

Програмирање и програмски језици

4-СМ

Јелица Васиљевић

Бинарна стабла – Брисање из претраживачког стабла

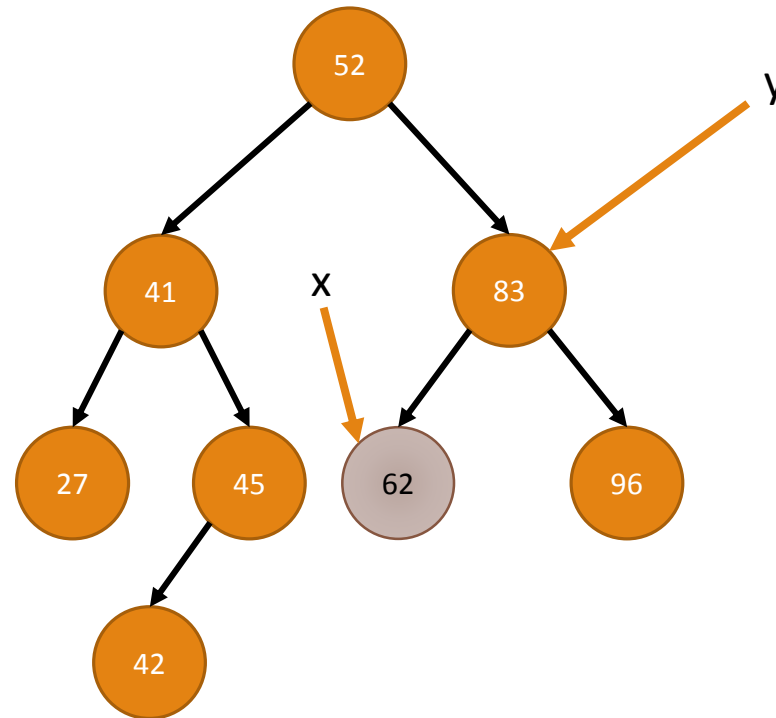
- Брисање чвора из **претраживачког** бинарног стабла може мање или више утицати на изглед стабла.
- Означимо чвор који желимо да обришемо са **x** а чвор који је његов родитељ са **y** .
- Након брсања чвора **x** , његови потомци се морају **реорганизовати** и повезати са родитељским чвором **y** тако да стабло и даље буде претраживачко (али и стабло, уопште)

Бинарна стабла – Брисање из претраживачког стабла

- Брисање чвора из стабла можемо раздвојити на три случаја, у зависности од броја потомака чвора :
 - Чвор је лист – нема потомке
 - Чвор има једног наследника
 - Чвор има два наследника

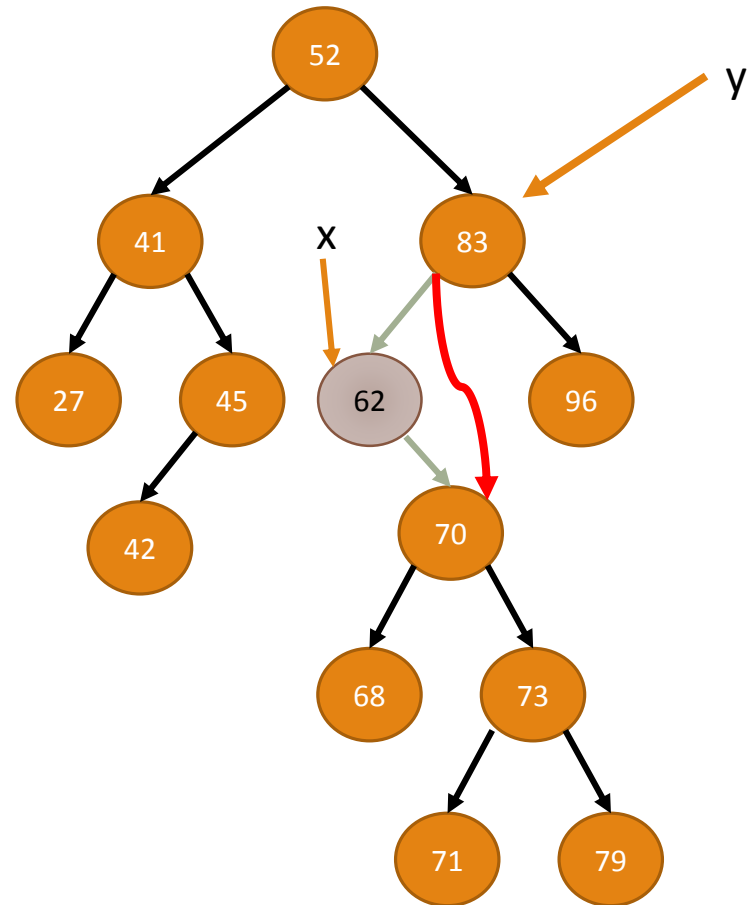
Бинарна стабла – Брисање из претраживачког стабла

- Брисање чвора који нема потомака – *листа*
 - Најједноставније брисање
 - Одговарајући потомак чвора *y* (леви или десни) се поставља на NIL



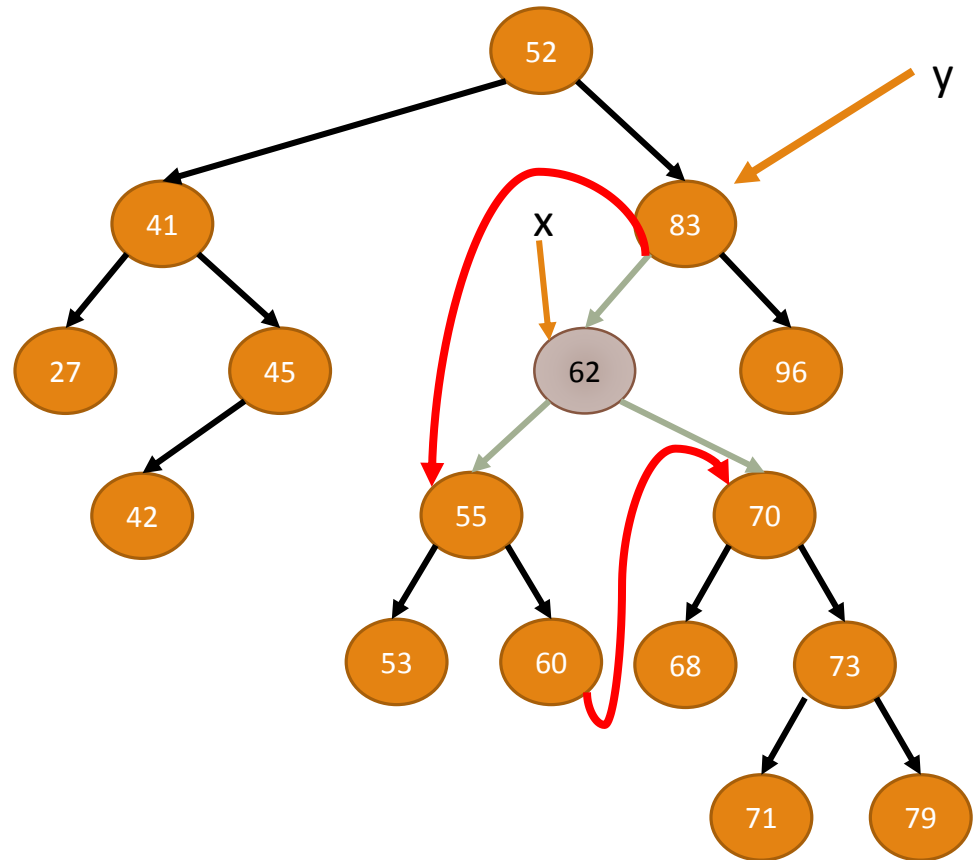
Бинарна стабла – Брисање из претраживачког стабла

- Брисање чвора који има једног потомка
- Одговарајући потомак чвора y је сада једини постојећи потомак чвора x



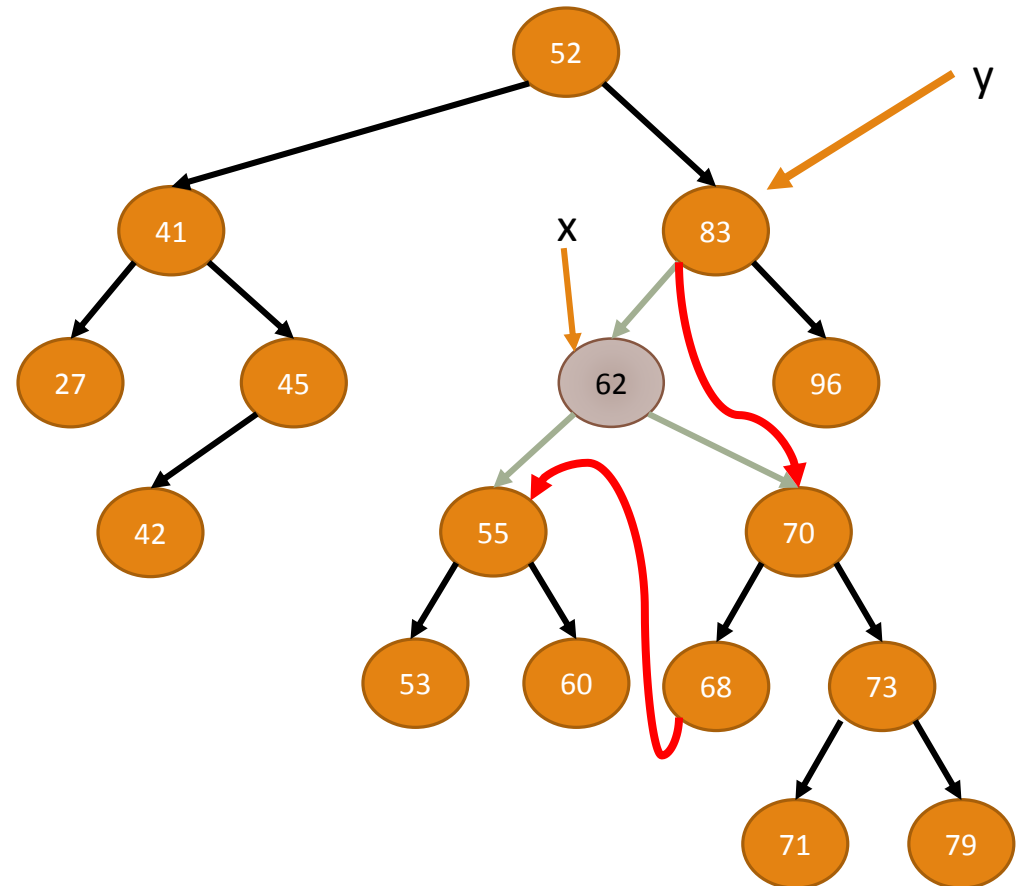
Бинарна стабла – Брисање из претраживачког стабла

- Брисање чвора који има два потомка
 - Проналазимо **крајњи десни потомак левог потомка** чвора који се брише
 - Његов **десни** показивач показује на **десно** подстабло чвора који се брише
 - Нови потомак чвора **y** је корен левог подстабла чвора који се брише



Бинарна стабла – Брисање из претраживачког стабла

- Брисање чвора који има два потомка (2. начин)
 - Проналазимо **крајњи леви потомак десног потомка** чвора који се брише
 - Његов **леви** показивач показује на **лево** подстабло чвора који се брише
 - Нови потомак чвора **y** је корен десног подстабла чвора који се брише



Бинарна стабла – Брисање из претраживачког стабла

1. Дефинисати пролошки предикат који из већ постојећег претраживачког бинарног стабла избацује чвор који садржи одређену вредност
 - Дефинисати предикат којим се корен стабла поставља на врх листе чињеница
 - Ради лакшег рада, дефинисати предикат којим се бришу сви чворви стабла као и предикат којим се изластавају сви чворови стабла.

Бинарна стабла – Брисање из претраживачког стабла

1. Решење :

```
brisi(X):-stablo(Levo,X,Desno),srediDecu(Levo,X,Desno),srediRoditelja(Levo,X,Desno), retract(stablo(Levo,X,Desno)),prviKoren.
```

```
srediDecu(nil,_,_).
```

```
srediDecu(Levo,_,Desno):-nadjiDesniList(Levo,Desno).
```

```
nadjiDesniList(Cvor,Desno):-stablo(nil,Cvor,nil),retract(stablo(nil,Cvor,nil)),assertz(stablo(nil,Cvor,Desno)).
```

```
nadjiDesniList(Cvor,Desno):-stablo(_,Cvor,D),nadjiDesniList(D,Desno).
```

```
srediRoditelja(nil,X,Desno):-stablo(X,Roditelj,D1),retract(stablo(X,Roditelj,D1)),assertz(stablo(Desno,Roditelj,D1)).
```

```
srediRoditelja(nil,X,Desno):-stablo(L1,Roditelj,X),retract(stablo(L1,Roditelj,X)),assertz(stablo(L1,Roditelj,Desno)).
```

```
srediRoditelja(Levo,X,_):-stablo(X,Roditelj,D1),retract(stablo(X,Roditelj,D1)),assertz(stablo(Levo,Roditelj,D1)).
```

```
srediRoditelja(Levo,X,_):-stablo(L1,Roditelj,X),retract(stablo(L1,Roditelj,X)),assertz(stablo(L1,Roditelj,Levo)).
```

```
srediRoditelja(____).
```

```
prviKoren :- stablo(L,K,D),not(stablo(K,____)),not(stablo(____,K)),retract(stablo(L,K,D)),asserta(stablo(L,K,D)).
```

```
brisi :- retractall(stablo(____)).
```

```
ispis :- listing(stablo).
```