

Fråga 1: Vilket asymptotiskt samband mellan exekveringstid och storlek på BST:t (dvs tidskomplexitet) bör studien resultera i enligt teorin?

Svar:

I ett binärt sökträd som byggs upp med slumpmässigt insatt data förväntas enligt teorin en tidskomplexitet på  $O(\log n)$ .

Resultatet från studie 1 stämmer överens med teorin. När storleken på det binära sökträdet dubblas så ökar exekveringstiden långsammare än linjärt. Detta syns tydligt i kvoterna i tabell 1 som ökar i början men minskar när  $n$  ökar.

Om sökningen hade haft en linjär tidskomplexitet  $O(n)$  hade kvoten ökat med 2 vid varje dubbling av  $n$ . Vilket inte är fallet här och detta indikerar ett logaritmiskt samband mellan exekveringstiden och trädets storlek.

Nodes	Average ms	Quota
10000	298	
20000	385	1,292
40000	490	1,273
80000	701	1,431
160000	1008	1,438
320000	1351	1,340
640000	1700	1,258
1280000	2070	1,218
2560000	2431	1,174

Tabell 1: Data från första studien.

Fråga 2: Hur kan man presentera data tydligare för att avgöra vilket asymptotiskt samband det handlar om?

Svar:

Ett tydligare sätt att presentera data är att visualisera data genom att skapa ett diagram från tabellen. Detta kan uppnås genom att plotta exekveringstiden mot noder. Detta gör så att det blir lättare att se om kurvan växer logaritmiskt., linjärt eller på något annat sätt. Ett ännu tydligare sätt är att bygga ett diagram på en log-log graf eftersom olika asymptotiska samband ofta ger olika rak linje beteende.

Fråga 3: Hue skiljer sig exekveringstiden i studie 2 jämfört med studie 1? Vad är orsaken?

Svar:

Exekveringstiden i studie 2 är mycket långsammare än i studie 1 vilket kan ses i tabell 2. Jämförelsevis så exekverades tiotusen noder i första studien på 298 ms medan tiotusen noder exekverades på 30191 ms. Kvoten i studie två är dessutom cirka 2 när man dubblar antal noder.

Orsaken till detta är att när det binära sökträdet byggs från sorterat data blir det kraftigt obalanserat. Då blir höjden ungefär  $n$  istället för  $\log n$ . Detta gör att sökningen blir  $O(n)$  istället för  $O(\log n)$ .

Nodes	Average ms	Quota
10000	30191	
20000	61509	2,037

Tabell 2: Data från andra studien.