Art Mehilane ja Raido Priske

Ülesanne 3: Separate Chaining

Separate Chaining

Ajalise komplekssuse parim Juhtum on O(1), kui kokkupõrkeid ei esine või need on minimaalsed.

Halvimas juhtumis, kui kõik elemendid satuvad samasse ämbrisse, võib see muutuda jõudluseks O(N).

Separate chaining saavutab hea ajalise jõudluse, kui koormustegur (load factor, elementide arv jagatud ämbrite arvuga) on madal ja hajusfunktsioon jaotab elemendid ühtlaselt.

Ruumiline komplekssus on suurem, eriti kui kasutatakse lingitud listid. Iga lisatud element nõuab lisamälu lingitud listi sõlme jaoks.

Open Addressing

Ajalise komplekssuse parim Juhtum on O(1), kui kokkupõrkeid ei esine.

Halvim Juhtum oleks O(1) kuni O(N), sõltuvalt koormustegurist ja kokkupõrgete lahendamise meetodist.

Jõudlus halveneb oluliselt, kui hajustabel on suures osas täis.

Open addressing ei sobi hästi suure koormusteguriga olukordades, kuna kokkupõrgete arv ja nende lahendamise keerukus suureneb.

Open addressing on **ruumiliselt** efektiivsem, eriti kui andmekogum on suhteliselt väike võrreldes hajustabeli suurusega, kuna see vähendab täiendava mälu vajadust.

Separate chaning meetodi kasutamine räsitabelites

POSITIIVNE

- ### Lihtne käsitsemine kokkupõrgetest

  Separate chaining lahendab kokkupõrkeid tõhusalt, sest mitu elementi võivad jagada sama hajustabeli indeksi ilma jõudluse olulise languseta.

- ### Stabiilsem jõudlus suurte andmekogumite puhul

  Jõudlus püsib suhteliselt stabiilne isegi siis, kui hajustabelisse lisatakse palju elemente.

- ### Lihtne implementeerida

  Separate chaining on suhteliselt lihtne implementeerida ja mõista, eriti lingitud listide kasutamisel.

NEGATIIVNE

- ### Ebaefektiivne ruumikasutus

  Iga separate chainingus kasutatav andmestruktuur (nt lingitud list) vajab täiendavat ruumi, mis võib olla ebaefektiivne.

- ### Mälu haldamine

  Lingitud listide kasutamine nõuab täiendavat mäluhaldust, mis võib olla keerulisem optimeerida.

- ### Jõudluse langus suure koormuse puhul

  Kui koormustegur on väga kõrge, võib jõudlus halveneda.