10. Universal and Perfect Hashing Intro
Definitions
Some universe containing all possible integer keys we shall map

$$U = \{0, 1, ..., u-1, u\}$$

Size of the bucket we hash into

$$m$$

A hash function in our finite hash fuction space

$$h_i \in H$$

Mængden af elementer vi får som input er

$$n$$

Univiseral Hashing Vi vil gerne minimere chancen for collision, det optimale ville være at have

$$\forall_{i,j \in U, i \neq j} : pr(h(i) = h(j)) = \frac{1}{m}$$

Problemet er, at totalt random er næsten umuligt at opnå, det er irrationelt at tro at alt input er random og hvis vi gør, laver vi en avarege case analysis. Vi bliver nødt til at introducere noget randomness.

Måden vi introducerer randomness, er ved at vælge en tilfældig

$$h_i \in H$$

sådan at

$$\forall_{h_i \in H} : pr(h(k) = h(l)) \leq \frac{1}{m}, k, l \in U, k \neq l$$

Såfremt vi kan opnå denne collision rate, så kan vi udregne det forventede antal collision givet et input længde på

$$n$$

keys. Lad os bruge en indicator random variable

$$X_{i,j}$$

til at denotere hvorvidt en en key

$$i$$

kolliderer med

$$j$$

. Vi kan da tælle antallet af kollisioner som

$$X_{i,j} = 1, h(i) = h(j)$$
$$X_{i,j} = 0, h(i) \neq h(j)$$

$$X = \sum_{j \in U, j \neq i} X_{i,j}$$

$$E[X] = \sum_{j \in U, j \neq i} E[X_{i,j}] = \sum_{j \in U, j \neq i} pr(X_{i,j} = 0) \cdot 0 + pr(X_{i,j} = 1) \cdot 1) = \sum_{j \in U, j \neq i} \frac{1}{m} = \frac{n}{m}$$

Så fremt at en key

$$j$$

allerede er i vores hashing skal vi tilføje en enkelt sådan at det er

$$1 + \frac{n}{m}$$

da det garrantere en collision.

Vi kan konstruere en sådan random hashfunction som følger. Antag at vi vælger et primtal

$$p > |U|$$

Da kan vi vælge to værdier

$$a, b < p$$

sådan at vi kan konstruerer hashfunktioner der kan mappe

$$k$$

til en bucket som

$$h(k)_{a,b} = (a \cdot k + b \mod p) \mod m, a \neq 0$$

Vi kan da se at vi har

$$p(p-1)$$

forskellige hashfunktioner at vælge fra.

Perfect Hashing Her er vi ikke interesseret i at kunne indsætte og slette, men arbejder med statisk data og vil gerne kunne søge i worst case konstant tid i stedet for expected. Hvor stort skal vi gøre m for at vi ikke har nogle collisions? Lad os prøve at sætte den til

$$m = n^2$$

i forhold til før. Vi kan tælle antallet af potentielle kollisioner som

$$\binom{n}{2}$$

. Ved brug af universal hashing har vi følgende

$$E[X] = \sum_{i,j} E[X_{i,j}] = \sum_{i,j} \frac{1}{m}] = \frac{\binom{n}{2}}{m} \leq \frac{1}{2}$$

Med markovs inequality får vi at

$$pr(X \geq 1) = \frac{E[X]}{1} < \frac{1}{2}$$

Så vi kan bare fortsætte med at prøve til vi har en god hashing, dette er også kendt fra the birthday paradox. Problemet er dog at vi kommer til at bruge polynomiel plads, så det vi kan lave er et lag mere hvor vi bruger samme logik i hver af disse for at få polynomiel pladsbrug.