



ARBORI BICOLORI

Șl. Dr. Ing. Șerban Radu

Departamentul de Calculatoare

Facultatea de Automatică și Calculatoare



Introducere

- Arborii binari de căutare obișnuiți prezintă un dezavantaj major
- Performanțele lor sunt bune dacă datele sunt inserate în ordine aleatoare
- Dacă însă datele sunt inserate în ordine crescătoare sau descrescătoare, aceste performanțe se degradează



Introducere

- Atunci când valorile inserate sunt deja ordonate (sau aproape ordonate), arborele rezultat este neechilibrat
- Arborii neechilibrați își pierd capacitatea de căutare (sau inserare sau ștergere) rapidă a unui element



Arbori bicolori

- Arborii bicolori sunt arbori binari de căutare cu anumite proprietăți în plus
- Utilizarea arborilor bicolori reprezintă, în majoritatea cazurilor, cea mai eficientă soluție de echilibrare, cel puțin în cazul în care datele sunt păstrate în memorie



Operații cu arbori bicolori

- Căutarea într-un arbore bicolor funcționează la fel ca într-un arbore binar
- Inserarea și ștergerea se modifică semnificativ

Inserarea de sus în jos

- Inserarea în arborii bicolori se numește inserare de sus în jos (top-down)
- Arborele va suferi modificări structurale atunci când algoritmul îl parcurge de sus în jos, pentru a găsi locul în care va insera un nod



Inserarea de jos în sus

- O altă posibilitate este inserarea de jos în sus (bottom-up)
- Aceasta presupune găsirea locului în care se va insera nodul și efectuarea ulterioară a unor modificări structurale în arbore
- Inserarea de jos în sus este mai puțin eficientă, deoarece presupune două parcurgeri ale arborelui



Observații

- Un arbore fără ramuri degenerază într-o listă înlanțuită
- Ca și în cazul unei liste înlanțuite, trebuie să parcurgem, în medie, jumătate din numărul total de elemente, pentru a găsi un anumit element

Observații

- Căutarea printr-un astfel de arbore, cu 10.000 de elemente, va necesita 5.000 de comparații, în timp ce, pentru un arbore echilibrat, numărul de comparații este cel mult 14
- Pentru date deja sortate, este indiferent dacă folosim un arbore binar de căutare sau o listă înlănțuită



Observații

- Datele parțial sortate vor genera arbori parțial dezechilibrați
- Deși nu au performanțe atât de scăzute ca arborii cu grad maxim de dezechilibru, arborii parțial dezechilibrați nu reprezintă o soluție optimă în ceea ce privește căutarea



Echilibrarea arborilor

- Fiecare nod din arbore trebuie să aibă un număr aproximativ egal de descendenți, atât în partea stângă, cât și în partea dreaptă
- Într-un arbore bicolor, echilibrarea este asigurată prin implementarea operației de inserare



Operația de inserare

- La inserarea unui element, funcția care efectuează operația verifică dacă se mențin anumite caracteristici ale arborelui
- În caz contrar, funcția modifică structura arborelui
- Prin menținerea acestor caracteristici, se păstrează starea de echilibrare a arborelui



Caracteristicile arborilor bicolori

- 1. Nodurile sunt colorate, fiecare nod este fie negru, fie roșu
- 2. Pe parcursul inserării și ștergerii, se asigură păstrarea unor aranjamente prestabilite ale acestor culori



Reguli de colorare

- La inserarea (sau la ștergerea) unui nod, trebuie respectate anumite reguli, numite **reguli de colorare**
- Respectarea acestor reguli asigură echilibrarea arborelui



Reguli de colorare

- 1. Fiecare nod este roșu sau negru
- 2. Rădăcina are întotdeauna culoarea neagră
- 3. Dacă un nod este roșu, fiii săi trebuie să fie negri (reciproca nu este neapărat adevărată)
- 4. Toate căile de la rădăcină spre frunze, sau spre fii inexistenți, trebuie să conțină un număr egal de noduri negre

Observații

- Fiul inexistent din ultima regulă este de fapt un loc în care se poate atașa un fiu unui nod care nu este frunză
- Acesta este un potențial fiu stâng al unui nod cu un fiu drept, sau invers, un potențial fiu drept al unui nod cu un fiu stâng

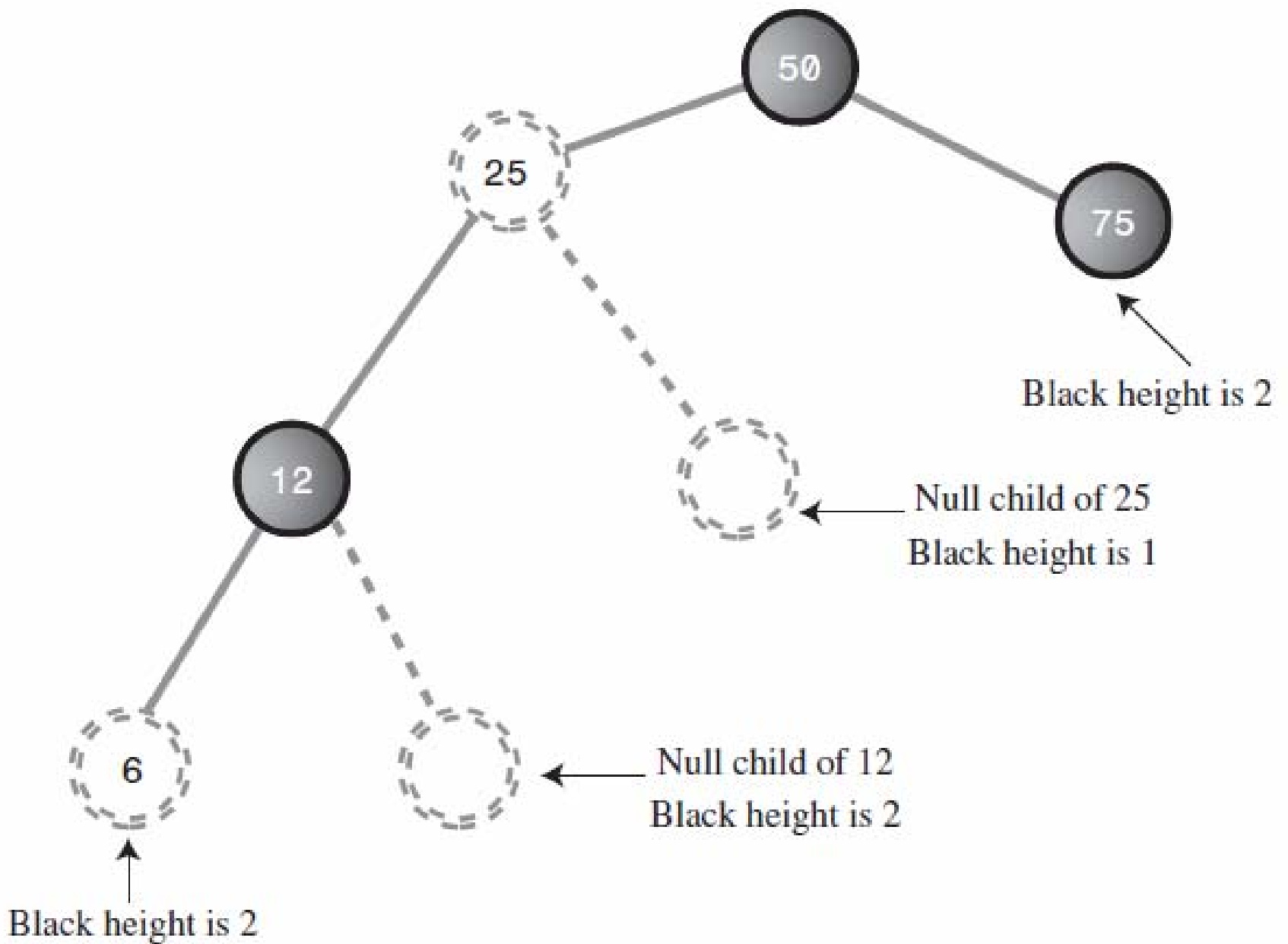
Observații

- Numărul nodurilor negre de pe calea dintre rădăcină și o anumită frunză se numește **înălțime neagră**
- Ultima regulă mai poate fi enunțată astfel:
- 4. Înălțimea neagră trebuie să fie constantă pentru toate căile de la rădăcină spre frunze



Fii inexistenți

- Un fiu inexistent este un posibil fiu al unui nod care nu este frunză
- Acest fiu nu există însă în arbore



Observații

- Calea de la rădăcină până la fiul drept al lui 25 (inexistent) are numai un singur nod negru, spre deosebire de căile spre nodurile 6 sau 75, care au două astfel de noduri
- Acest arbore nu respectă regula 4, deși ambele căi spre nodurile frunză au același număr de noduri negre



Rotații

- Pentru a echilibra un arbore, este necesar să efectuăm o rearanjare fizică a nodurilor
- Dacă toate nodurile sunt la stânga rădăcinii, trebuie să deplasăm unele dintre ele în partea dreaptă
- Această deplasare se efectuează utilizând rotații



Rotații

- Rotațiile reprezintă modalități de rearanjare a nodurilor
- Acestea se folosesc pentru a rezolva două probleme:
 - 1. Ridicarea unora dintre noduri și coborârea altora, pentru a echilibra arborele
 - 2. Asigurarea respectării ordinii caracteristice arborilor binari de căutare



Observații

- Regulile de colorare și modificările culorilor se utilizează pentru a se putea decide când se execută o rotație
- Într-o rotație, nodurile nu efectuează rotații propriu-zise, ci doar relațiile dintre ele se modifică



Observații

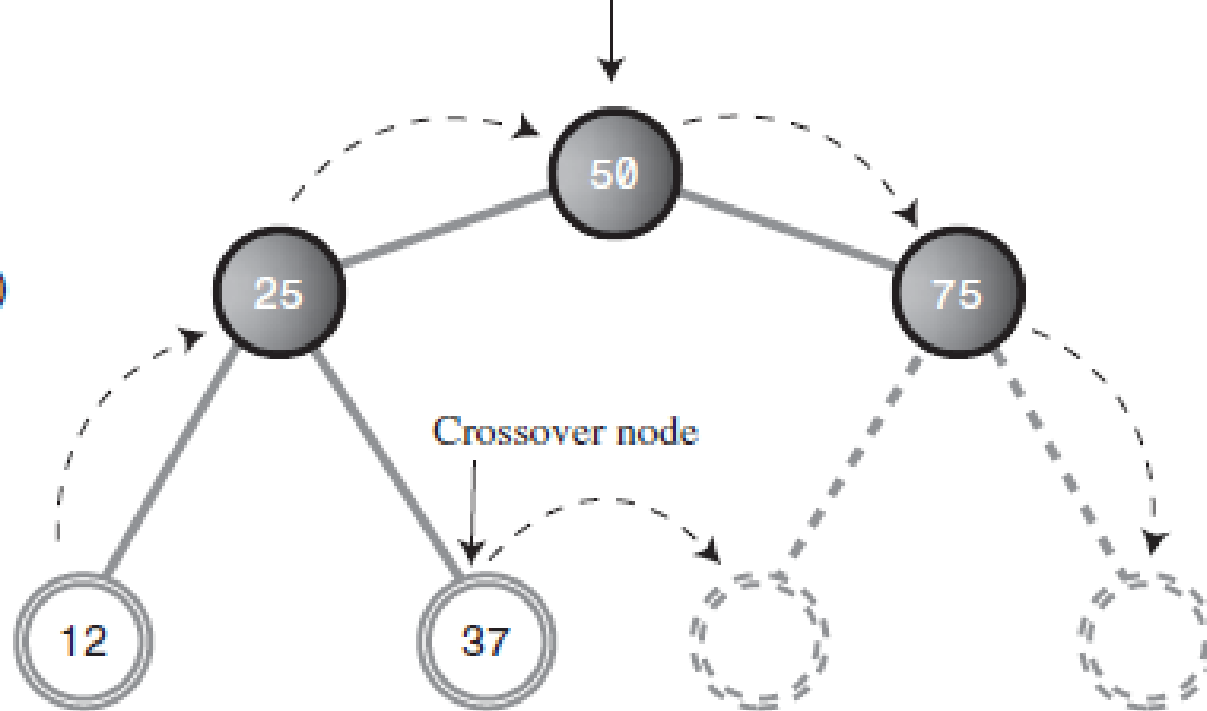
- Unul dintre noduri este ales ca “vârf” al rotației
- Dacă efectuăm o rotație spre dreapta, acest “vârf” se va deplasa în jos și spre dreapta, în poziția fiului său drept
- Fiul stâng al nodului din vârf se va deplasa în sus, luând locul părintelui său



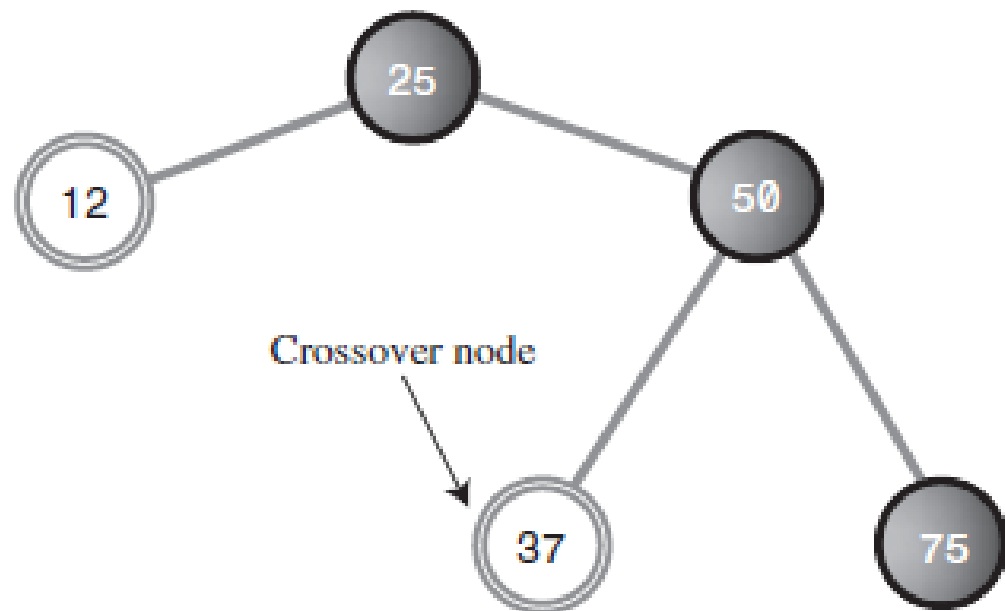
Observații

- Nodul din vârf nu reprezintă “centrul” rotației
- Orice nod poate fi vârful unei rotații, dacă dispune de fiul necesar
- Dacă se efectuează o rotație spre dreapta, nodul din vârf trebuie să aibă un fiu stâng
- Dacă se efectuează o rotație spre stânga, nodul din vârf trebuie să aibă un fiu drept

a)



b)



Un nod care traversează

- În urma rotației la dreapta, toate nodurile se vor deplasa
- Nodul 12 urmează deplasarea lui 25 în sus, iar fosta rădăcină 50 urmează deplasarea lui 75 în jos
- Nodul 37 s-a desprins de 25, al cărui fiu drept era, devenind fiul stâng al lui 50

Observații

- Unele dintre noduri se deplasează în sus, altele în jos, dar 37 a traversat arborele
- Rotația a produs o încălcare a regulii 4, care va fi rezolvată mai târziu
- În poziția inițială, 37 se numește un **nepot interior** al nodului din vârf, 50, iar 12 este un **nepot exterior**



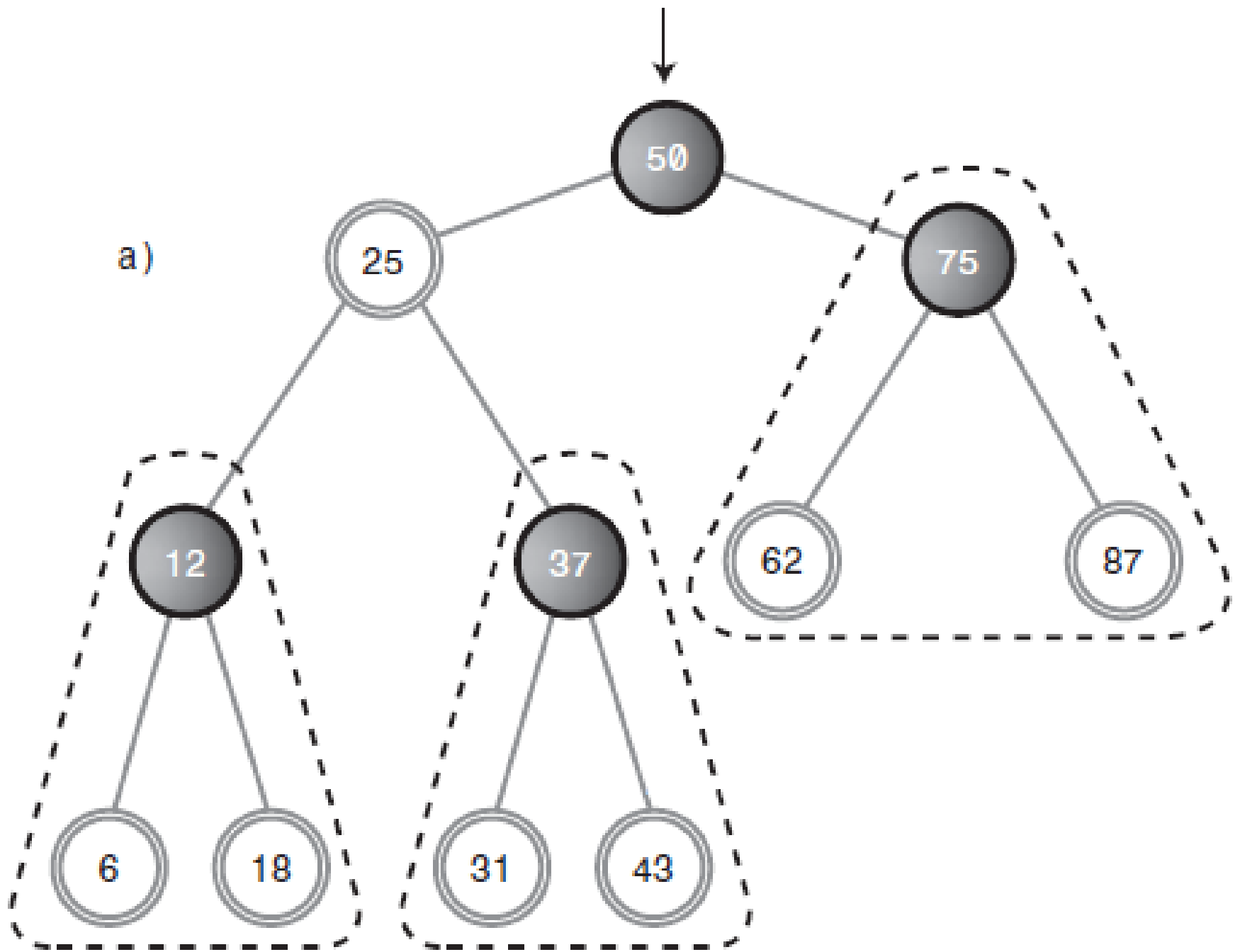
Observații

- Nepotul interior, dacă este fiul nodului care s-a deplasat în sus (fiul stâng al nodului din vârf, în cazul unui rotații la dreapta), va fi întotdeauna deconectat de la părintele său și reconectat cu fostul său bunic

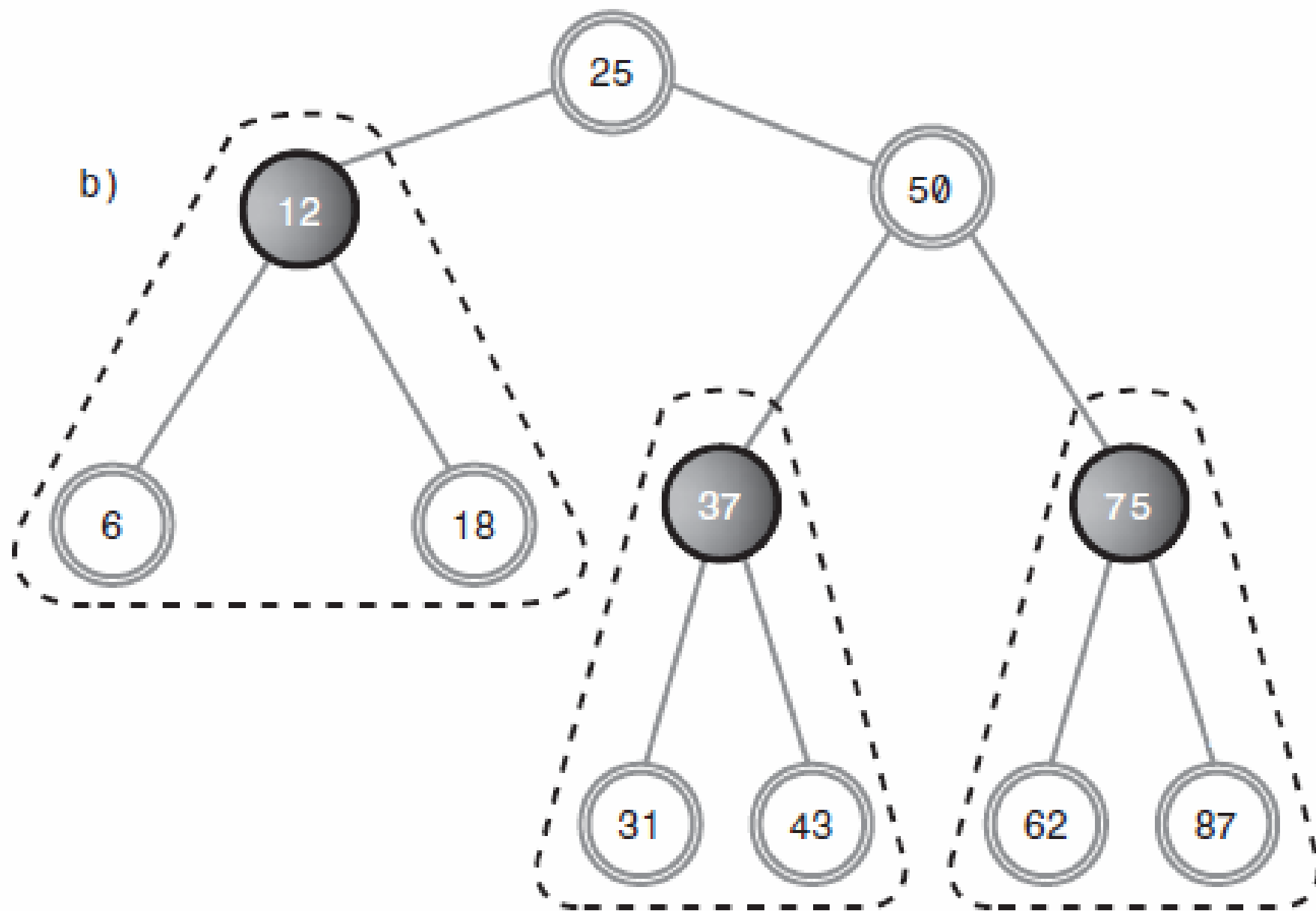


Subarbori în mișcare

- În urma unei rotații, anumite noduri își modifică poziția
- Se pot deplasa însă sub arbori întregi



b)



Exemplu

- Într-o rotație la dreapta, având ca vârf rădăcina 50, se observă că mai multe noduri s-au deplasat simultan
- Nodul din vârf (50) își va înlocui fiul drept
- Fiul stâng (25) al nodului din vârf își înlocuiește fostul părinte
- Întregul subarbore cu rădăcina 12 se deplasează în sus

Exemplu

- Întregul subarbore cu rădăcina 37 traversează arborele, devenind fiul stâng al lui 50
- Subarboarele cu rădăcina 75 se deplasează în jos
- Pozițiile relative ale nodurilor din același subarbore nu sunt afectate de rotație
- Întregul subarbore se deplasează solidar

Notății

- Utilizăm literele X, P și G pentru a nota o secvență de noduri înrudite
- X reprezintă un nod care a produs o încălcare a regulilor de colorare
- Uneori, X se referă la un nod nou inserat, iar alteori la un fiu, în cazul în care atât părintele, cât și fiul sunt de culoare roșie

Notatii

- X este un anumit nod
- P este părintele lui X
- G este bunicul lui X (adică părintele lui P)
- Când se parcurge arborele în jos pentru a găsi locul de inserare, se efectuează o inversare a culorilor, oriunde se întâlnește un părinte negru cu doi fii roșii (ceea ce încalcă regula 2)

Observații

- Această inversare provoacă uneori un conflict roșu-roșu (se încalcă regula 2)
- Dacă X este fiul roșu și P părintele roșu, conflictul se poate rezolva apelând la o rotație simplă sau dublă, după cum X este nepotul exterior sau interior al lui G
- Efectuând inversări de culoare și rotații, se ajunge la punctul de inserare și se inserează nodul

Observații

- După inserarea noului nod X , dacă P este negru, pur și simplu îi atașăm un fiu roșu
- Dacă P este roșu, avem două posibilități, după cum X este nepotul exterior sau interior al lui G
- Se efectuează două schimbări de culoare



Observații

- Dacă X este nepot exterior, se efectuează o singură rotație
- Dacă X este nepot interior, se efectuează două rotații
- În urma acestor operații, arborele se va reechilibra



Inversări de culoare la parcurgerea descendentă a arborelui

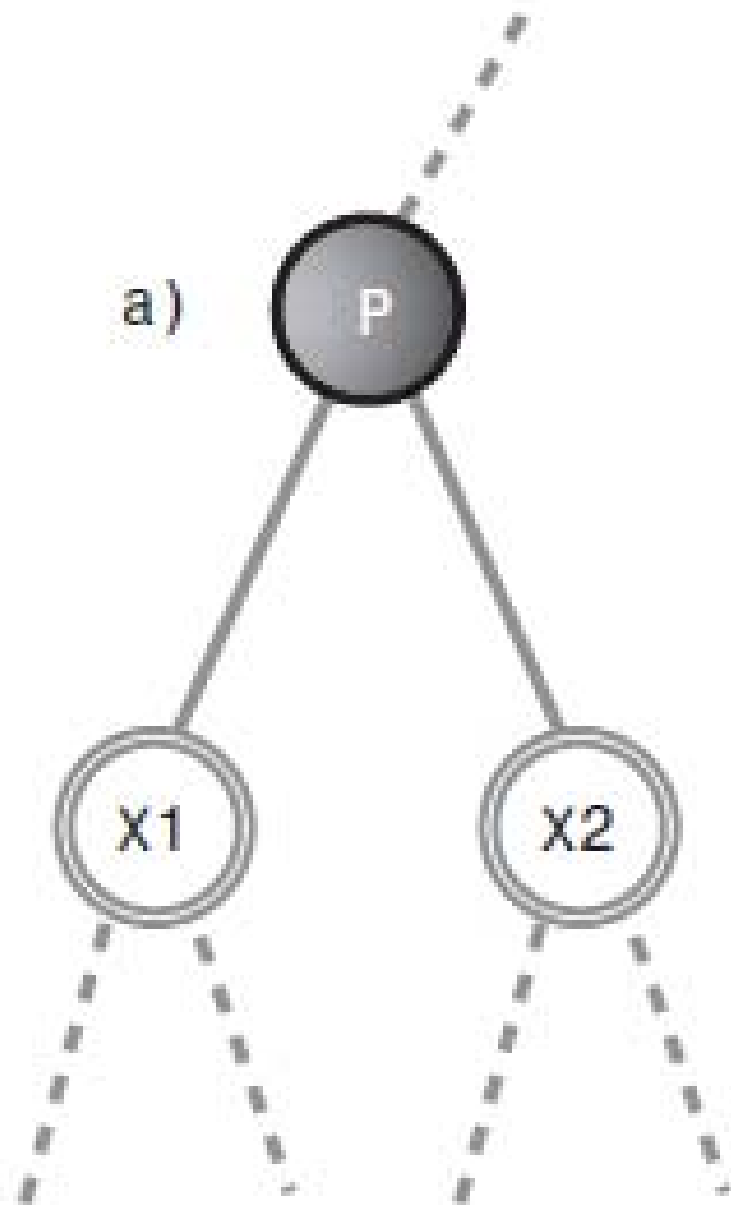
- Metoda de inserare într-un arbore bicolor începe prin a efectua ceea ce se face și în cazul unui arbore de căutare – parcurgerea drumului de la rădăcină până la locul în care trebuie inserat nodul, deplasându-se la stânga sau la dreapta, în funcție de comparația dintre valoarea inserată și cheia nodului curent



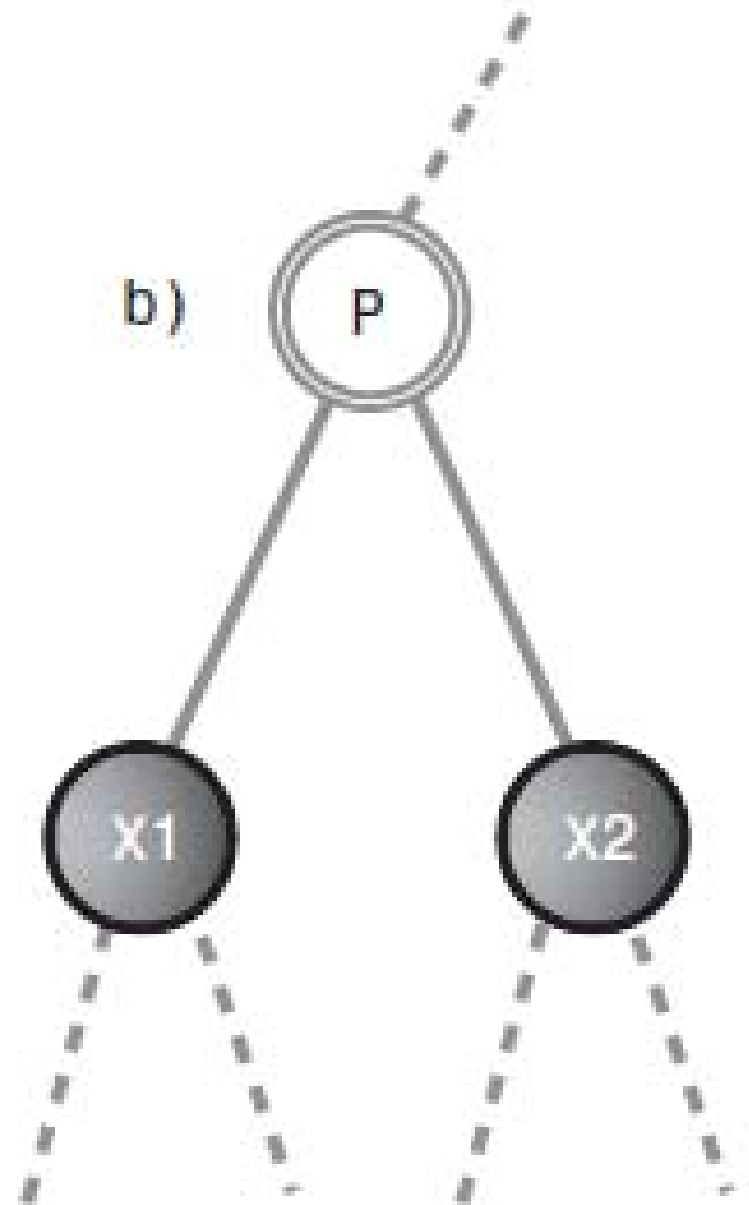
Inversări de culoare

- Într-un arbore bicolor, găsirea locului de inserare se complică cu efectuarea unor inversiuni de culoare și a unor rotații
- La fiecare întânire a unui nod negru cu doi fii roșii, culoarea fiilor trebuie să devină neagră, iar a părintelui roșie (exceptând cazul în care părintele este nodul rădăcină, care rămâne întotdeauna negru)

a)



b)



Notății

- Se notează nodul din vârful triunghiului, care este negru, înainte de inversiune cu P
- Fie $X1$ și $X2$ fiul stâng, respectiv fiul drept al lui P
- Inversiunea nu modifică numărul de noduri negre de pe căile de la rădăcină spre frunze (sau fii inexistenți) și care trec prin P

Observații

- Toate aceste căi trec prin P și apoi prin $X1$ și $X2$
- Înainte de inversarea culorilor, doar P este negru, deci triunghiul (constând din nodurile P , $X1$ și $X2$) adaugă un singur nod negru acestor căi

Observații

- După inversarea culorilor, P nu mai este negru, dar ambii săi fii sunt, deci triunghiul contribuie tot cu un singur nod negru la fiecare din căile care îl parcurg
- Prin urmare, inversarea culorilor nu conduce la încălcarea regulii 4



Observații

- Inversarea culorilor este utilă, deoarece poate transforma frunzele roșii în frunze negre
- Astfel, va fi mai ușor să atașăm noi noduri roșii, fără a încălca regula 3

Observații

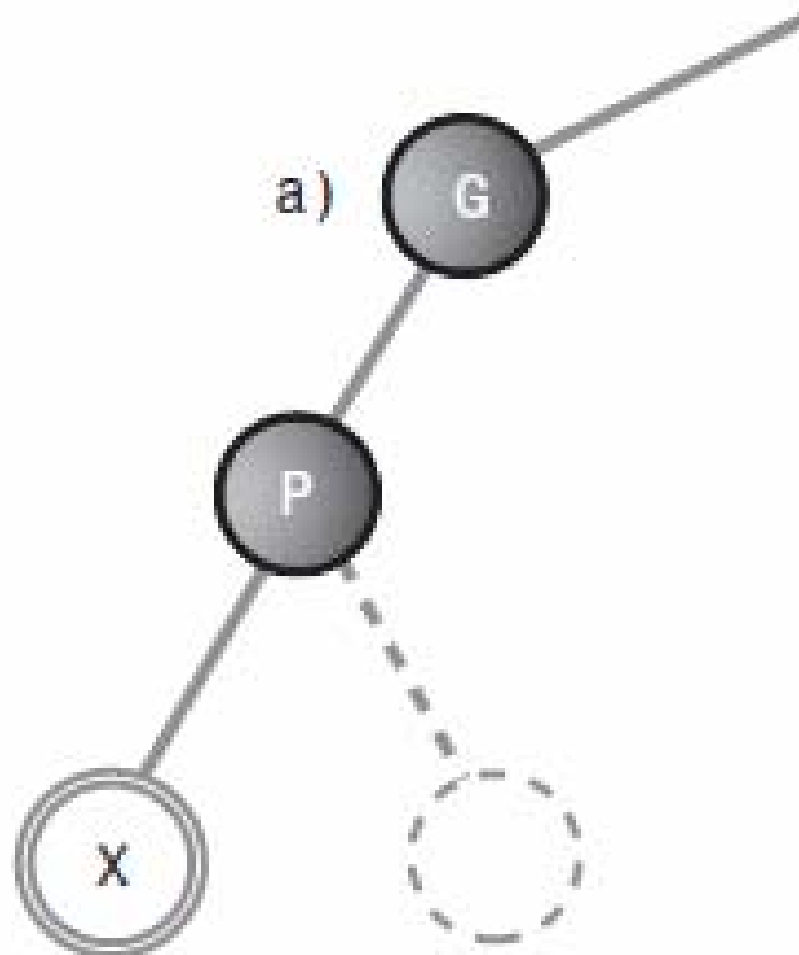
- Deși nu încalcă regula 4, inversarea culorilor poate duce la eludarea regulii 3
- Dacă părintele lui P este negru, nu se întâmplă nimic atunci când P va deveni roșu
- Dacă însă părintele lui P este roșu, după inversarea culorilor vor apărea două noduri roșii adiacente

Noduri roșii adiacente

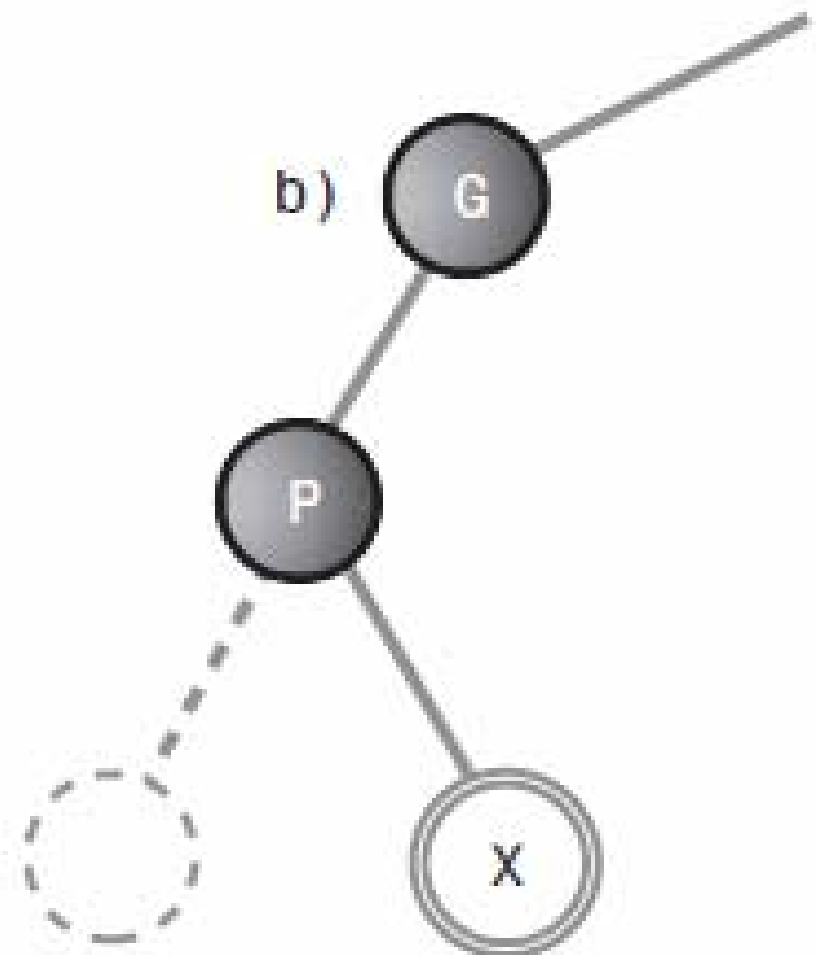
- Situația trebuie rezolvată înainte de a continua parcurgerea arborelui, pentru a determina locul de inserare
- Soluția este efectuarea unei rotații
- După ce s-a găsit locul potrivit din arbore, efectuând (dacă este cazul) inversări de culori și rotații pe parcurs, se poate insera noul nod la fel ca într-un arbore de căutare

Rotații după inserarea nodului

- Inserarea unui nod poate conduce la încălcarea regulilor de colorare
- După inserare, trebuie să verificăm apariția unor abateri de la reguli, pe care (dacă există) să le corectăm
- Noul nod inserat, X , este întotdeauna roșu
- X poate fi poziționat în mai multe moduri, în raport cu P și G

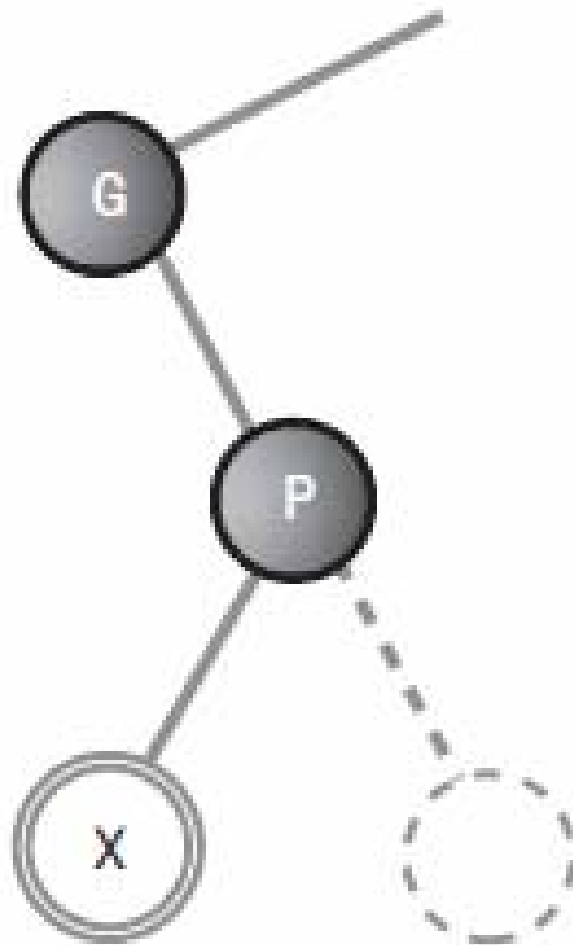


Outside grandchild
(Left child)



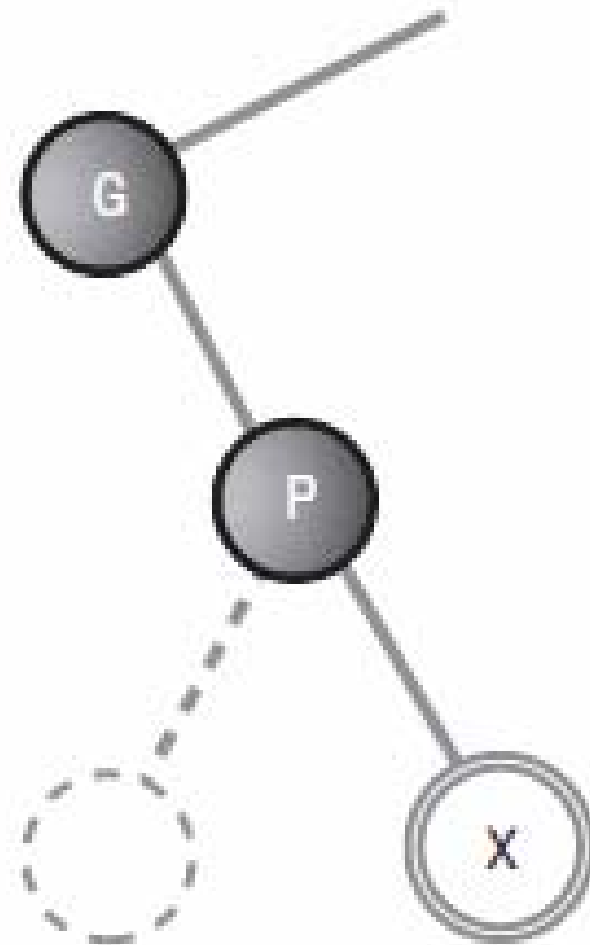
Inside grandchild
(Right child)

c)



Inside grandchild
(Left child)

d)



Outside grandchild
(Right child)

Observații

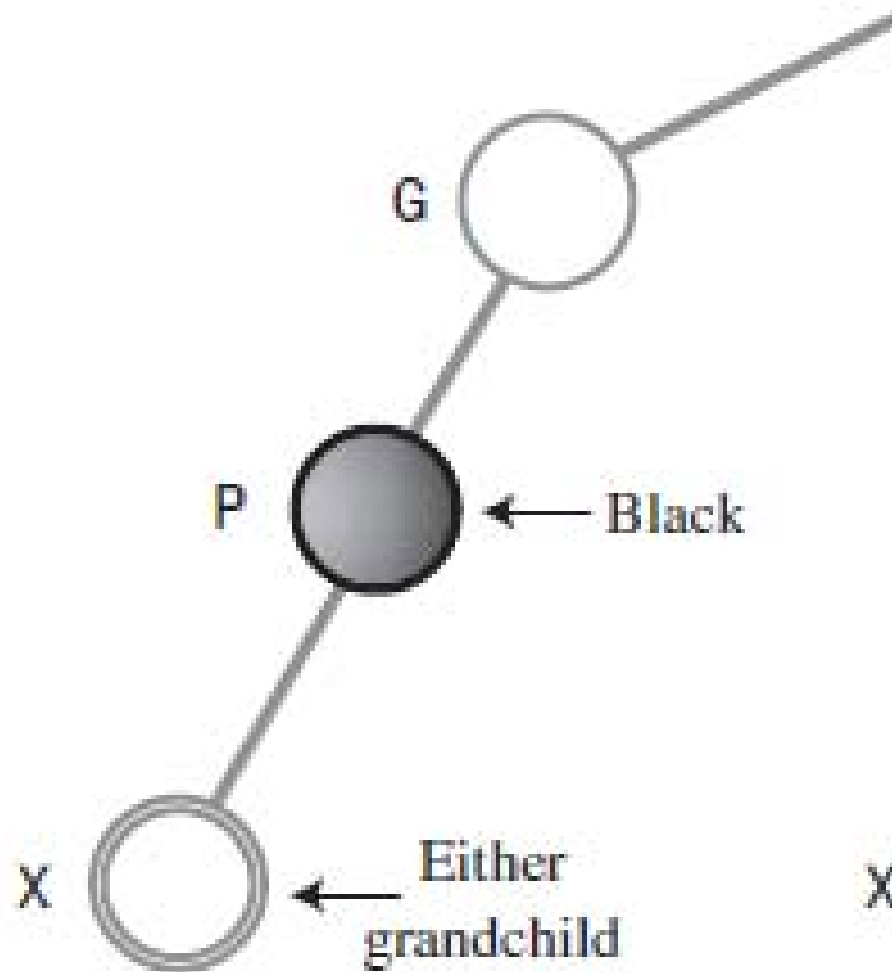
- X este un nepot exterior, dacă este situat față de părintele său P în aceeași parte în care acesta este situat în raport cu părintele său G
- X este nepot exterior pentru G, dacă:
 - X este fiu stâng al lui P și P este fiu stâng al lui G
 - X este fiu drept al lui P și P este fiu drept al lui G
- X este nepot interior al lui G, dacă este situat de partea opusă a lui P, față de cum este P situat în raport cu părintele său G

Observații

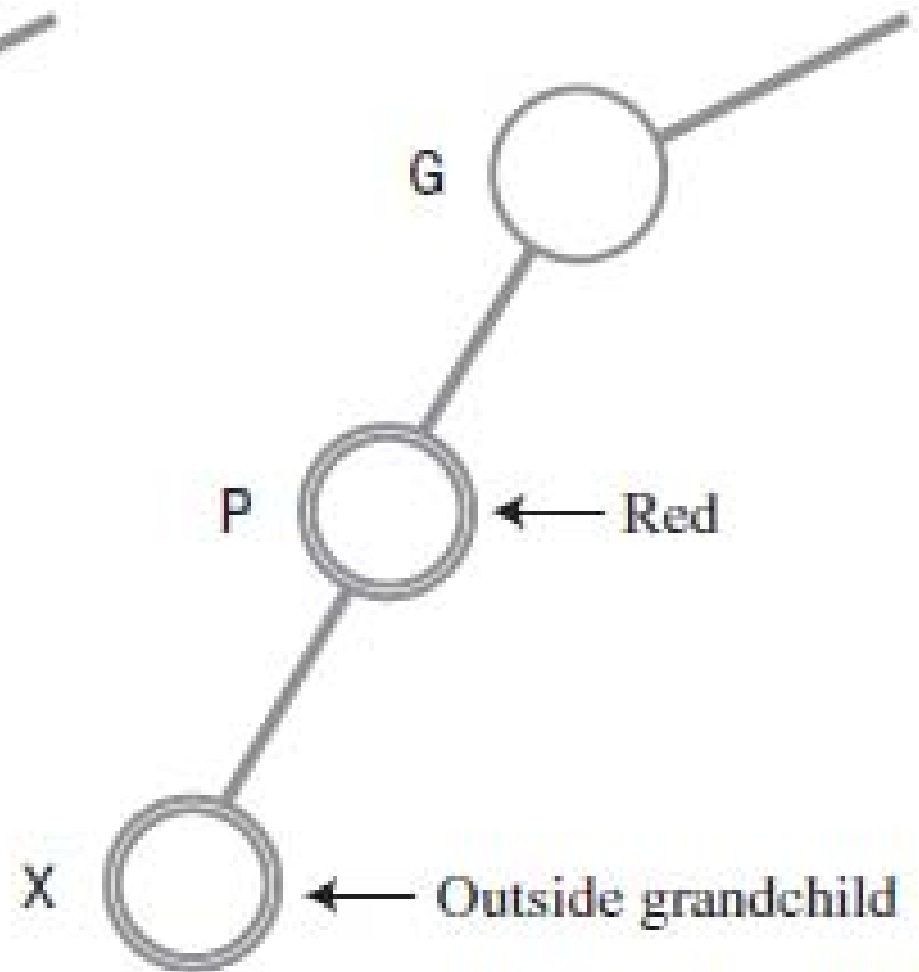
- Dacă X este un nepot exterior, poate fi fiul stâng sau drept al lui P , după cum P este la rândul său fiu stâng sau drept al lui G
- Există două situații similare în care X este nepot interior al lui G

3 moduri de dispunere a nodurilor

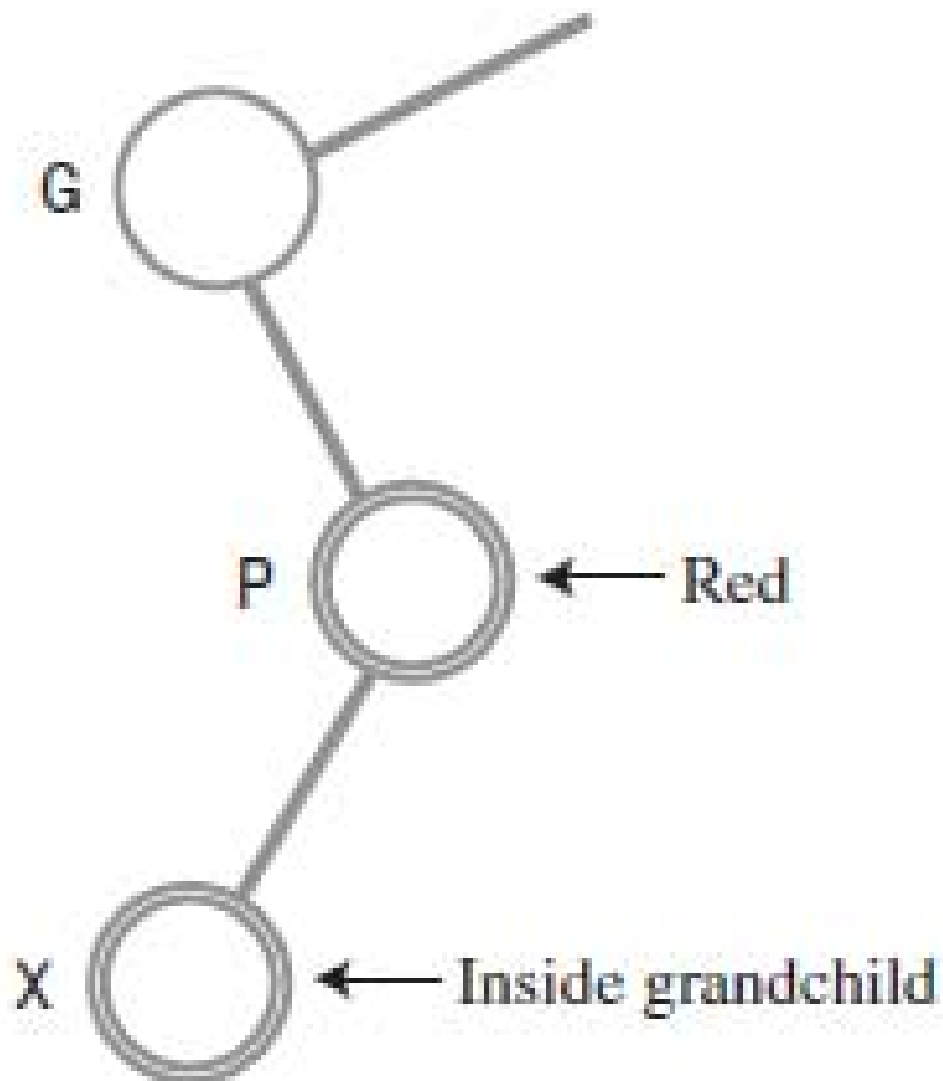
- Situațiile posibile după inserare:
- 1. P este negru
- 2. P este roșu și X este un nepot exterior al lui G
- 3. P este roșu și X este un nepot interior al lui G



a) Possibility 1: P is black



b) Possibility 2: P is red, and X is outside



c) Possibility 3: P is red,
and X is inside

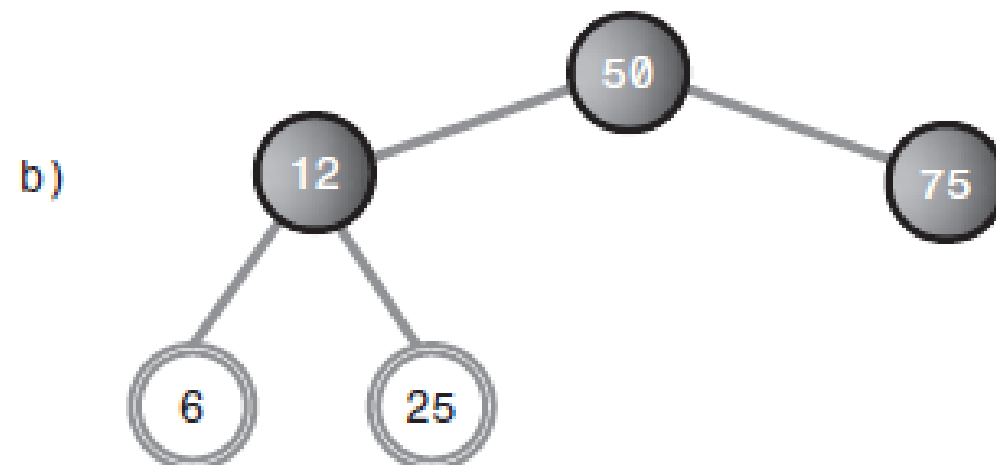
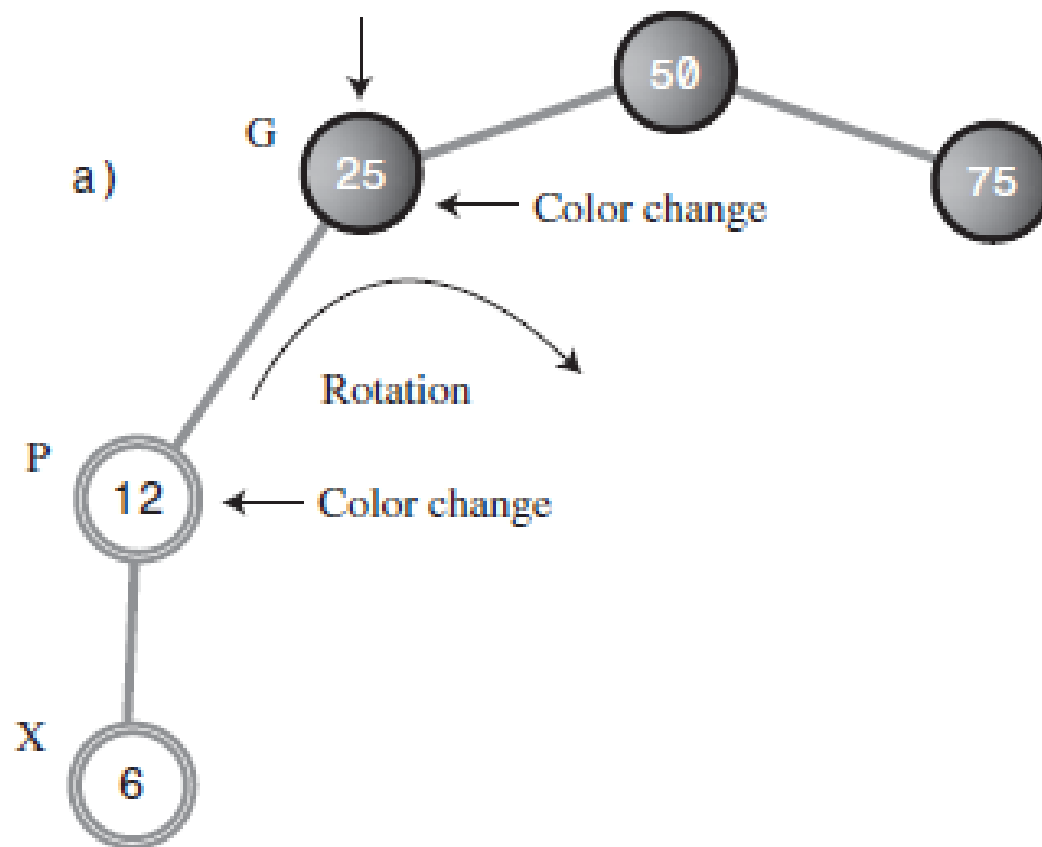
Cazul 1 – P este negru

- Nodul nou inserat este întotdeauna roșu
- Dacă părintele este negru, nu apare un conflict de culoare (regula 3) și nicio creștere unilaterală a numărului de noduri negre (regula 4)
- Regulile de colorare sunt respectate
- Inserarea este efectuată cu succes



Cazul 2 – P este roșu și X este exterior

- Sunt suficiente o rotație și câteva modificări ale culorilor
- Se poate reface corectitudinea de colorare (echilibrul arborelui) în 3 pași

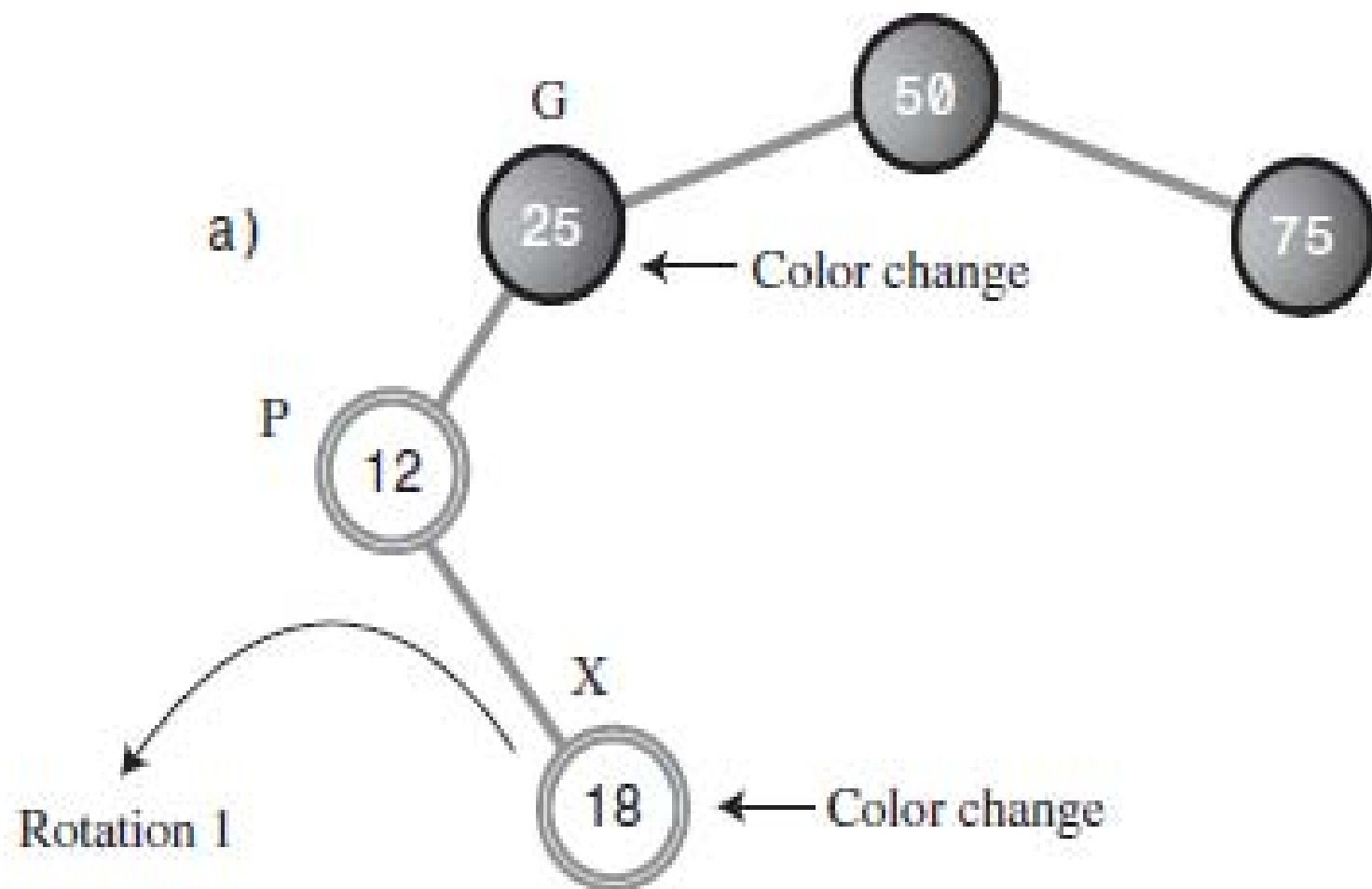


3 pași

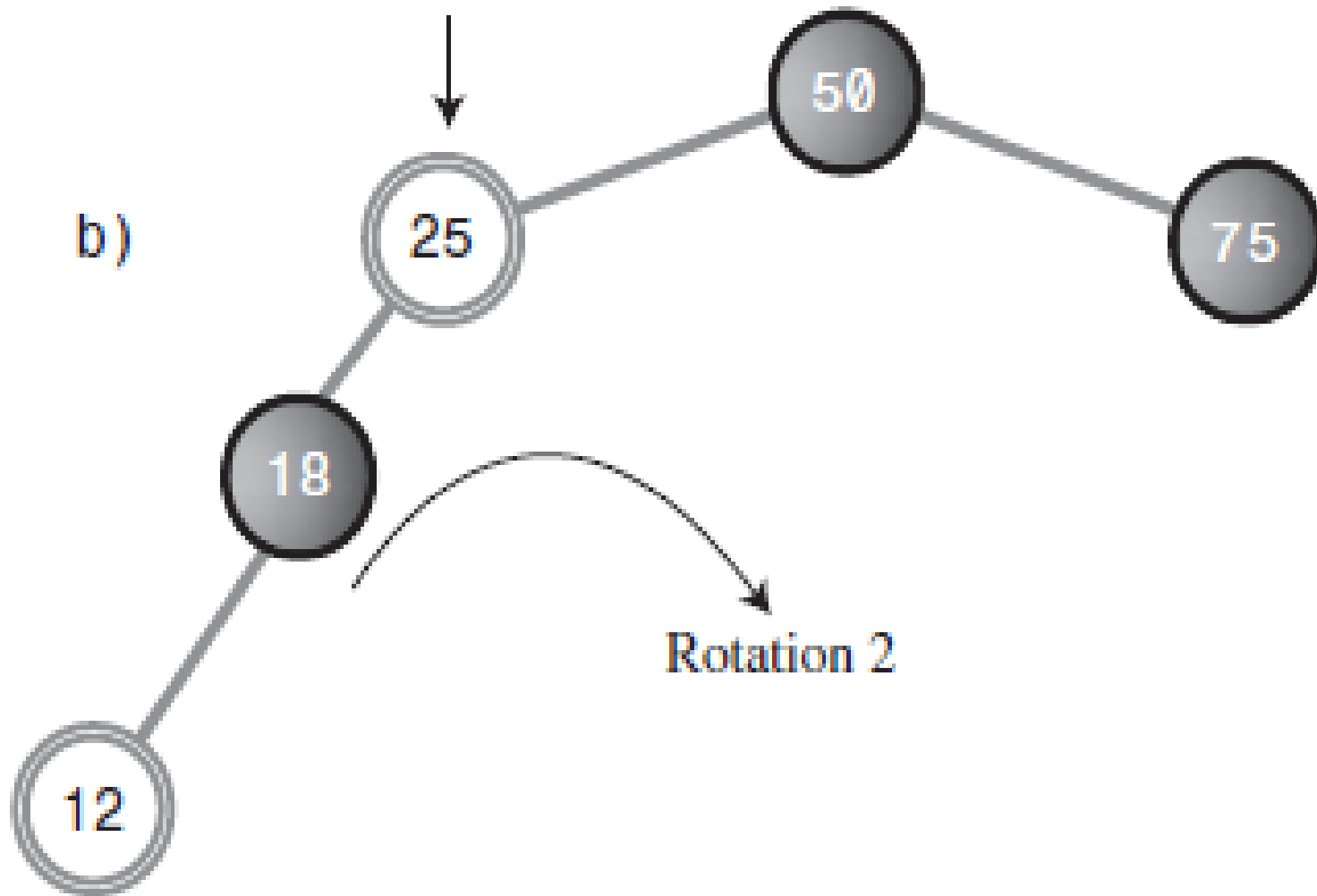
- 1. Se modifică culoarea bunicului G (25) al lui X (6)
- 2. Se modifică culoarea părintelui P (12) al lui X (6)
- 3. Se efectuează o rotație cu nodul G (25) în vârf, în direcția care asigură ridicarea lui X (6) în arbore
- În exemplu, rotația se va efectua spre dreapta

Cazul 3 – P este roșu și X este interior

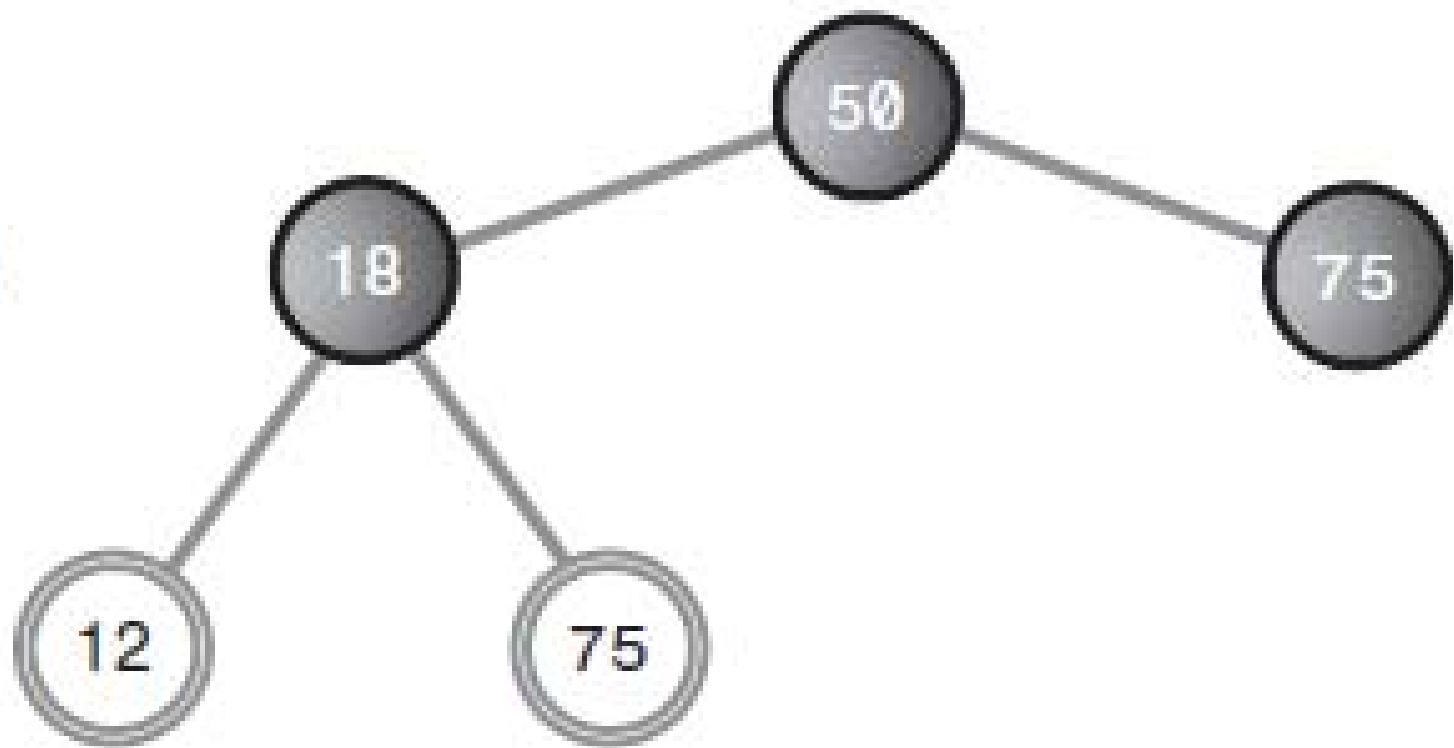
- Avem nevoie de două rotații și o modificare a culorii unui nod



b)



c)



Pași pentru echilibrarea arborelui

- 1. Se efectuează o rotație cu părintele P (12) situat în vârf, în direcția care asigură ridicarea lui X (spre stânga, în acest caz)
- 2. Se efectuează încă o rotație, cu bunicul G (25) situat în vârf, în direcția care asigură ridicarea lui X (18) (spre dreapta)
- 3. Se modifică culoarea lui P (12) – care este roșu și devine negru

Observații

- Secvența de operații aduce arborele într-o configurație în care respectă regulile de colorare și îl reechilibrează
- Există un caz simetric în care P este fiul drept al lui G



Observații

- Utilizarea inversării culorilor, la parcurgerea descendentă, elimină situațiile în care o rotație poate propaga încălcări ale regulilor de colorare mai sus în arbore
- Operația asigură că una sau două rotații sunt suficiente pentru a restabili corectitudinea întregului arbore



Observații

- Datorită inversării culorilor, inserările în arborii bicolori sunt mult mai eficiente decât în alte tipuri de arbori echilibrați, cum sunt arborii AVL
- Inversările asigură suficiența unei singure parcurgeri descendente a arborelui

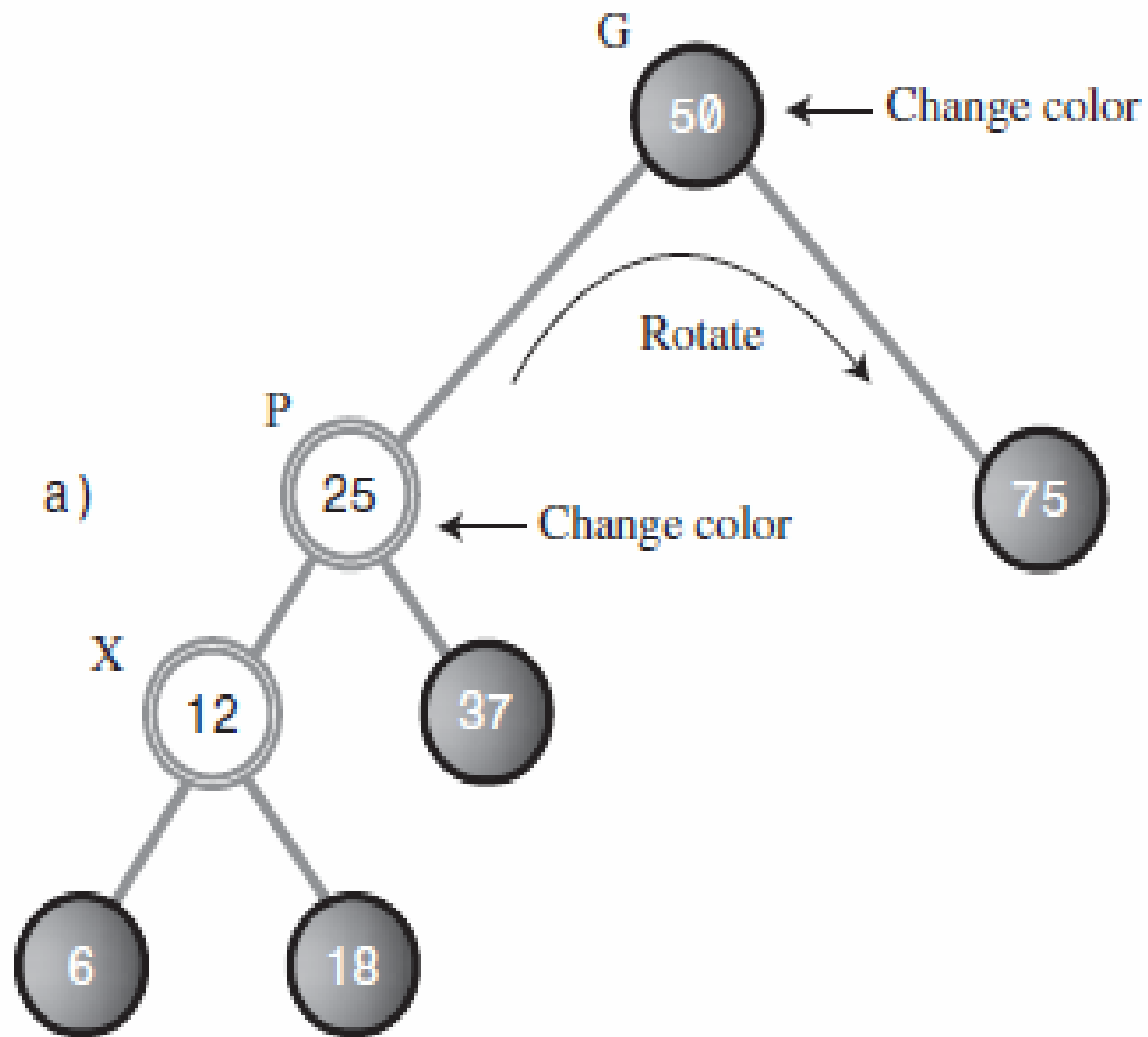


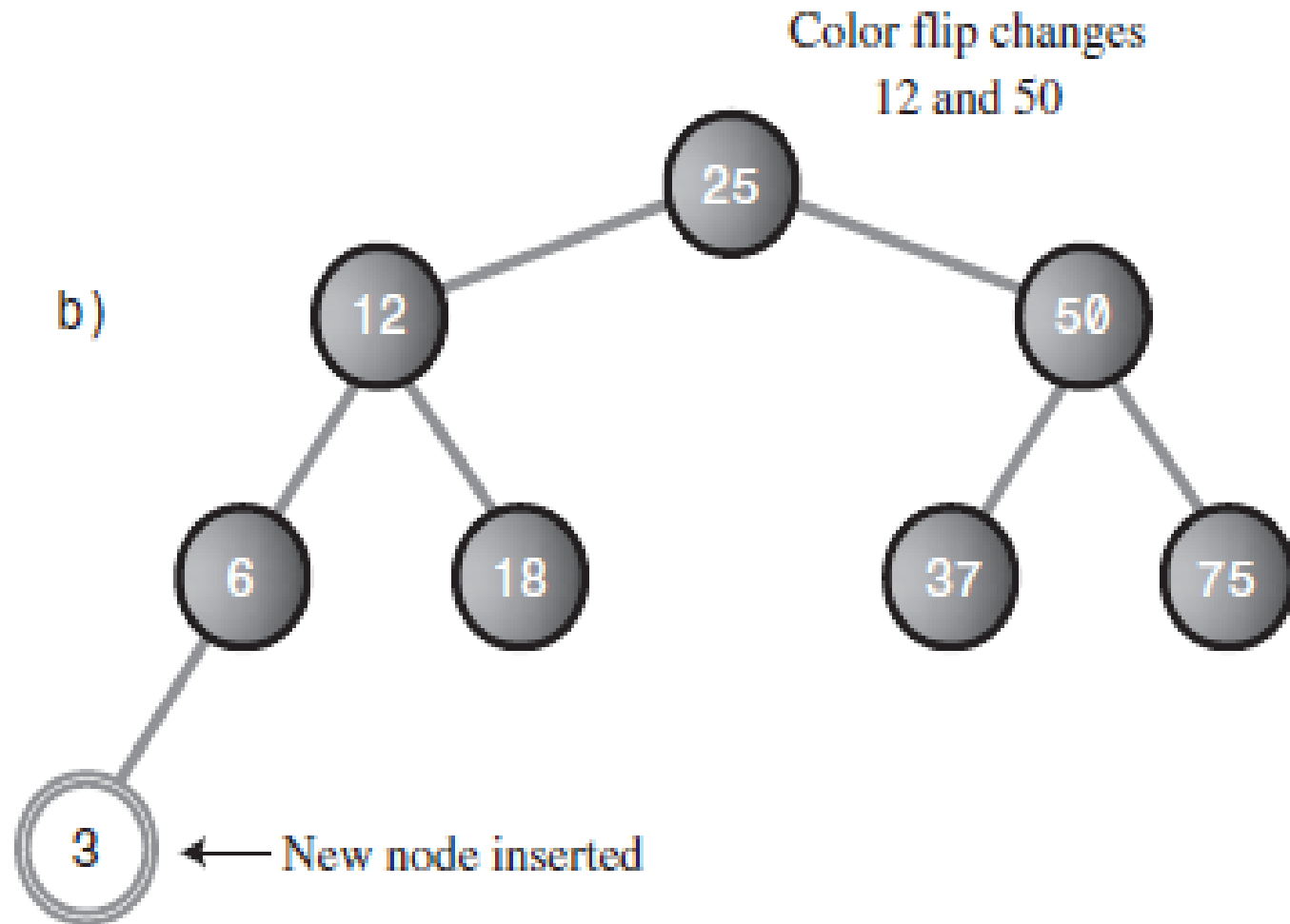
Rotații la parcurgerea descendentă

- Inversarea culorilor poate produce o încălcare a regulii 3 (un nod fiu și părintele său nu pot fi amândoi de culoare roșie)
- Această problemă se poate rezolva prin efectuarea unei rotații
- Există două posibilități – nodul implicat poate fi un nepot exterior sau interior

Nepot exterior

- Nod implicat – fiul din perechea părinte – fiu, care a provocat conflictul de colorare
- Metoda utilizată pentru a rezolva această situație este similară cu operația efectuată după inserarea unui nepot exterior
- Trebuie să efectuăm două modificări de culori și o rotație





Observații

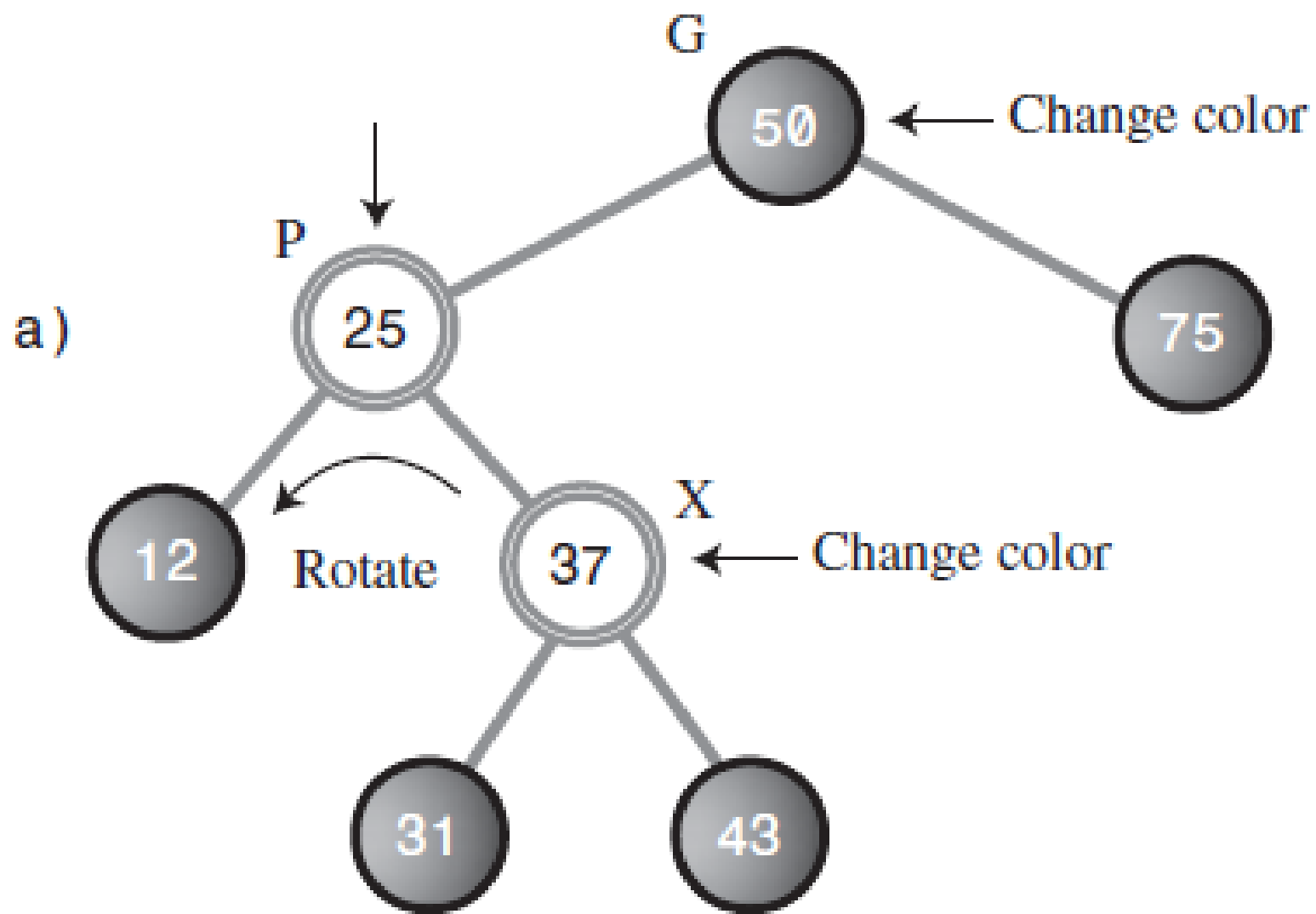
- Părintele lui X (12) este nodul P (25), iar bunicul lui X este G (50)
- 1. Se modifică culoarea bunicului G
- 2. Se modifică culoarea părintelui P
- 3. Se rotește arborele, cu bunicul lui X în vârf, în direcția care asigură ridicarea lui X (spre dreapta)

Observații

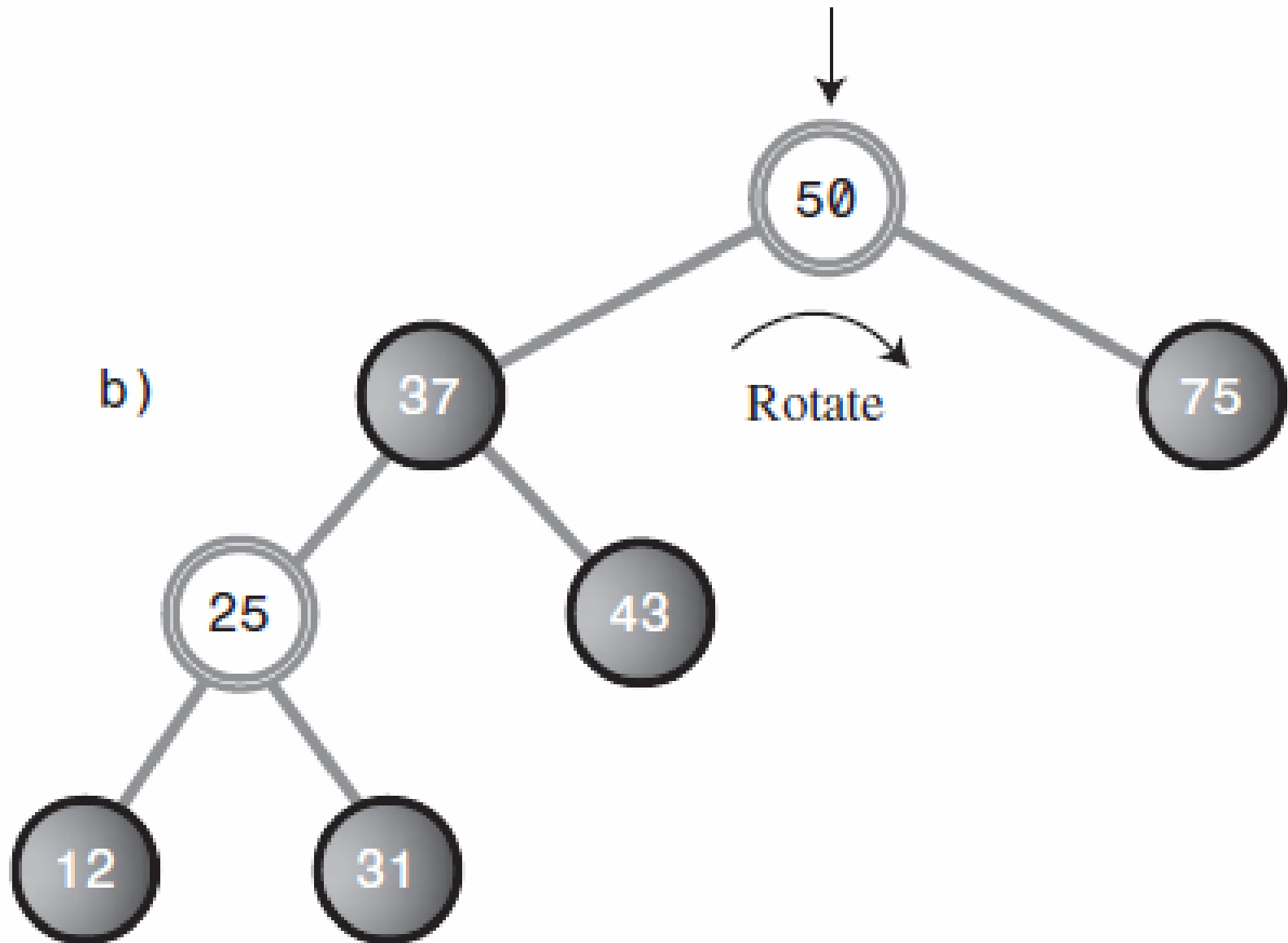
- Nodul cu valoarea 3 poate fi inserat acum în mod obișnuit
- Din cauză că părintele său, 6, este negru, inserarea se efectuează imediat

Nepot interior

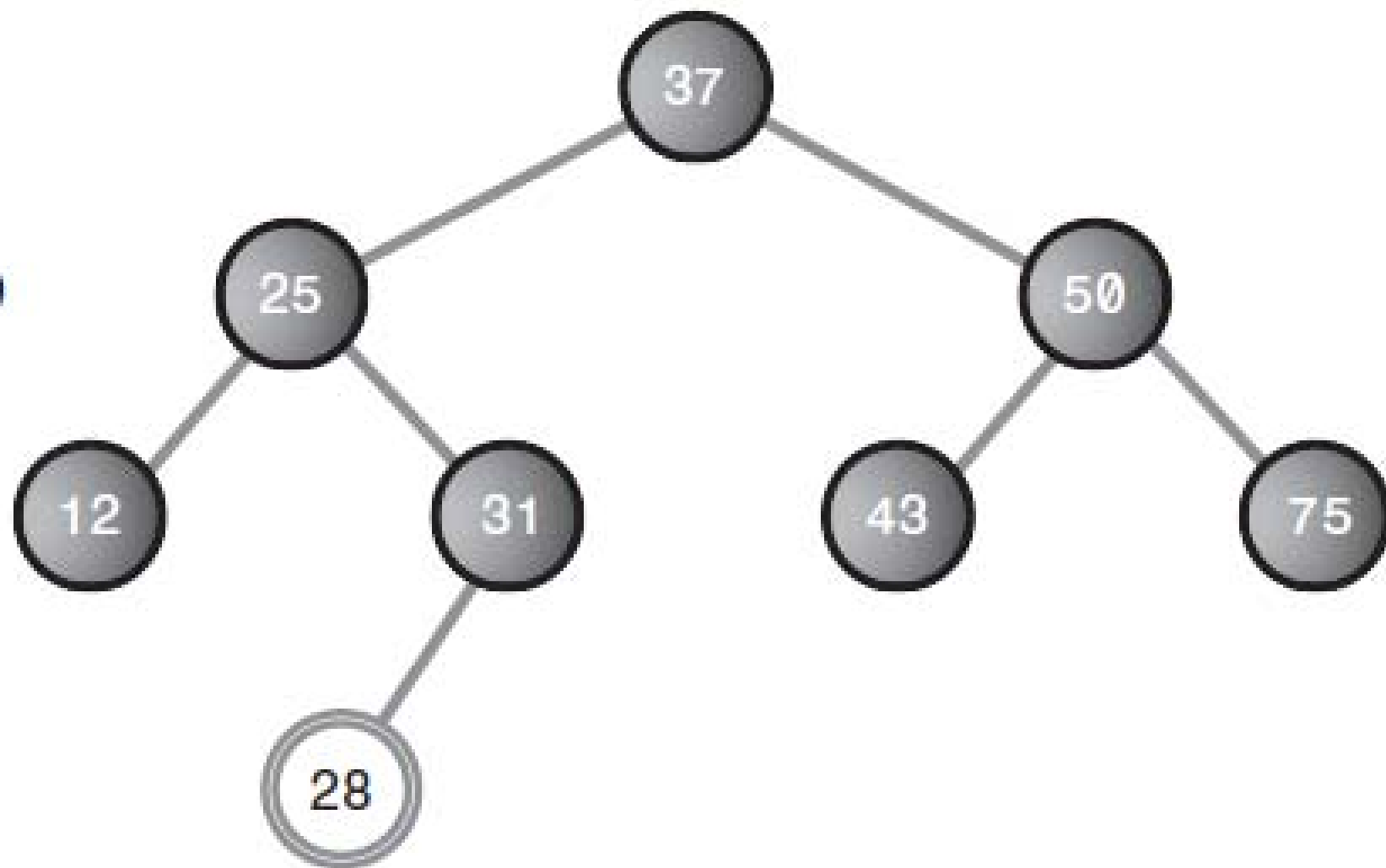
- Dacă nodul X, care produce un conflict de culoare la parcurgerea descendentă a arborelui, este un nepot interior, vor fi necesare două rotații pentru restabilirea regulilor în arbore
- Situația este asemănătoare cu operația efectuată după inserarea unui nepot interior



b)



c)





Rezolvarea conflictului de culoare

- 1. Se modifică culoarea lui G (50)
- 2. Se modifică culoarea lui X (37)
- 3. Se rotește subarborele cu P (25) în vârf, în direcția care asigură ridicarea lui X în arbore (spre stânga)
- 4. Se rotește arborele cu G în vârf, în direcția care asigură ridicarea lui X (spre dreapta)



Observații

- Acum se poate insera nodul 28
- Culoarele nodurilor 25 și 50 se vor inversa, cele două noduri devenind negre

Eficiența arborilor bicolori

- Ca și arborii binari de căutare, arborii bicolori permit efectuarea operațiilor de căutare, inserare și ștergere, într-un timp de $O(\log_2 N)$
- Pentru fiecare nod, este necesară mai multă memorie, pentru a memora și culoarea

Eficiența arborilor bicolori

- Timpii necesari pentru inserare și ștergere cresc cu o valoare constantă, necesară efectuării inversării culorilor și rotațiilor la parcurgere și la locul inserării (ștergerii)
- În medie, inserarea presupune efectuarea unei rotații
- Inserarea se efectuează într-un timp de $O(\log_2 N)$, dar este puțin mai lentă decât pentru un arbore binar de căutare

Implementarea inserării

- Se adaugă un câmp care să descrie culoarea
- Pe calea descendentă către locul de inserare, se verifică dacă nodul curent este negru, iar cei doi fii ai săi sunt amândoi roșii
- Dacă este astfel, se modifică culorile celor trei noduri (cu excepția cazului în care părintele este chiar rădăcina, care trebuie menținută neagră)



Implementarea inserării

- După o inversare a culorilor, se verifică dacă nu se încalcă regula 3
- În acest caz, se execută rotațiile corespunzătoare: una pentru un nepot exterior, două pentru un nepot interior
- Când se ajunge la un nod frunză, se inserează nodul nou, cu mențiunea că acesta trebuie să fie roșu



Implementarea inserării

- Se verifică încă o dată prezența conflictelor de culoare, executând rotațiile necesare
- Dacă se execută corect inversările de culori și rotațiile, înălțimile negre ale nodurilor se vor conserva, iar arborele va rămâne echilibrat

Arbori AVL

- **Arborii AVL** (Adelson-Velskii și Landis) reprezintă primul tip de arbori echilibrați, descoperiți în 1962
- Într-un arbore AVL, fiecare nod memorează o altă informație suplimentară: diferența dintre înălțimea subarborelui său stâng și cea a subarborelui drept
- Diferența nu trebuie să depășească 1

Arbori AVL

- După inserare, se verifică rădăcina subarborelui aflat cel mai jos și în care s-a inserat noul nod
- Dacă diferența dintre înălțimile celor doi fii este mai mare decât 1, se execută o rotație simplă sau dublă, pentru a egaliza cele două înălțimi



Arbori AVL

- Algoritmul se deplasează apoi cu un nivel mai sus în arbore, egalizând din nou înălțimile subarborilor, dacă este cazul
- Acest proces continuă până la atingerea rădăcinii arborelui



Arbori AVL

- Timpul de căutare este de ordinul $O(\log_2 N)$, deoarece echilibrarea arborelui este garantată
- Pentru inserare sau ștergere, este necesară efectuarea a două parcurgeri: una descendentă, pentru determinarea locului de inserare, și una ascendentă, pentru reechilibrare
- De aceea, arborii AVL sunt mai puțin eficienți decât arborii bicolori



Arbori multicăi

- Un arbore multicăi este un arbore echilibrat, în care fiecare nod poate avea mai mult de doi fii
- Un dezavantaj al acestor arbori este că fiecare nod ocupă mai multă memorie decât la arborii binari, deoarece trebuie să memoreze câte un pointer către fiecare din fiii săi

Concluzii

- Menținerea echilibrării unui arbore binar asigură efectuarea operației de căutare a unui nod din arbore într-un timp minim
- Inserarea unor date deja sortate va genera un arbore cu un grad de dezechilibru maxim, în care căutarea se va efectua într-un timp $O(N)$

Concluzii

- Modurile permise, în care pot fi dispuse nodurile dintr-un arbore bicolor, sunt specificate prin reguli de colorare
- Aceste reguli se aplică pentru operațiile de inserare și ștergere a unui nod
- O inversare de culori schimbă un nod negru cu doi fii roșii, într-un nod roșu, cu doi fii negri

Concluzii

- Într-o rotație, un nod este desemnat ca nod din vârf
- O rotație spre dreapta deplasează nodul din vârf în locul fiului său drept, iar fiul stâng al nodului din vârf, în locul părintelui său
- O rotație spre stânga deplasează nodul din vârf în locul fiului său stâng, iar fiul drept al nodului din vârf, în locul părintelui său



Concluzii

- Inversările de culori și, în unele cazuri, rotațiile, se utilizează la parcurgerea descendentă a arborelui, pentru căutarea locului de inserare
- Aceste inversări simplifică restabilirea corectitudinii la colorare a arborelui, după efectuarea inserării



Concluzii

- După inserarea unui nod, se verifică din nou conflictele de culoare
- Dacă se detectează o încălcare, se execută rotațiile corespunzătoare pentru asigurarea corectitudinii arborelui
- În urma acestor operații, arborele devine echilibrat, sau cel puțin aproape echilibrat



Concluzii

- Adăugarea informației necesare echilibrării într-un arbore are un impact negativ minor asupra performanțelor medii, evitând în schimb degradarea acestora, în cazul defavorabil în care datele sunt inițial sortate