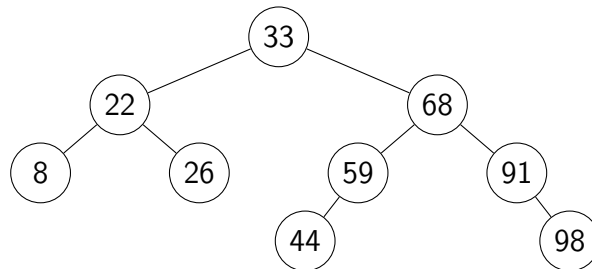


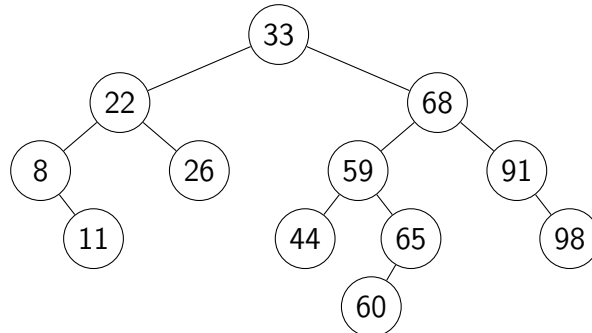
# 1. gyakorlat – Bináris keresőfák

2018. február 3.

1. Vegyük az alábbi bináris keresőfát!



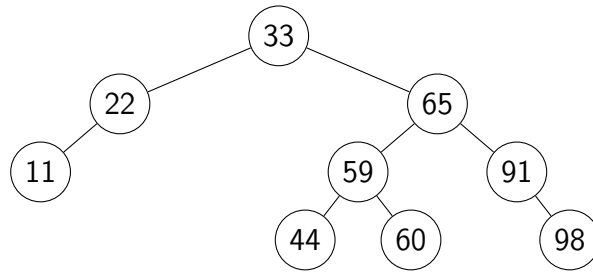
- a) Keressük meg a fában a 11, 89, 44, 90 kulcsokat!
- b) Szűrjük be a fába a 11, 65, 60 kulcsokat!



- c) Hogyan keresnénk meg egy  $r$  gyökerű (rész)fa maximális elemét?  
Legkisebb elem: gyökértől kezdve végig balra megyünk a fában.
- d) Hogyan keresnénk meg  $x$  megelőzőjét?

1. Ha  $x$ -nek van bal fia, akkor a bal részfa maximális eleme lesz a megelőző.
2. Ha  $x$ -nek nincs bal fia, akkor addig megyünk fel a fában, amíg az első  $x$ -től kisebb kulcsot meg nem találjuk.
3. Ha a gyökérig eljutva sem találunk egyetlen  $x$ -nél kisebb kulcsot, akkor nem található megelőzője  $x$ -nek a BKF-ban (vagyis  $x$  a fában található legkisebb elem).

Megjegyzés: az előző eljárások értelemszerű módosításával megkaphatjuk az  $r$  gyökerű részfa minimális elemét, illetve az  $x$  kulcs rákövetkezőjét meghatározó algoritmusokat.  
e) Töröljük a beszúrások után előálló fából a 26, 8, 68 kulcsokat!



2. Tegyük fel, hogy egy bináris keresőfában a 15-ös elemet keressük. Lehetséges keresési sorozat-e az alábbi: 20, 9, 12, 8, 15?

Nem, ugyanis 15 nagyobb, mint a 12, ezért a 8-as a 12 jobb fia kellett, hogy legyen, de ez ellentmondás, hiszen a 12 jobb részfájában nem lehetnek nála kisebb elemek.

3. Írassuk ki a fa elemeit kulcs szerint növekvő sorrendben!

```

void inorder(x) {
    if(x!=nil) {
        inorder(x.bal);
        print(x.kulcs);
        inorder(x.jobb);
    }
}
  
```

4. Szűrjük be egy üres bináris keresőfába a következő elemeket a megadott sorrendben: 1,2,3,4,5,6,7.

