

Elección de estructuras (Parte 2)

Los iteradores contraatacan

Algoritmos y Estructuras de Datos 2

Departamento de Computación,
Facultad de Ciencias Exactas y Naturales,
Universidad de Buenos Aires

13 de Octubre de 2017

¿Se acuerdan?

Nos encargaron implementar un PADRÓN que mantiene una base de datos de personas, con DNI, nombre, fecha de nacimiento y un código de identificación alfanumérico.

- ▶ El DNI es un entero (y es único).
- ▶ El nombre es un string.
- ▶ El código de identificación es un string (y es único).
- ▶ La fecha de nacimiento es un día de 1 a 365 (sin bisiestos) y un año.
- ▶ Sabemos además la fecha actual y por lo tanto la edad de cada persona (la cual sabemos que nunca supera los 200 años).

Además de poder agregar y eliminar personas del PADRÓN se desea poder realizar otras consultas en forma eficiente.

Especificación

TAD PADRON

observadores básicos

fechaActual	: padron	→	fecha	
DNI	: padron	→	conj(DNI)	
nombre	: DNI $d \times$ padron p	→	nombre	$\{d \in \text{DNIs}(p)\}$
edad	: DNI $d \times$ padron p	→	nat	$\{d \in \text{DNIs}(p)\}$
código	: DNI $d \times$ padron p	→	código	$\{d \in \text{DNIs}(p)\}$

generadores

crear	: fecha <i>hoy</i>	→	padron	
avanzDia	: padron p	→	padron	$\{\text{sePuedeAvanzar}(p)\}$
agregar	: persona $t \times$ padron p	→	padron	$\left\{ \begin{array}{l} \text{dni}(t) \notin \text{DNIs}(p) \wedge \text{código}(t) \notin \text{códigos}(p) \wedge \\ \text{nacimiento}(t) \leq \text{fechaActual}(p) \end{array} \right\}$
borrar	: DNI $d \times$ padron p	→	padron	$\{d \in \text{DNIs}(p)\}$

otras operaciones

códigos	: padron	→	conj(código)	
persona	: código $c \times$ padron p	→	persona	$\{c \in \text{códigos}(p)\}$
tienenAños	: nat \times padron	→	nat	
jubilados	: padron	→	nat	

Fin TAD

Los requerimientos de complejidad temporal

Nos piden que respetemos las siguientes complejidades:

1. Agregar una persona nueva en $O(\ell + \log n)$.
2. Dado un código / DNI, encontrar los datos de la persona en $O(\ell) / O(\log n)$.
3. Dada una edad, decir cuántos tienen esa edad en $O(1)$.
4. Decir cuántas personas están en edad jubilatoria en $O(1)$.
5. Avanzar el día actual en $O(m)$.

donde:

- ▶ n es la cantidad de personas en el sistema.
- ▶ ℓ es la longitud del código recibido como parámetro.

Donde estábamos ...

padrón **se representa con** *estr*, donde

estr **es** tupla \langle *porCódigo*: diccTrie(string, persona)
porDNI: diccAVL(nat, persona)
cantPorEdad: arreglo_dimensionable(nat)
cumplenEn: arreglo_dimensionable(conjAVL(persona))
día: nat
año: nat
jubilados: nat \rangle

y *persona* **es** tupla \langle *nombre*: string, *código*: string,
dni: nat, *día*: nat, *año*: nat \rangle

Extras

- ▶ Nos piden poder modificar la información de una persona en $O(\log n)$ accediendo a través del DNI y al mismo tiempo disminuir al mínimo la redundancia de la información de las personas.
- ▶ Dada una cadena ℓ , cantidad de personas cuyo nombre empieza con ella $O(|\ell|)$.
- ▶ Dado un DNI, cuantas personas hay con DNI mayor o igual $O(\log n)$.
- ▶ Iterar las cantidades de años en que al menos una persona tiene esa edad $O(x)$

x : cantidad de años tales que alguna persona tiene esos años

Extras 2

- ▶ Dada una edad, saber qué DNIs tienen esa edad en $O(1)$.

Tener en cuenta las complejidades anteriores. En particular:

- ▶ Agregar una persona nueva en $O(\ell + \log n)$.
- ▶ Avanzar el día actual en $O(m)$.

siendo m es la cantidad de personas que cumplen años el día al que se llegan.

Extras 3

Se nos pide relajar la restricción que plantea que las personas cumplen a lo sumo M años. A cambio, se modifican los siguientes requerimientos:

- ▶ Agregar una persona debe tener costo $\mathcal{O}(\ell + \log n + x)$
- ▶ En lugar de obtener los DNI para una edad en particular, se pide tener un iterador de tuplas (edad, conj(DNI)) que tenga como costo para iterar todas las edades con algún dni en $\mathcal{O}(x)$.
- ▶ No se pide más conocer la cantidad de dnis para una edad en particular ya que se obtiene del iterador anterior.

con x cantidad de años tales que alguna persona tiene esos años y m la cantidad de personas que cumplen años el día al que se llegan.