Hashing perfecto

Dado un S conocido y fijo, puedo construir un h que no genere colisiones en S ??

Puedo hacer algo tipo las Vegas, ir probando $h \in H$ hasta encontrar una función que no produzca colisiones. Pero qué pasa si H es muy grande?

Elijo $h \in H$ 2-universal

$$Pr(h(x) = h(y)) \leq \frac{1}{N}$$

prob. de or es $\leq$ que la suma de todos

$$Pr(\exists x \neq y,\ h(x) = h(y)) \leq \sum_{x \neq y \in S} Pr(h(x) = h(y))$$

$$= \frac{|S|(|S|-1)}{2} \cdot \frac{1}{N}$$

cota para escoger h no perfecta

$\therefore$ si elijo $N = |S|^2$ (tabla grande)

entonces $Pr(h \text{ no es perfecta}) < \frac{1}{2}$

Luego es poco probable que un algoritmo Las Vegas demore mucho en probar y encontrar h que no produzca colisiones.

(El trabajo de probar cada h toma $\Theta(|S|)$, e inicializar puede hacerse en $\Theta(1)$, si tabla es grande y uso pocos eltos)

$\Rightarrow$ es un alg. Las Vegas con tiempo esperado $\Theta(|S|)$

número de intentos "cte" → probable?

Pero $N = |S|^2$ es inaceptable

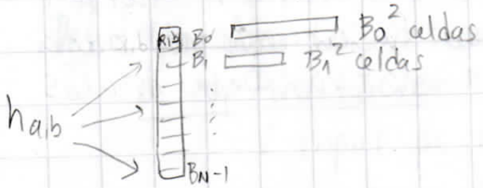
cuadrático

Queremos tiempo de construcción lineal y espacio lineal $\mathcal{O}(|S|)$

Probamos una función distribuidora.

$$h : S \to [0 .. N-1] \qquad N = |S|$$

Contamos cuántos elementos pone en cada celda, B_i



a cada celda le construyo otra tabla de tamaño cuadrático para buscar rápidamente ahí con alta probabilidad de no colisión. Los B_i^2 no son muy grandes pues ya está distribuidos

Elijo una $h \neq$ para cada celda.

Cuánto es el valor esperado de $\sum_{i=0}^{N-1} B_i^2 = \sum_{x,y} C_{xy}$

$\boxed{1} \to x, y, z \qquad B^2 = 9 \qquad x, y$ y z colisionan

$$\begin{array}{lll} C_{xx} = 1 & C_{yx} = 1 & C_{zx} = 1 \\ C_{xy} = 1 & C_{yy} = 1 & C_{zy} = 1 \\ C_{xz} = 1 & C_{yz} = 1 & C_{zz} = 1 \end{array}$$

$$\sum_{i=0}^{N-1} B_i^2 = \sum_{x,y} C_{xy} = N + \sum_{x \neq y} C_{xy}$$

$$E\left(\sum_{i=0}^{N-1} B_i^2\right) = N + \sum_{x \neq y} E(C_{xy})$$

cantidad de pares (x, y), $x \neq y$

$$\leq N + \underbrace{N(N-1)}\frac{1}{N}$$

$$< 2N$$