



Cerca per localització

Es pot pensar en formes de cerca que no requeresquen **comparacions** entre claus (igual que en el cas de l'ordenació).

El cas més senzill és quan el tipus de dades de les claus permet indexar un vector.

clau	informació

D'aquesta manera es poden insertar, eliminar i consultar elements per claus en temps **constant**. L'únic problema és l'espai.



Funció de dispersió

Si el tipus de dades de les claus, δ , no permet indexar un vector, sempre es pot pensar a canviar les claus de tipus:

$$f : \delta \longrightarrow 0..M - 1$$

Si $|\delta| = M$ i f és bijectiva tornem a tindre el cas més senzill.

Cas interessant en la pràctica: $|\delta| \gg M$

(M és el nombre de posicions de memòria que podem dedicar donada una aplicació particular).

Exemple: δ = enters en un rang molt gran

Exemple: δ = cadenes de símbols



Cerca per dispersió

Hi ha dues components principals:

- una funció de dispersió que done lloc a una ocupació homogènia de la taula.
- una estratègia de tractament de col·lisions

col·lisió: Dues claus x i y compleixen que $f(x) = f(y)$ i les correspon una mateixa posició de la taula.



Funcions de dispersió

Si $\delta = 0..N - 1$ sempre podem fer $f(k, M) = k \bmod M$

Açò mateix es pot fer sempre que es tinga accés a la representació en bits de qualsevol tipus de clau.

És convenient escollir M com a nombre primer si es fa servir una funció de dispersió modular.

La dispersió modular distribueix homogèniament les claus sempre que aquestes estiguen uniformement distribuïdes (en δ).

Per a claus de tipus real (entre 0 i 1): $f(k, M) = (k \cdot 2^w) \bmod M$



Funcions de dispersió

Una modificació de la dispersió modular: $f(k, M) = \lfloor k\alpha \rfloor \bmod M$

α és un valor arbitrari (p.e. $\alpha = 0.618033$)

Per a claus alfanumèriques (taules de símbols):

$$f(k, M) = (\sum_{i=1}^c k_i \cdot B^i) \bmod M$$

on $0 \leq k_i < B$ és el valor de l' i -èssim símbol (la clau és com un nombre en base B).

En ASCII, $B = 128$ (7 bits) però és molt millor fer $B = 127$ que és primer.



Funcions de dispersió

Per fer-ho eficient cal fer servir la regla de Horner (amb aritmètica modular per tal de evitar desbordaments)

```
int hash(char *v, int M)
{ int h = 0, b = 127;
  for (;*v != '\0'; v++)
    h = (b*h + *v) % M;
  return h;
}
```

Dispersió universal (per a claus alfanumèriques): el mateix però fent servir un valor de B aleatori per a cada símbol.



Funcions de dispersió

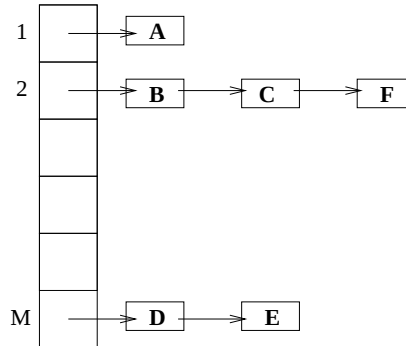
En general, la funció de dispersió,

- ha de ser molt ràpida de calcular
- ha de tenir en compte la distribució original (esperada) de les claus.
- s'ha de calcular a partir de *tota* la clau.
- ha d'assegurar en la mesura del possible la distribució homogènia de les claus en la taula.



Dispersió oberta (Encadenament separat)

En realitat es té una taula de M llistes lligades. Cada llista lligada contindrà tots els elements als que els correspon la mateixa entrada.



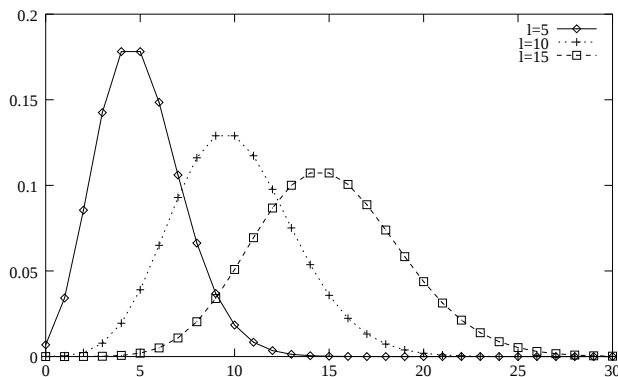
Si la funció de dispersió és correcta, la grandària de totes les llistes serà molt pròxima a N/M



Dispersió oberta (Encadenament separat)

Probabilitat que una llista tinga k items:
$$\binom{N}{k} \left(\frac{1}{M}\right)^k \left(1 - \frac{1}{M}\right)^{N-k}$$

Si $\lambda = N/M$,
$$\binom{N}{k} \left(\frac{\lambda}{N}\right)^k \left(1 - \frac{\lambda}{N}\right)^{N-k} \leq \frac{\lambda^k e^{-\lambda}}{k!}$$





Dispersió oberta (Encadenament separat)

No només es compleix que la longitud mitjana de les llistes és N/M si no que a més a més és relativament improbable que alguna llista difereixca molt d'aquest valor.

Els costs promig són aleshores representatius (normalment $\lambda = 5, 10$).

Cerques sense èxit: $1 + \lambda$

Cerques amb èxit: $1 + \frac{\lambda}{2}$

Les llistes són una opció. Es poden utilitzar llistes ordenades, arbres binaris o qualsevol altra estructura.



Dispersió tancada (direccionament obert)

Per què perdre memòria emmagatzemant punters?

Per què no fer servir posicions lliures en la taula en cas de colisió?

A favor: menys memòria. no es manipulen punters.

En contra: augmenta la probabilitat de colisió? Cal elegir la grandària?

La dispersió tancada es pot veure com l'oberta però on les llistes estan “integrades” dins de la taula.



Dispersió tancada (direccionament obert)

La relació entre el nombre d'elements emmagatzemats i el nombre total de posicions rep el nom de **factor de càrrega**, $\lambda = N/M$.

En els dos tipus de dispersió té connotacions diferents:

dispersió oberta: és normalment major que 1. Es correspon amb la longitud mitjana de les llistes.

dispersió tancada: és necessàriament menor que 1 (i no convé que s'apropi!). Mesura el nivell d'ocupació.

Normalment, $\lambda \leq 0.5$.



Dispersió tancada. Sondejos

En general, l'inserció d'un element en la taula requerirà una sèrie de proves (sondejos, exploracions, *probes* en anglès).

$f_0, f_1, f_2, \dots$

on $f_i : \delta \longrightarrow 0..M - 1$ i $f_0 = f$ la funció de dispersió "normal".

La idea consisteix en anar calculant successives f_i mentre ocorresquen col·lisions.

Un cas molt general queda representat per:

$$f_i(x) = [f(x) + g(i)] \bmod M$$



Exploració lineal (*linear probing*)

La idea és recórrer la taula (de manera ascendent o descendent) fins a trobar una posició lliure.

Es correspon amb $g(i) = i$ (o $g(i) = -i$).

La recuperació de l'element requereix una cerca lineal basada en la comparació de la clau fins trobar-la (èxit) o arribar a una posició buida (fracàs).

La taula resultant de l'inserció d'un conjunt d'elements depén (fortament) de l'ordre.



Exploració lineal. Inserció

0	0	⇐
1	1	⇐
2		
3		
4		
5		
6		
7		
8		
9		

0	0	↘
1	1	↘
2	81	←
3		
4	4	⇐
5	25	⇐
6	16	⇐
7		
8		
9		

0	0	
1	1	
2	81	
3		
4	4	↘
5	25	↘
6	16	↘
7	64	←
8		
9		

0	0	
1	1	
2	81	
3		
4	4	
5	25	
6	16	↘
7	64	↘
8	36	←
9	9	⇐

0	0	↘
1	1	↘
2	81	↘
3	49	←
4	4	
5	25	
6	16	
7	64	
8	36	
9	9	↘

Inserció de les claus: 0, 1, 81, 4, 25, 16, 64, 36, 9, 49.



Estratègia lineal. Inserció

Si cada intent fóra independent, i com que la probabilitat de trobar una posició buida és $1 - \lambda$, el nombre esperat d'operacions per cada inserció seria $\frac{1}{1-\lambda}$.

Açò representa una fita optimista ja que, en realitat, els intents no són independents degut al fet dels agrupaments de claus.

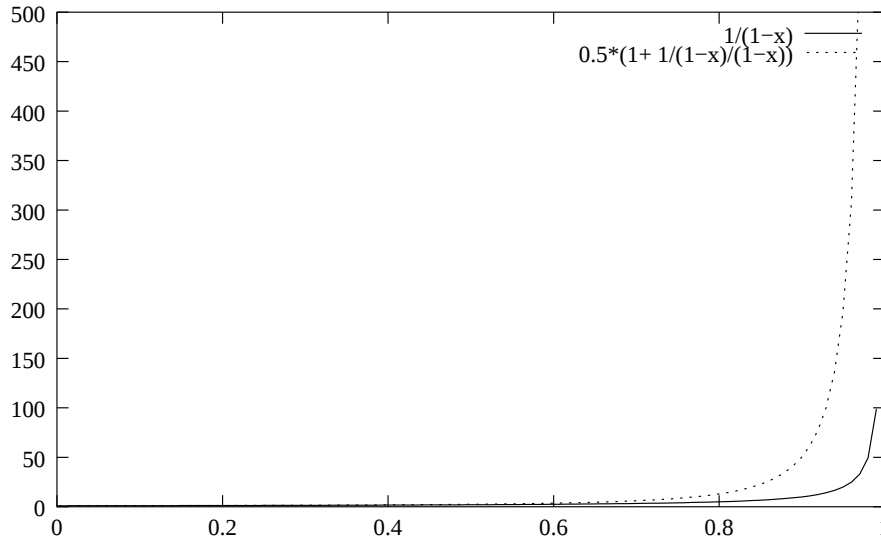
El nombre esperat d'operacions per cada inserció és en realitat:

$$\frac{1}{2} \left(1 + \frac{1}{(1-\lambda)^2} \right)$$



Estratègia lineal. Inserció

Comparació entre el cost de una inserció i la seua fita optimista en funció del factor de càrrega.





Estratègia lineal. Cerca

A l'hora de cercar elements, l'anàlisi és exactament la mateixa en el cas de cerques sense èxit: cal recórrer les cadenes de claus que han col·lisionat (si escau) fins trobar una posició buida.

En el cas de cerques amb èxit, l'anàlisi és diferent.

Observació: el cost de trobar un element X en la taula és exactament igual al cost de la seua inserció!



Estratègia lineal. Cerca

Aleshores si $I(\lambda)$ és el cost d'una inserció, donat λ , el cost d'una cerca exitosa és

$$\frac{1}{\lambda} \int_{\lambda'=0}^{\lambda} I(\lambda') d\lambda' = \frac{1}{2} \left(1 + \frac{1}{(1-\lambda)} \right)$$

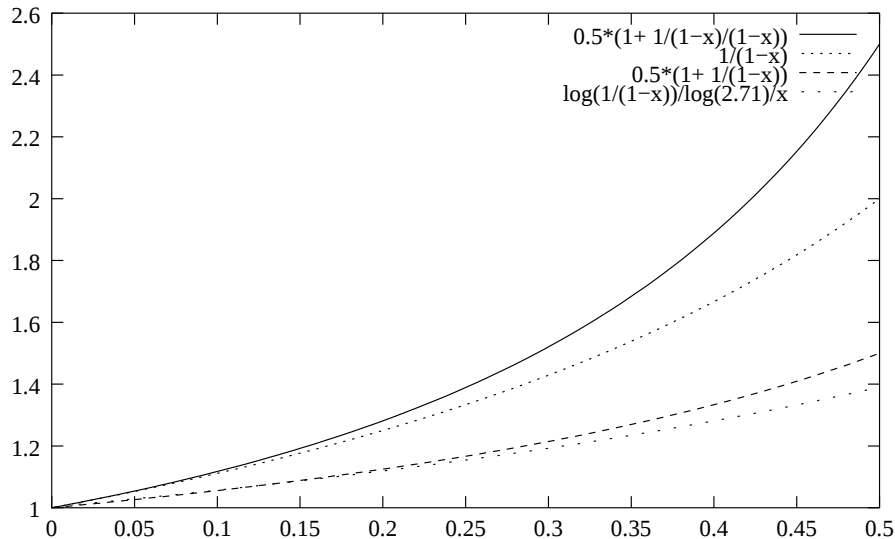
Exactament el mateix càlcul es pot fer suposant que els intents són independents (és a dir, si els agrupaments de claus en la taula fóren independents). Només cal substituir el cost de la inserció en la integral anterior.

$$\frac{1}{\lambda} \ln \frac{1}{1-\lambda}$$



Estratègia lineal. Cerca

Els costos de cerques amb èxit i sense èxit amb les corresponents fites optimites en el cas d'independència:





Estratègia quadràtica

Una forma d'evitar els agrupaments i, sobretot, la dependència entre unes claus i altres consisteix en recórrer posicions no contigües.

Una forma de fer-ho és amb $g(i) = i^2$ (per exemple). Aleshores,

$$f_i(x) = [f_o(x) + i^2] \bmod M$$

La forma eficient de calcular $f_i(x)$ fa servir el valor $f_{i-1}(x)$ ja calculat. En particular,

$$f_i(x) = [f_{i-1}(x) + 2i - 1] \bmod M$$



Estratègia quadràtica

0	0	⇐
1	1	⇐
2		
3		
4		
5		
6		
7		
8		
9		

0	0	↘
1	1	↘
2	81	←
3		
4	4	⇐
5	25	⇐
6	16	⇐
7		
8		
9		

0	0	
1	1	
2	81	
3		
4	4	↘
5	25	↘
6	16	
7		
8	64	←
9		

0	0	
1	1	
2	81	
3		
4	4	
5	25	
6	16	↘
7	36	←
8	64	
9	9	⇐

0	0	↘
1	1	
2	81	
3	49	⇐
4	4	
5	25	
6	16	
7	36	
8	64	
9	9	↘

Inserció de les claus: 0, 1, 81, 4, 25, 16, 64, 36, 9, 49.

Si l'última clau insertada haguera sigut 11 (per exemple) no s'haguera pogut insertar!!



Estratègia quadràtica

La inserció mitjançant estratègia quadràtica només es pot garantir si M és primer i si $\lambda \leq 0.5$.

Sigueu $\{[f_0(x) + i^2] \bmod M\}_{i=0}^{\lfloor M/2 \rfloor}$ les $\lfloor M/2 \rfloor$ primers exploracions per tal d'insertar un element.

Siga $0 \leq i \leq j \leq \lfloor M/2 \rfloor$ i les exploracions i -èssima i j -èssima:

$$[f_0(x) + i^2] \bmod M \text{ i } [f_0(x) + j^2] \bmod M.$$

SUPOSEM que foren iguals. Aleshores (en aritmètica mòdul M),

$$f_0(x) + i^2 = f_0(x) + j^2 \Rightarrow i^2 - j^2 = 0 \Rightarrow (i + j)(i - j) = 0$$

Aquesta última igualtat contradiu el fet que M siga primer!



Agrupaments primari i secundari

- Amb l'estratègia lineal, les claus que col·lisionen se situen molt prop unes d'altres (formen "tireres"). A més a més, unes i altres poden mesclar-se. Açò es coneix com **agrupament primari**.
- Tant amb l'estratègia lineal com amb la quadràtica, les seqüències exploracions de posicions son sempre les mateixes per a totes les claus que col·lisionen.

Les "tireres" segueixen existint encara que en forma inconnexa dins la taula. Açò es coneix com **agrupament secundari**.

- Seria desitjable eliminar tots dos tipus d'agrupaments per tal que les diferents probabilitats de col·lisió foren el més independents possible.



Dispersió doble

Consisteix en la utilització d'una segona funció de dispersió per tal de cercar les següents posicions en cas de col·lisió. Aquesta segona funció ens dona un increment fix per recórrer les posicions de la taula.

Ara, $f(x) = [f_0(x) + g(i, x)] \bmod M$ on $g(i, x) = i \cdot h(x)$.

La segona funció de dispersió, h , ha de complir que $h(x) \neq 0$. A més a més, els valors que done, han de garantir que no es puguen produir bucles infinits!

En la pràctica, si M és primer, es pot prendre $h(x) = R - (x \bmod R)$ on R és un nombre primer menor que M .

Una altra elecció és $h(x) = 1 + (x \bmod R)$



Dispersió doble

0	0 ←	0	0	0	0	0	0 ↘	0	0
1	1 ←	1	1 ↘	1	1	1	1	1	1
2		2		2		2	64 ←	2	64
3		3		3		3		3	
4		4	81 ←	4	81 ↘	4	81 ↘	4	81
5		5		5		5	25 ←	5	25
6		6		6	4 ←	6	4 ↘	6	4 ↘
7		7		7		7		7	
8		8		8		8	16 ←	8	16
9		9		9		9		9	36 ←

Inserció de les claus: 0, 1, 81, 4, 25, 16, 64, 36, (9, 49).

$$h(x) = 3 - (x \bmod 3)$$

$$h(81) = h(36) = h(9) = 3,$$

$$h(4) = h(16) = h(64) = h(49) = 2$$



Dispersió doble

Es pot demostrar que la dispersió doble tendeix als resultats de l'anomenada dispersió aleatòria (les fites optimistes d'abans).

Tot i això, i suposant que es treballa amb factors de càrrega $\lambda \leq 0.5$ estem parlant d'una diferència de mitja comparació menys!

El que si que es redueix sensiblement és la probabilitat de poder trobar alguna seqüència d'exploracions molt llarga (per a factors de càrrega petits).



Dispersió dinàmica. Redispersió

Quan no es puga predir el nombre d'elements que caldrà emmagatzemar, cal contemplar la possibilitat que la taula puga fer-se més gran.

Per tal que no es reduezca l'eficiència, es fa servir la tècnica dels vectors extensibles (es duplica la talla).

- La nova talla ha de ser un nombre primer.
- La redimensió s'ha de produir en el moment en què $\lambda > 0.5$.
- En alguns tipus de dispersió (quadràtica, doble) pot ocòrrer que una inserció falle. Aleshores cal redimensionar.
- Cal tornar a introduir tots els elements en la taula nova!



Dispersió dinàmica. Redispersió

0	0	↘
1	1	↘
2	81	←
3		
4	4	←
5	25	←
6	16	←
7		
8		
9		

0	0	←
1	1	←
2	16	←
3		
4	4	←
5		
6		
7		
8		
9		
10		
11	81	←↘
12	25	←
13		

0	0	
1	1	
2	16	
3		
4	4	
5		
6		
7	49	←
8	64	←↘
9	36	←↘
10	9	←
11	81	←↘
12	25	←
13		

Inserció de les claus: 0, 1, 81, 4, 25, 16. Redispersió.

Inserció de les claus: 64, 36, 9, 49.



Eliminació en taules de dispersió

dispersió oberta: No hi ha cap problema.

estratègia lineal: S'esborra l'element i es continua recorreguent la “tira” de claus que col·lisionen i es recoliquen.

dispersió tancada (en general): Es marca l'element com a esborrat però continua fent-se servir per trobar els demás (eliminació pereosa).