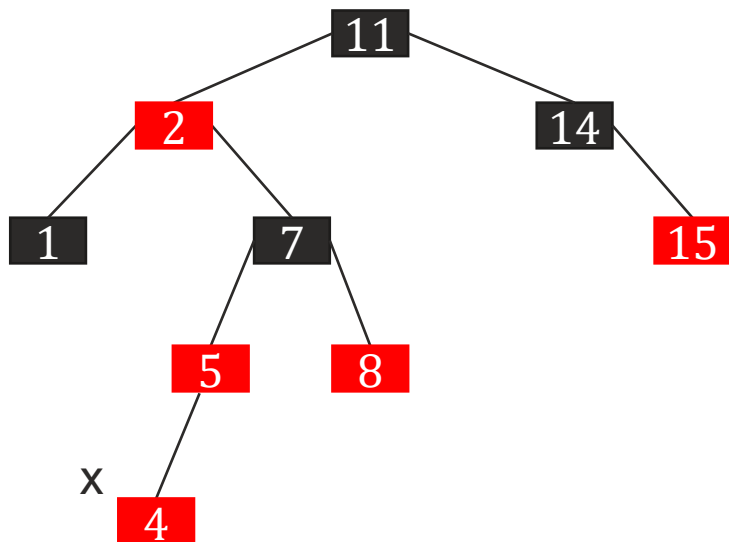


ADATSZERKEZETEK ÉS ALGORITMUSOK

Piros-Fekete Fa Beszúrás
„Hierarchikus adatszerkezetek, keresési fák”

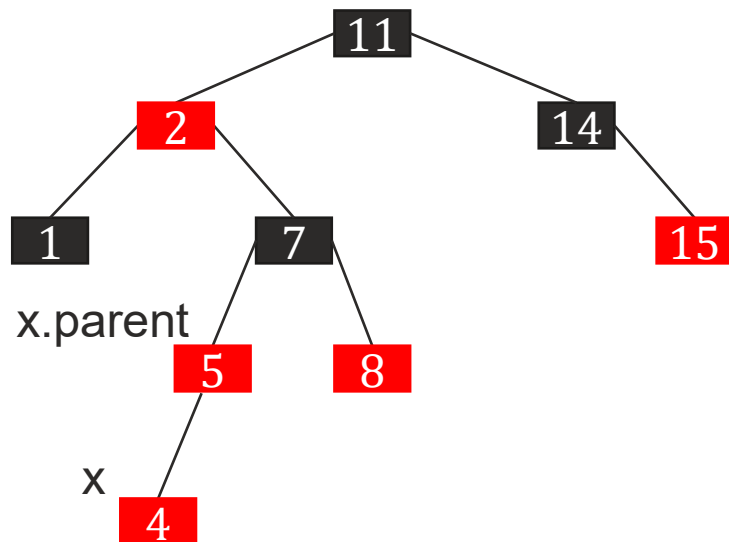
Piros-Fekete fák – beszúrás

- Egy új csúcs beszúrása
 - Szükség van a PF fa tulajdonság **helyreállítására**
 - Szúrjuk be a 4-et
 - Keressük meg a helyét a bináris keresőfában
 - Színezzük pirosra – Miért?



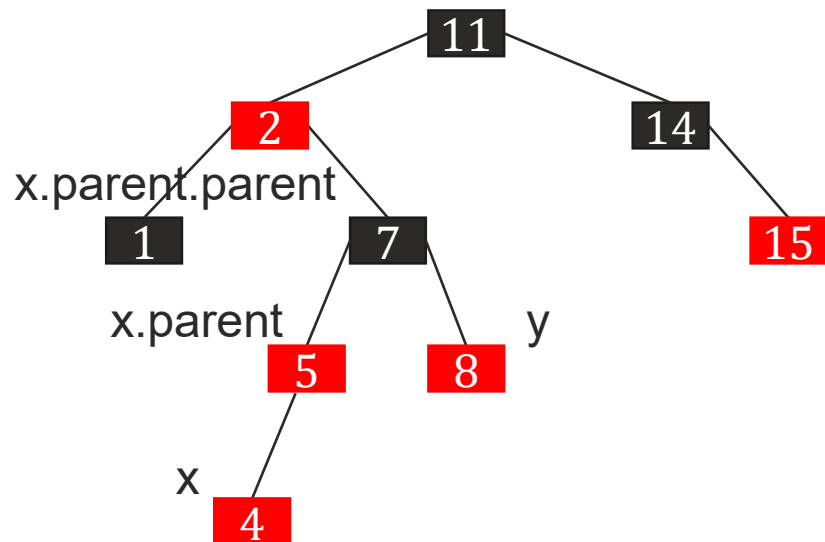
Piros-Fekete fák – beszúrás

- Állítsuk helyre a PF tulajdonságot
 - Amíg az x és szülője egyaránt **piros**, vagy el nem értük gyökeret helyreállítási lépéseket kell tenni



Piros-Fekete fák – beszúrás

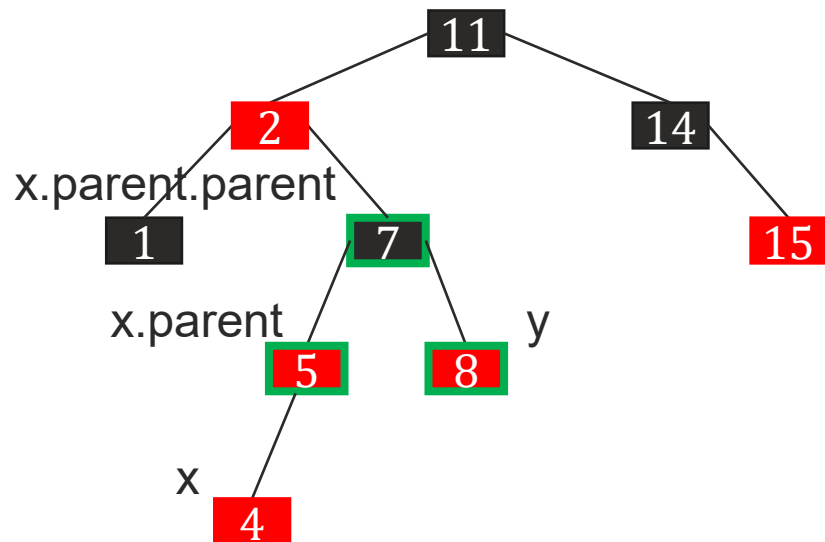
- Állítsuk helyre a PF tulajdonságot
 - Jelöljük meg a csúcsokat
 - A szülő balra van a nagyszülőtől, és az x balra van a szülőtől
 - Itt a szülő azonos oldali gyereke a nagyszülőnek, de lehetne ellenkező oldali is
 - A nagyszülő jobbgyereke (nagybácsi) legyen az y



Piros-Fekete fák – beszúrás

- 1. eset

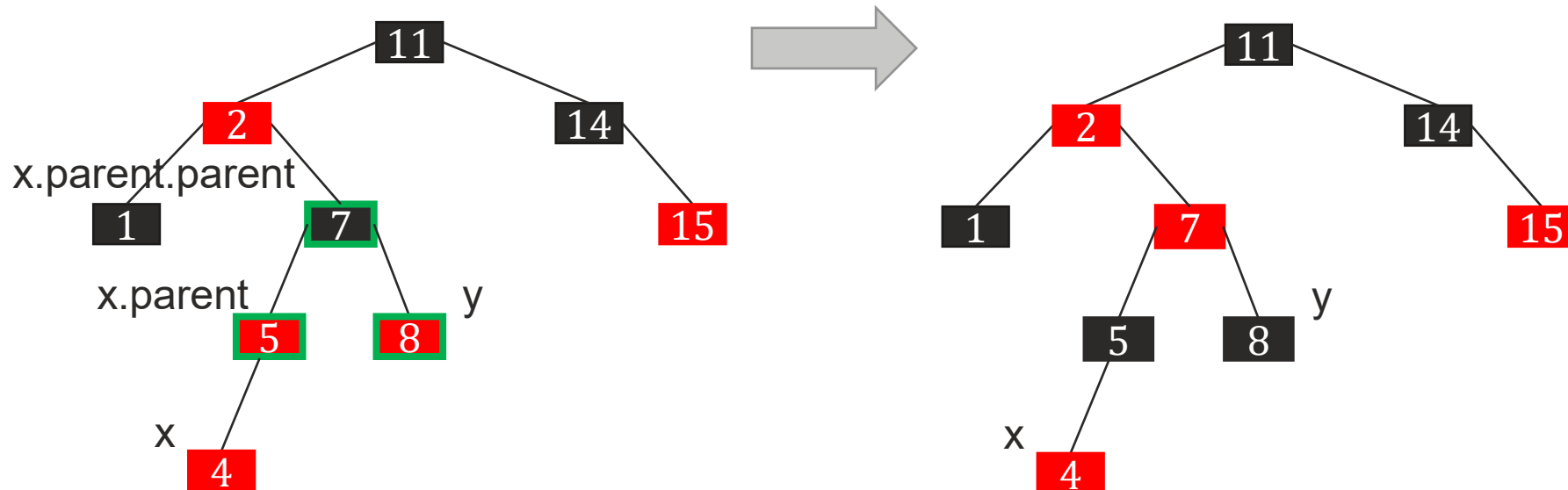
- Ha a nagybácsi színe **piros**, cseréljük meg y, a nagyszülő és a szülő színét



Piros-Fekete fák – beszúrás

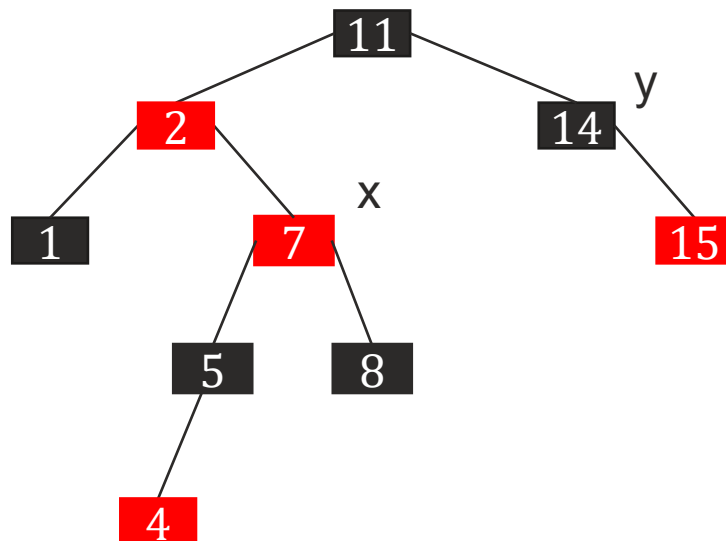
- 1. eset

- Ha a nagybácsi színe **piros**, cseréljük meg y, a nagyszülő és a szülő színét



Piros-Fekete fák – beszúrás

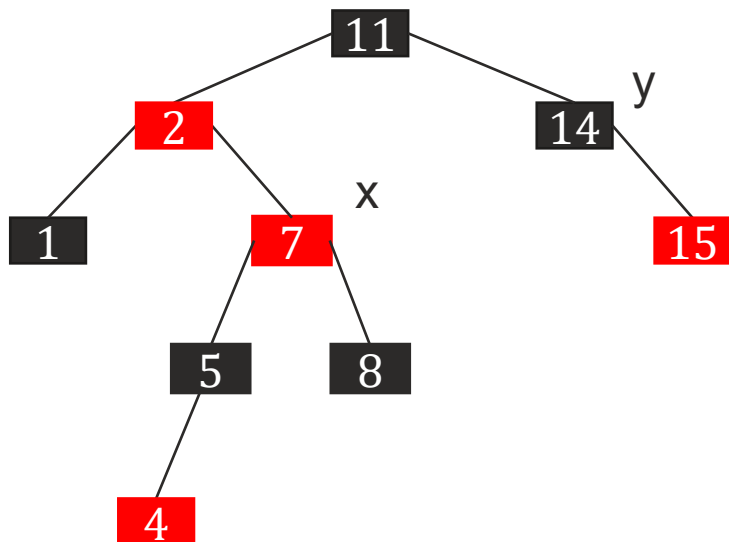
- Az előzőt folytatva
 - x jelölje az eddigi x nagyszülőjét
 - Ebben a helyzetben az x szülője megint balgyerek
 - Az y pedig jelölve továbbra is az x nagybácsiját
 - Ez most nem piros, hanem fekete, ami egy másik, új eset



Piros-Fekete fák – beszúrás

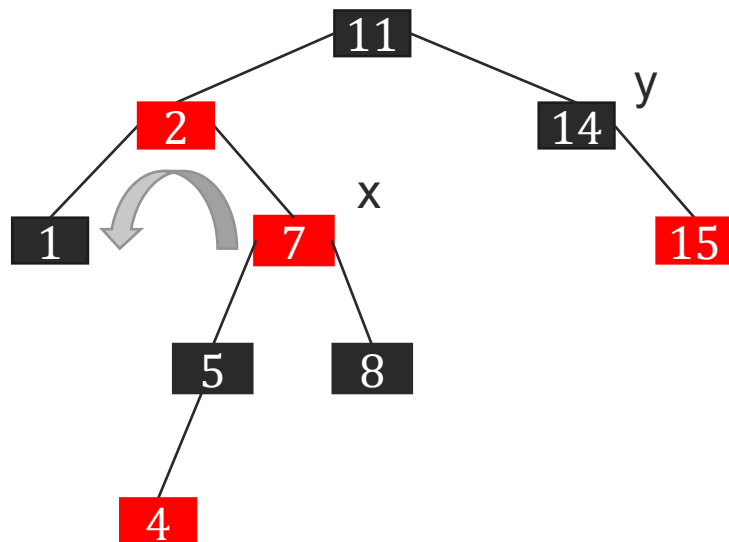
- 2. eset

- Ha az x nagybácsija fekete, az x és szülője is **piros** és a szülő ellenkező oldali gyereke a nagyszülőnek, mint az x a szülőnek ...
 - Megjegyzés: ez az eset függetlenül is előfordulhat az előzőtől



Piros-Fekete fák – beszúrás

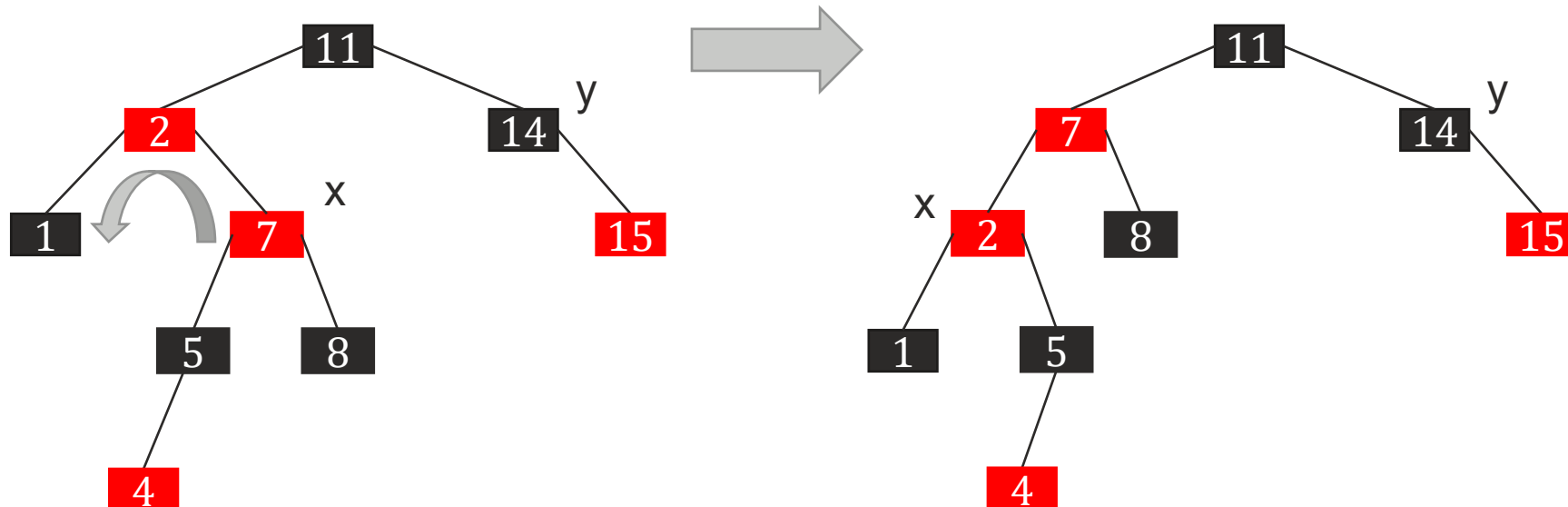
- 2. eset
 - ... akkor forgassunk x és szülője körül balra ...



Piros-Fekete fák – beszúrás

- 2. eset

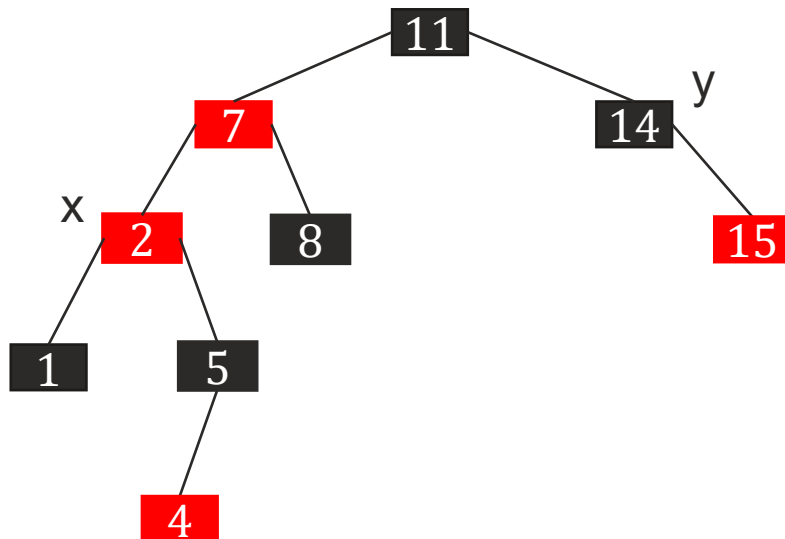
- ... jelöljük át, legyen x az eddigi x szülője és
- forgassunk x és gyereke körül balra ...
 - Ezzel az x és szülője, valamint a szülő és nagyszülő közötti oldaliságot azonossá tettük, mást még nem oldottunk meg.



Piros-Fekete fák – beszúrás

- 3. eset

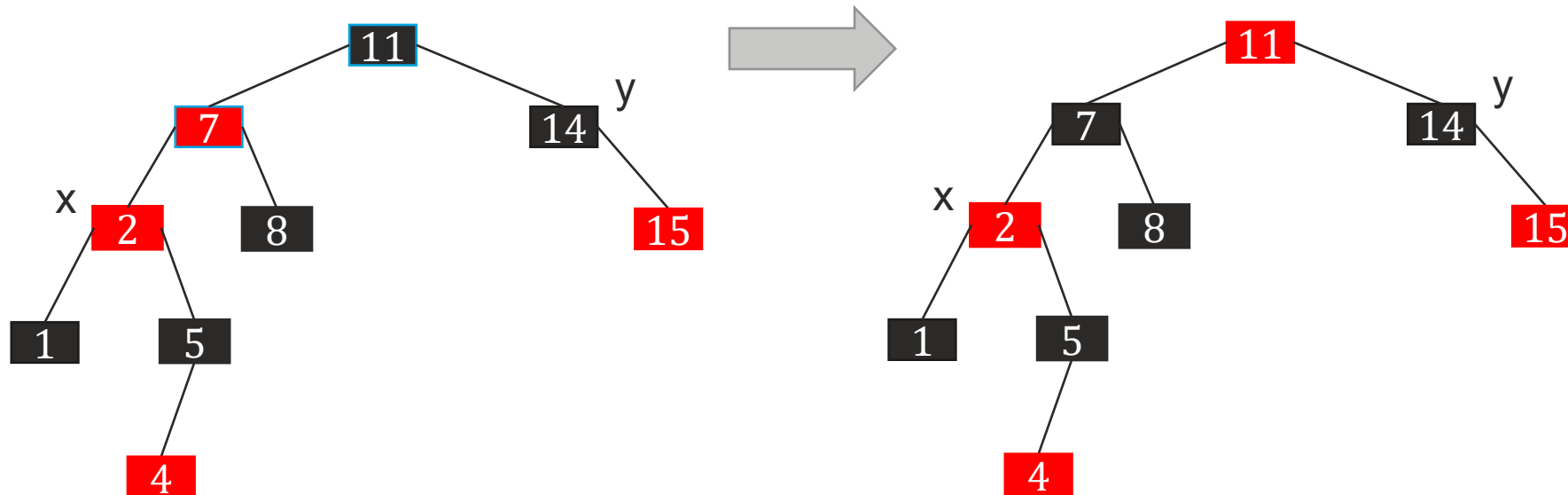
- Ha az x nagybácsija fekete, az x és szülője is piros és a szülő azonos oldali gyereke a nagyszülőnek, mint az x a szülőnek ...
 - Megjegyzés: ez az eset függetlenül is előfordulhat az előzőtől



Piros-Fekete fák – beszúrás

- 3. eset

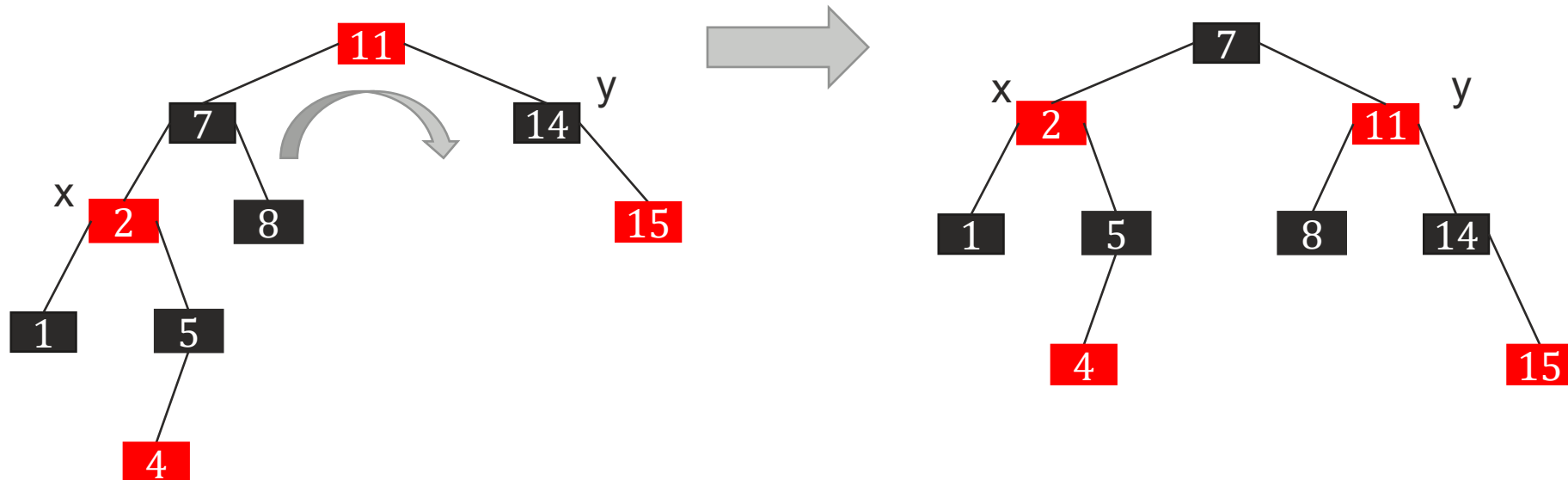
- ... akkor kicseréljük a színeket a szülő és a nagyszülő között ...



Piros-Fekete fák – beszúrás

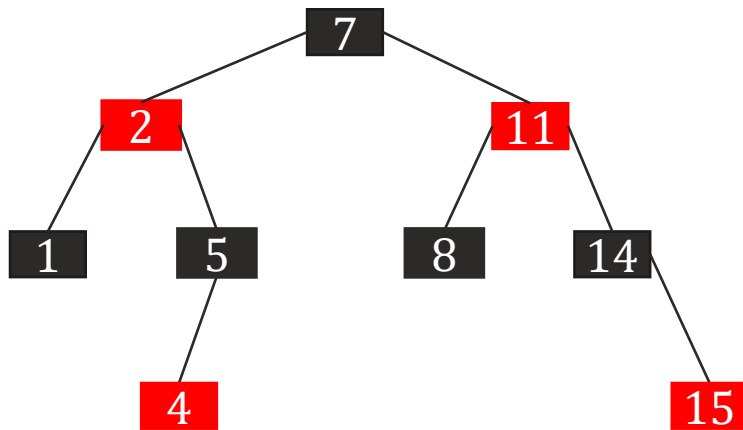
- 3. eset

- ... akkor kicseréljük a színeket a szülő és a nagyszülő között
- ... és forgatunk a szülő és a nagyszülő mentén jobbra



Piros-Fekete fák – beszúrás

- 3. eset
 - ... akkor kicseréljük a színeket a szülő és a nagyszülő között
 - ... és forgatunk a szülő és a nagyszülő mentén jobbra
 - Ez most már PF fa



Piros-Fekete fák – beszúrás

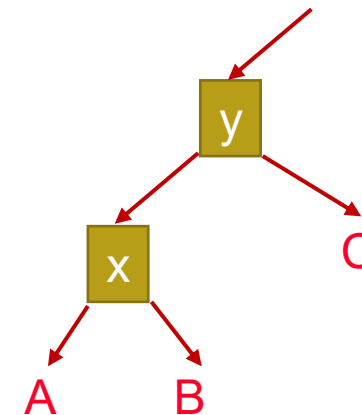
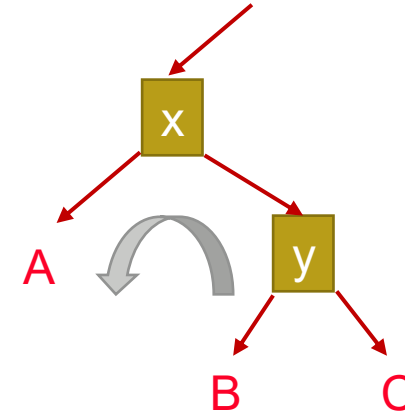
- Van egy ekvivalens esetszétválasztás, amikor a szülő a nagyszülő jobbján van!
 - Természetesen akkor az irányok megváltoznak, megfordul
 - Az azonos oldaliság vizsgálata megmarad
- Mindig addig megyünk felfelé, amíg az x szülője is piros

Piros-Fekete fák – algoritmusok

- Forgatások:
 - BALRA-FORGAT, JOBBRA-FORGAT
 - Mindkettő megőrzi a kulcsok inorder sorrendjét
 - A művelet időigénye $\mathcal{O}(1)$
 - A BALRA-FORGAT feltételezi, hogy $\text{jobb}[x] \neq \text{NIL}$

BALRA-FORGAT(T, x)

```
y ← jobb[x]; jobb[x] ← bal[y]
if bal[y] ≠ NIL
    then szülő[bal[y]] ← x
szülő[y] ← szülő[x]
if szülő[x] = NIL
    then gyökér[T] ← y
else if x = bal[szülő[x]]
    then bal[szülő[x]] ← y
    else jobb[szülő[x]] ← y
bal[y] ← x
szülő[x] ← y
```



Piros-Fekete fák – algoritmusok

- Lehetséges esetek annak megfelelően, hogy az x , illetve x szülője bal- vagy jobbgyerek-e, és hogy milyen színű a nagybácsi, illetve a nagyszülő:
 1. eset: szülő és nagybácsi piros, nagyszülő fekete
 - oldaltól nem függ
 2. eset: szülő piros, nagybácsi fekete, x jobbgyerek
 - ellenkező oldali gyerek (balra forgatok, így 3.eset)
 3. eset: szülő piros, nagybácsi fekete, x balgyerek
 - azonos oldali gyerek

Piros-Fekete fák – algoritmusok

PF-Fába-beszúr(T,x)

Fába-beszúr(T,x) -- beszúrom a bináris keresőfába

szín[x]←PIROS

while x ≠ gyökér[T] and szín[szülő[x]]=PIROS do

 if szülő[x] = bal[szülő[szülő[x]]] then

 y ← jobb[szülő[szülő[x]]]

 if szín[y] = PIROS then

 szín[szülő[x]]← FEKETE --1. eset

 szín[y] ← FEKETE --1. eset

 szín[szülő[szülő[x]]]←PIROS --1. eset

 x←szülő[szülő[x]] --1. eset

 else if x = jobb[szülő[x]] then

 x←szülő[x] --2. eset

 BALRA-FORGAT(T,x) --2. eset

 szín[szülő[x]]← FEKETE --3. eset

 szín[szülő[szülő[x]]]←PIROS --3. eset

 JOBBRA-FORGAT(T, szülő[szülő[x]]) --3. eset

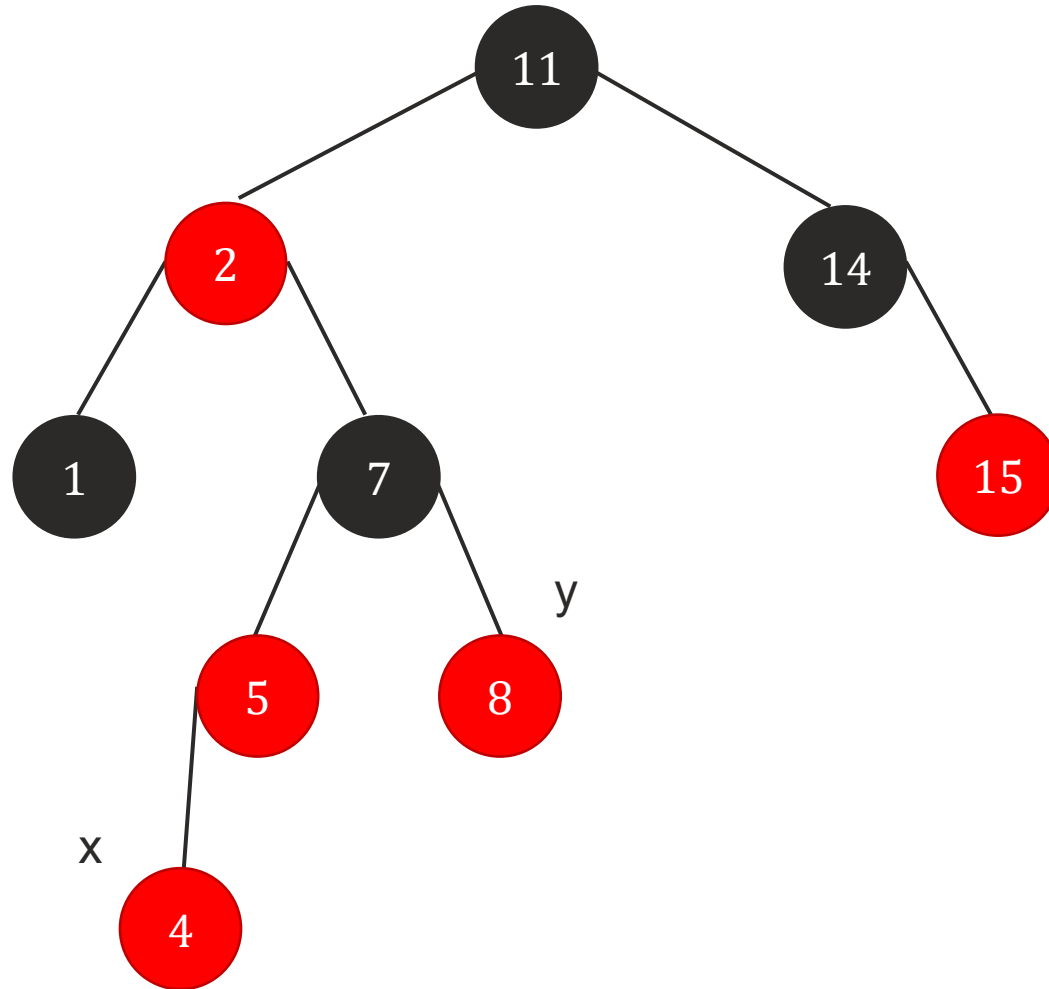
else

 -- Az előzőeknek szerint, csak az oldalak fordítva

szín[gyökér[T]]← FEKETE

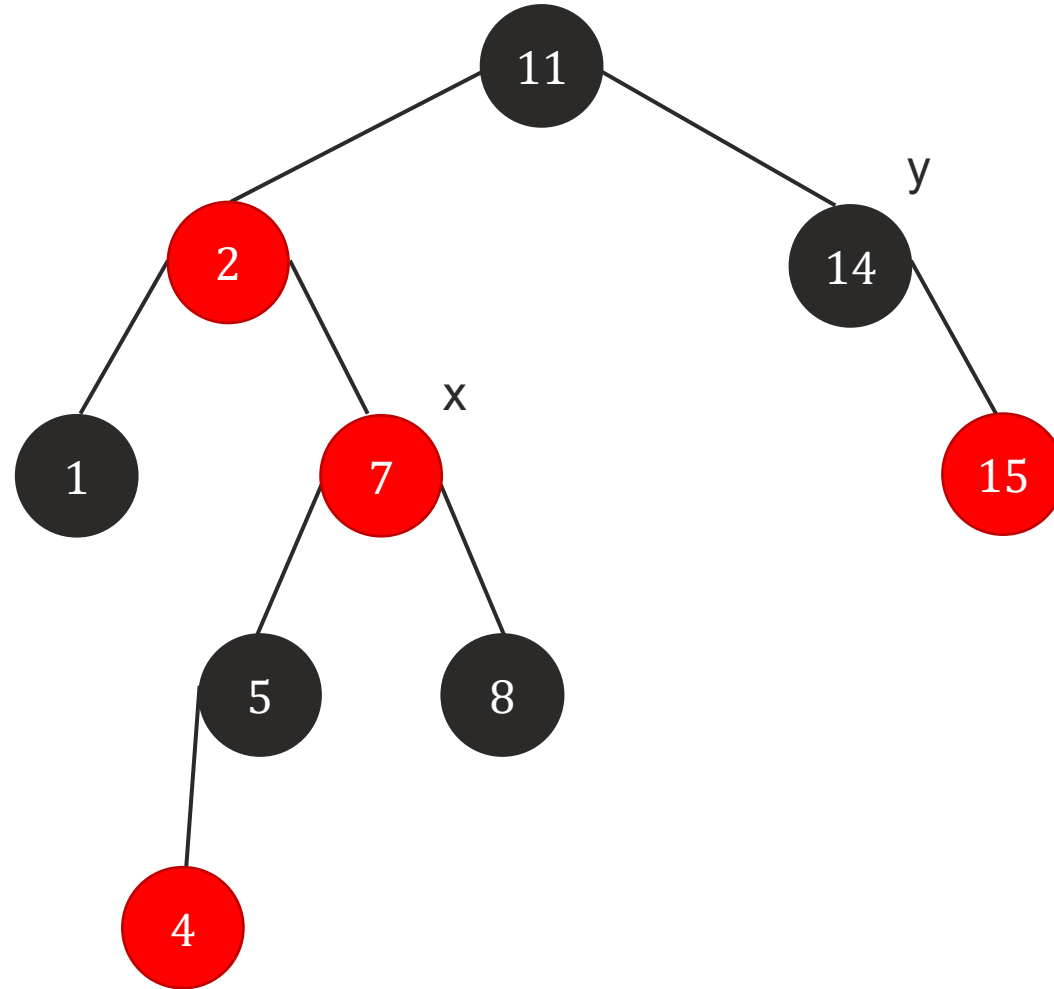
Példa

- Fába-beszúrral betettük x-et, és pirosra színeztük.
- Mivel a szülője is piros, így a 3. tulajdonság nem teljesül.
- y – a jobb nagybácsi is piros, tehát 1. eset áll fenn ...



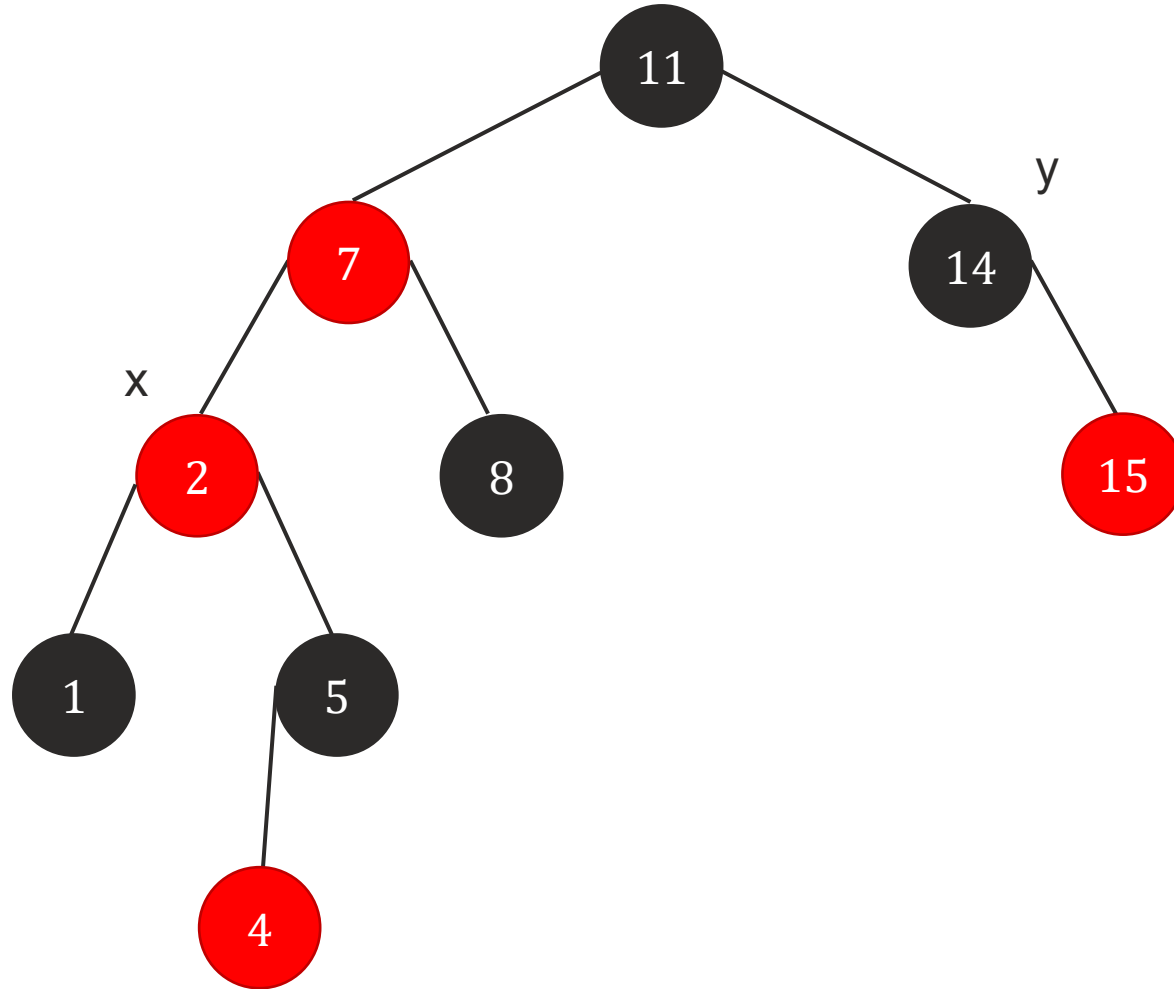
Példa

- x feljebb kerül, x és az apja szintén piros, de most a jobb nagybácsi fekete, és x jobb fia szülőjének
- 2. eset, balraforgatás ...



Példa

- most már x bal fia szülőjének
- ez a 3. eset, így x nagyszülőjének jobbraforgatása és a színek cseréje jön ...



Példa

- ... és kész a piros-fekete fa

