

Övningar i binära träd, grafer och rekursiva datastrukturer

Förstå koncept:

- 1) Vi har pratat om rekursiva algoritmer tidigare, men vad är en rekursiv datastruktur för någonting?
- 2) Vilken tidskomplexitet har en sökning i ett rödsvart träd? Varför?
- 3) Vad är risken med obalanserade binära träd?
- 4) Ge exempel på några datastrukturer i Java som använder sig av balanserade träd.
- 5) Vad är en graf?
- 6) Ge några exempel på användningsområden för grafer.
- 7) Vad är garbage collection för någonting? Varför används det i Java?
- 8) Vad är några av fördelarna med en foreach-loop (enhanced for loop)?
- 9) Vad är några av nackdelarna?
- 10) Vilka är de tre vanligaste sätten att traversera ett binärt sökträd? Vad är skillnaden mellan dem?
- 11) Vad är Dijkstras algoritm för någonting? Hur fungerar den?
- 12) Vad är undantag och hur kan man hantera dem?
- 13) Vad är stack unwinding för någonting?
- 14) Vad är skillnaden mellan checked och unchecked exceptions (märkta och omärkta undantag)?
- 15) Vad är tvåstegsinitialisering och varför ska man undvika det?
- 16) Vad är en girig algoritm? Hur fungerar den?