

DV1: DATAVETENSKAPENS BYGGSTENAR

Analysmoment av OU3

Theodor Jonsson

2022-01-18

Inledning

Att analysera tidskomplexiteten för insättning i en hashtabell är en aning komplicerat. Detta beror på att tidskomplexiteten bestäms av hur nycklarna fördelar sig över hashtabellens buckets och de kan bli kollision alltså två nycklar blir hashade till samma index. I denna rapport ska vi undersöka tidskomplexitet för insättning i en hashtabell med inga kollisioner och insättning med möjliga kollisioner.

Metod

Vi skapar en hashtabell av varianten closed hashing med linear probing med en kapacitet av 1000000 buckets. Vi mäter den genomsnittliga tiden det tar att sätta in element i hashtabellen genom att sätta in element med randomiserade nycklar i en tom tabell tills vi har nått den efterfrågade fyllnadsgraden. Den totala tiden mäts och delas med antal insatta element till sist tömmer vi tabellen. Detta loopas om tills vi har nått fyllnadsgraden av 99,8%. Fyllnadsgraden startar på 0,2% och ökar efter varje loop med 0,2%

Resultat

Funktionen Insert behöver först hitta en bucket åt värdet vi försöker sätta in fast denna funktion har en loop så blir detta $O(1)$ för utan kollision kommer de alltid vara första iterationen av loopen som hittar rätt bucket på grund av att den ej behöver en offset. Resultatet i Diagram 1 visar tiden för den genomsnittliga insättning av ett element per fyllnadsgrad i en hashtabell med varierande kollisionsfrekvens

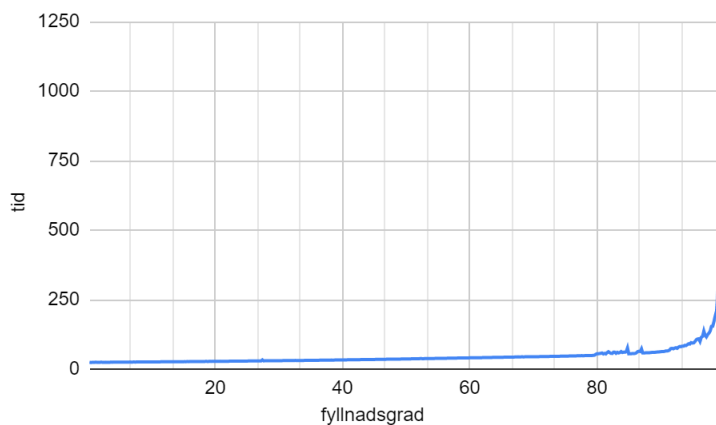


Diagram 1. Blåa linjen visar datapunkter för tid för insättning i ökande fyllnadsgrader tiden mäts i nanosekunder.

Diskussion

Det experimentella resultatet liknar den teoretiska resultatet tills de experimentella närmar sig max fyllnadsgrad då de experimentella tid per insättning ökar kraftigt. Denna hashtabell bör bara användas när fyllnadsgraden ej går över 90% då de blir för ineffektivt att sätta in element efter 90%.