

## Opgaver onsdag den 26. februar

### Opgave 1

(I denne opgave skal du ikke programmere, men selv udregne hvordan hashtabellerne kommer til at se ud - anvend papir og blyant !)

Denne opgave handler om indsættelse af tal i en hashtabel ved forskellige kollisionsstrategier.

Følgende tal skal indsættes i en hashtabel i den angivne rækkefølge

4371 1323 6173 4199 4344 9679 1989

Hashtabellen har plads til 10 elementer og hash funktionen er givet ved  $h(x) = x \bmod 10$ . Hvorledes ser hashtabellen ud efter indsættelsen af ovenstående tal, når der anvendes

- 1) kædereglen
- 2) lineær probing
- 3) kvadratisk probing
- 4) dobbelt hashing, hvor den anden hashfunktion er givet ved  $h_2(x) = 7 - (x \bmod 7)$

### Opgave 2

I filen hashing.zip kan du finde en delvis implementation af klassen MyHashSetChaining samt en anvendelses klasse MyHashSetDemo. I klassen er der implementeret et forsimplet HashSet under anvendelse af kædereglen som kollisionsstrategi. Læs koden og afprøv klassen, så du er sikker på, at du forstår implementationen og kan forklare den.

Programmer metoderne remove og clear.

Lav en metode rehash() der kaldes fra add – metoden, når load faktoren bliver større 0.75.

### Opgave 3

Programmer klassen MyHashSetLinearProbing der er med i ovenstående zip.fil. Klassen skal have den samme funktionalitet som klassen MyHashSetChaining, men kollision skal håndteres ved linear probing. Bemærk attributten DELETED der kan bruges til at indikere at pladsen tidligere har været brugt.

(Du kan se bort fra rehashing)

Hvad er størrelsen af tidskompleksiteten af metoderne?