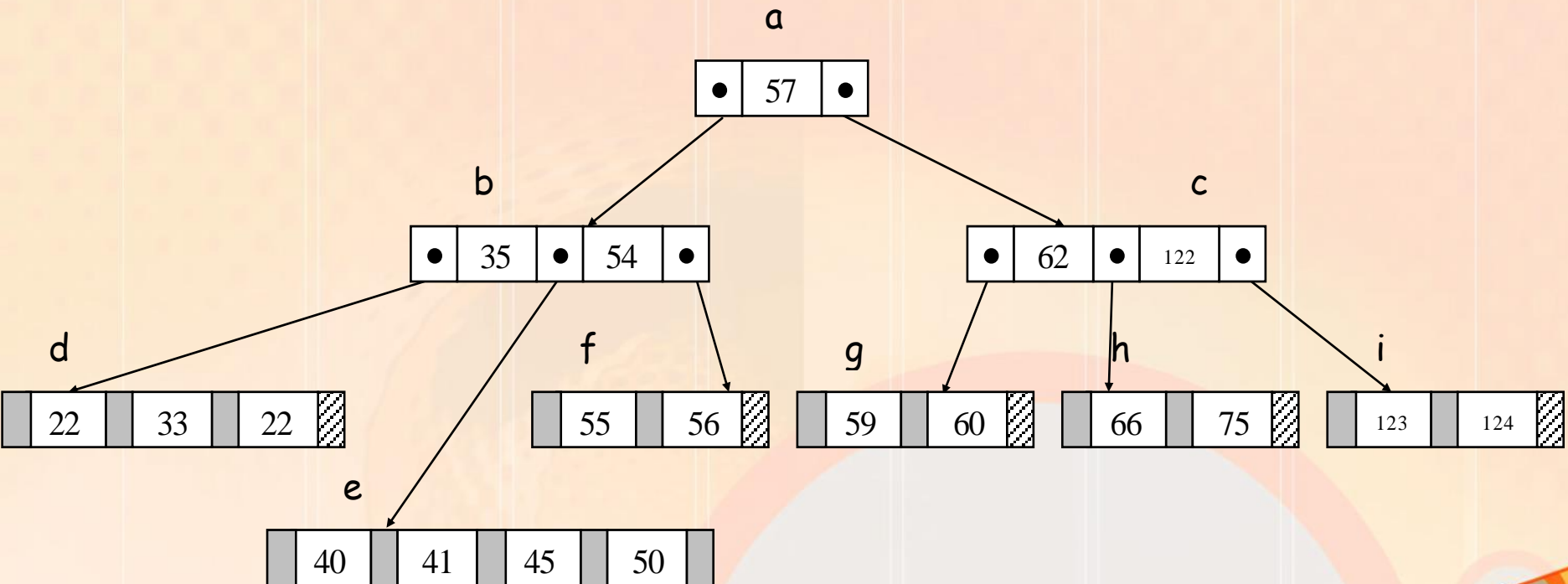
# ARBORI B

**Observatie:**

**Cazul cel mai nefavorabil: partajarea sau fuzionarea parcurg întreaga înălțime a arborelui, se va forma o nouă rădăcină, înălțimea arborelui scăzând cu un nivel.**

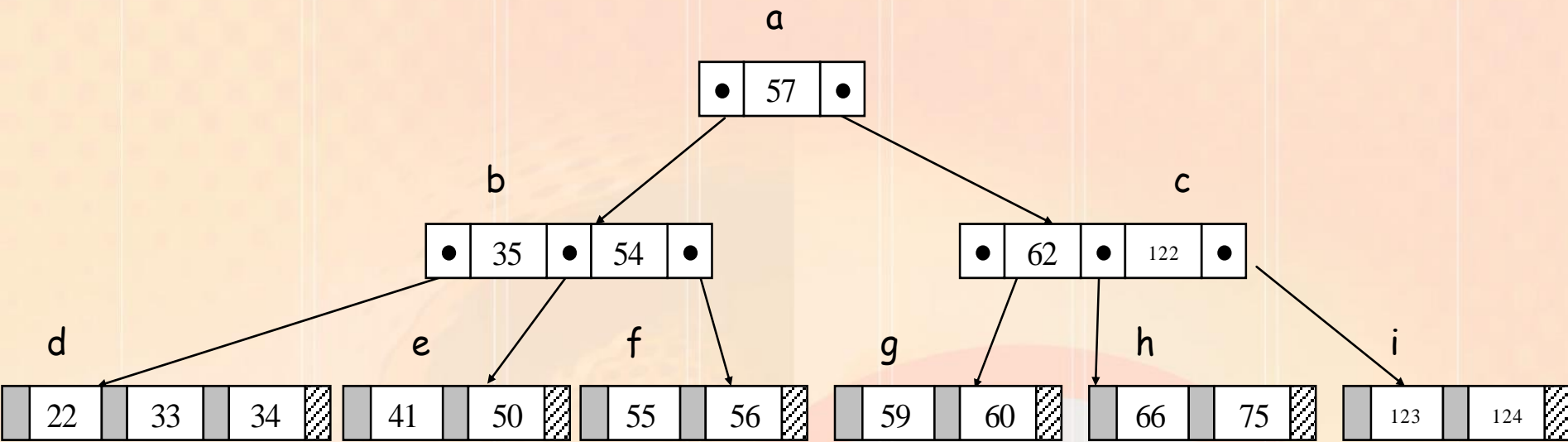
# ARBORI B

Se consideră următoarea configurație de arbore B de ordin 5:



# ARBORI B

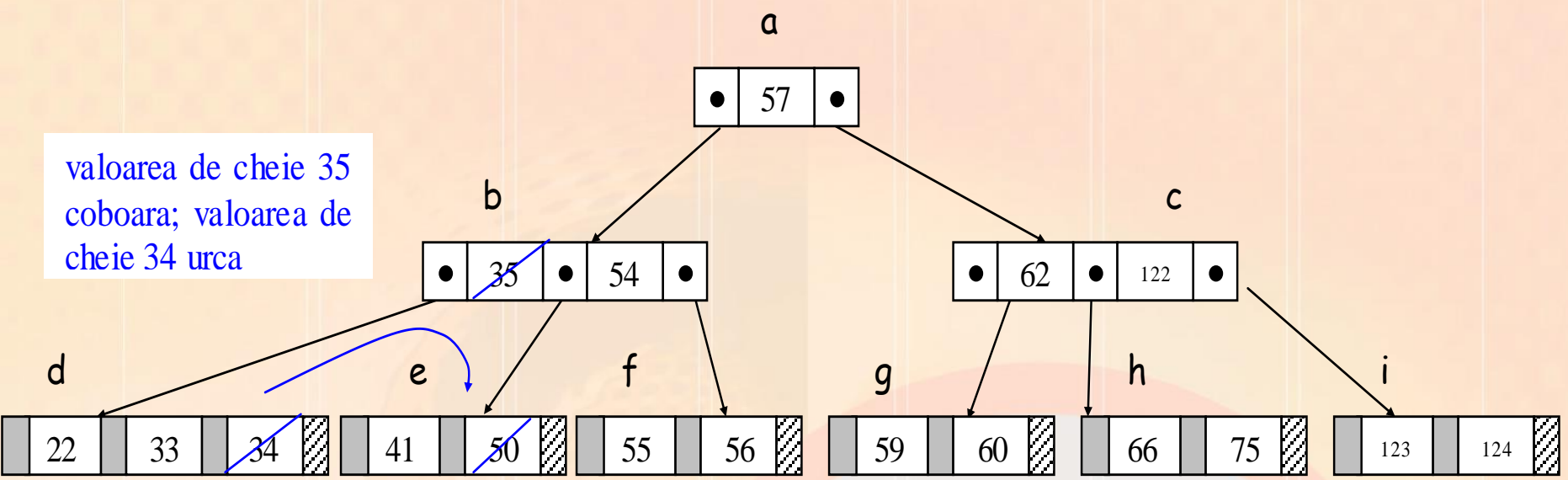
Ștergerea valorilor de cheie 40 și 45 din nodul e nu ridică probleme



# ARBORI B

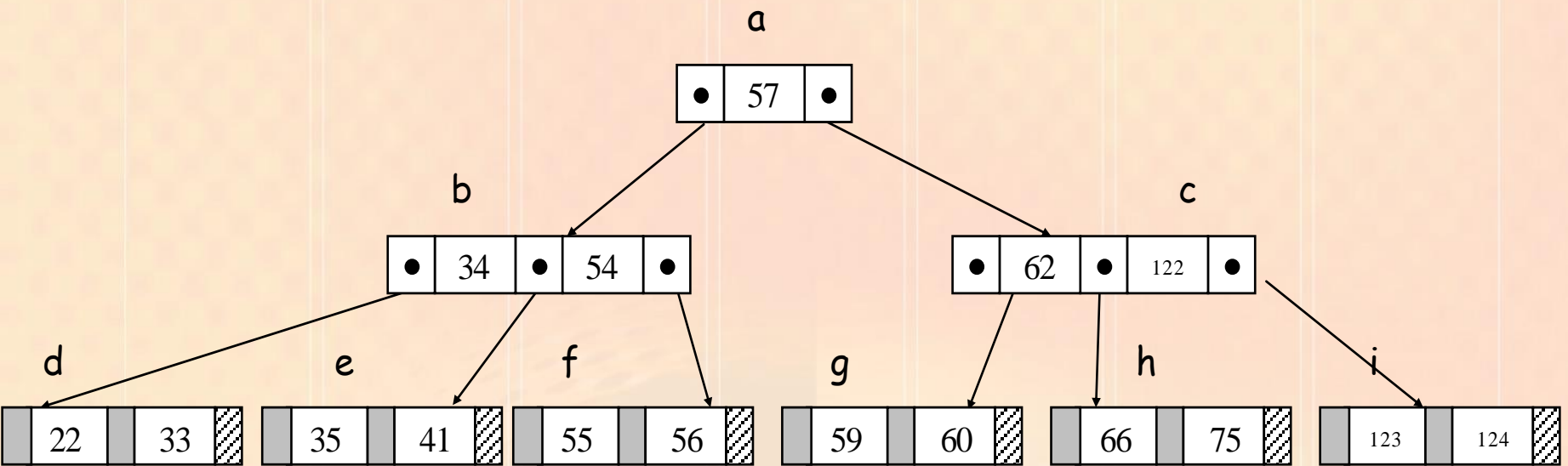
## Ștergerea valorii de cheie 50: partajare între nodurile d și e.

valoarea de cheie 35 coboara; valoarea de cheie 34 urca





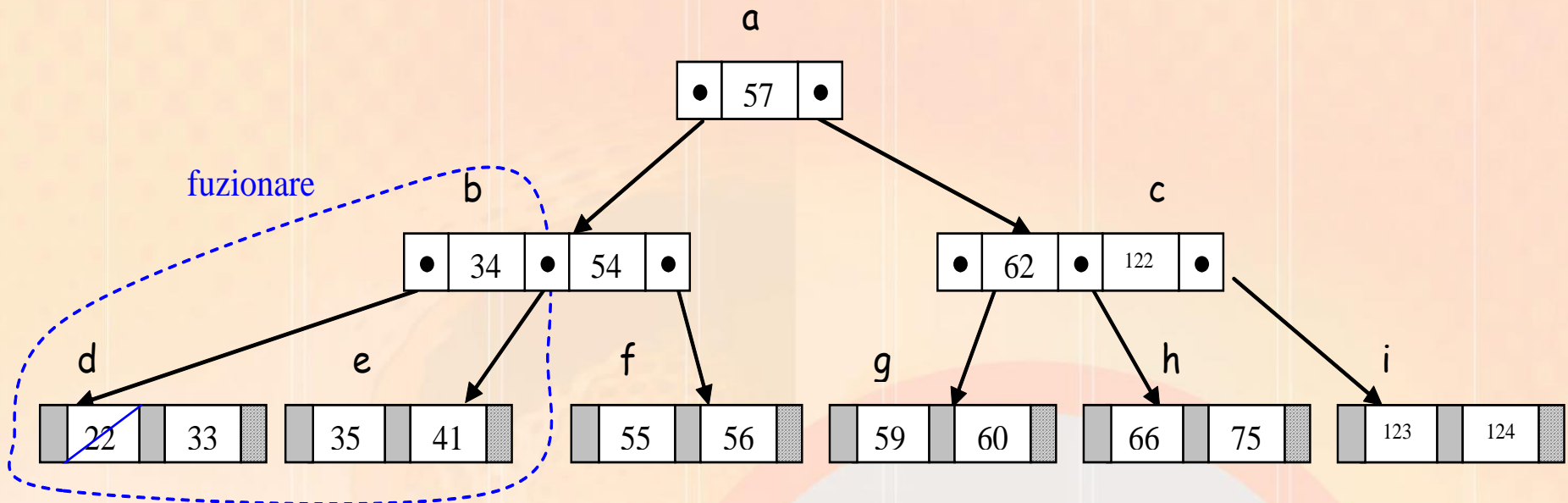
# ARBORI B



**Structura arborelui după partajare**

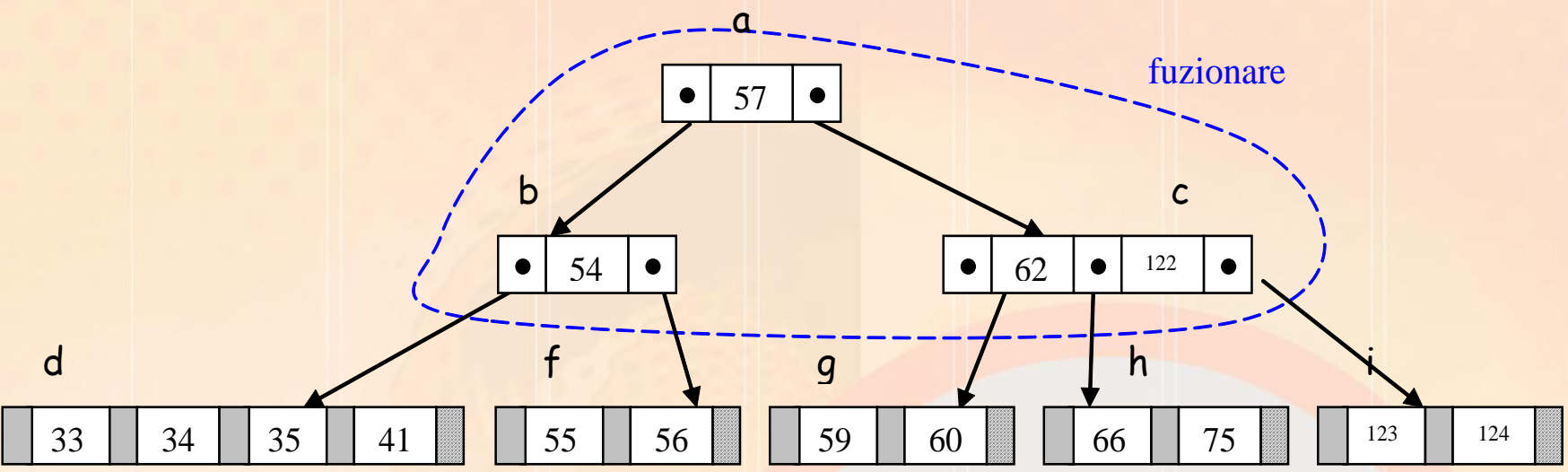
# ARBORI B

Ștergerea valorii de cheie 22: fuzionarea nodurilor d și e.



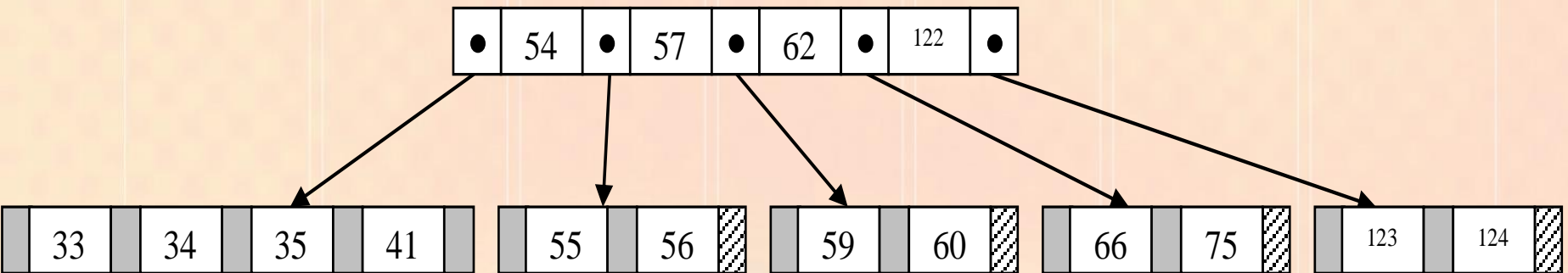
# ARBORI B

În urma fuzionării nodurilor d și e, nodul b va conține prea puține valori de cheie: fuzionare nodurile b și c



# ARBORI B

## Structura finală a arborelui B:



# ARBORI B

## Algoritmul de ștergere dintr-un arbore B:

- **daca(valoarea de cheie care se șterge nu este într-un nod frunză) atunci: înlocuiește valoarea de cheie cu succesori/predecesor;**
- **nodul\_curent = nodul\_frunza;**

# ARBORI B

- while (nodul\_curent este UNDERFLOW ):
  - încearcă partajarea cu unul din nodurile vecine aflate pe același nivel, via nodul părinte;
  - daca(nu este posibila partajarea) atunci:
    1. fuzionează nodul\_curent cu un nod vecin, folosind o valoare de cheie din nodul părinte;
    2. nodul\_curent = nod\_părinte pentru nodul\_curent.