

## PAC 2

### Exercici 1 [2 punts]

Tenim una cua prioritària representada en un arbre parcialment ordenat d'aritat 4 (és a dir, cada node te un màxim de 4 fills). Indiqueu quants nivells són necessaris per a emmagatzemar 1.000 elements. Digueu quantes comparacions entre elements seran necessàries en el pitjor dels casos quan es vulgui inserir un nou element en aquesta cua de 1.000 elements. Indiqueu quantes comparacions entre elements seran necessàries en el pitjor cas quan esborrem el menor dels elements (el més prioritari) d'aquesta cua prioritària.

El nombre de nodes  $x$  que caben en un arbre de aritat 4 complet de  $h$  nivells és

$$1 + 4 + \dots + 4^{h-1} = (4^h - 1) / (4 - 1)$$

Per tant resolent:

$$\frac{4^h - 1}{4 - 1} = X; h = \log_4(3x + 1)$$

i com que el nombre de nivells no es pot fraccionar, si el resultat és un nombre real, cal arrodonir al sencer immediatament superior. D'aquesta manera, el nombre de nivells necessaris per a emmagatzemar 1.000 elements és  $\log_4(3.001) = 5,77 \rightarrow 6$  nivells.

En el pitjor cas (inserir l'element més prioritari en un arbre complet, generant un nivell més) haurem de reestructurar tot l'arbre, comparant el node inserit (com una fulla en un altre nivell) amb el seu pare fins arribar a l'arrel, de manera que es realitzaran tantes comparacions com nivells hi hagi. Aquest és el cas genèric, però en el nostre arbre, que emmagatzema 1000 elements en 6 nivells, queda lloc per a un altre node dins del nivell 6, de manera que el nombre de comparacions necessàries és 5 (no cal "obrir" un altre nivell, i el nombre de comparacions és el nombre de nivells-1).

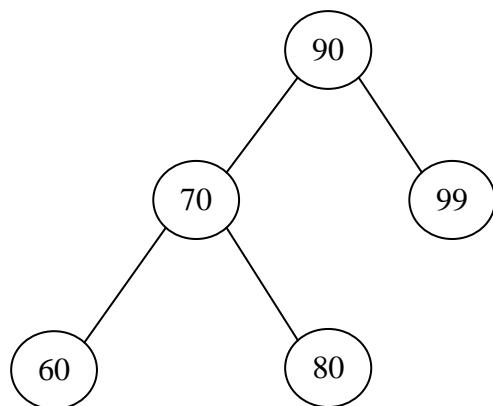
Quan eliminem el menor element (el més prioritari), cal eliminar l'element que es troba a l'arrel, el que obliga a reestructurar l'arbre perquè mantingui la relació d'ordre parcial. A cada reestructuració s'ha de comparar el node amb tots els seus fills per intercanviar amb el menor, i repetir aquest procés, en el pitjor cas fins arribar a les fulles, cal comparar

$$4 * (h - 1) = 4 * 5 = 20 \text{ cops.}$$

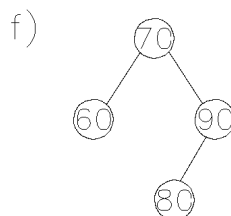
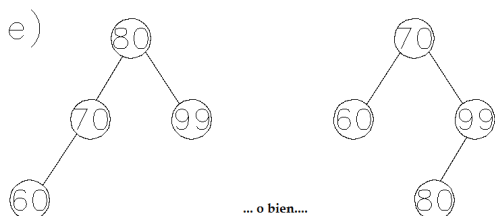
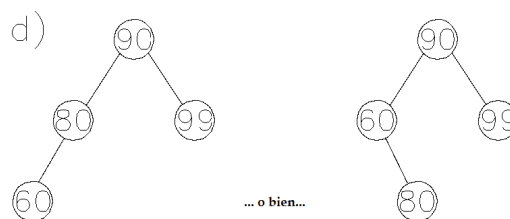
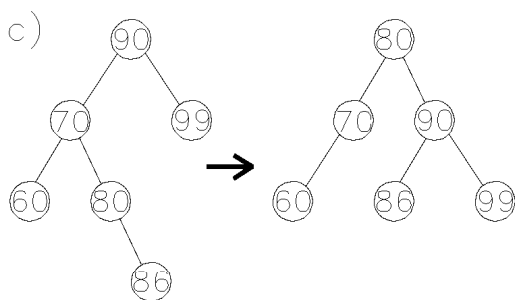
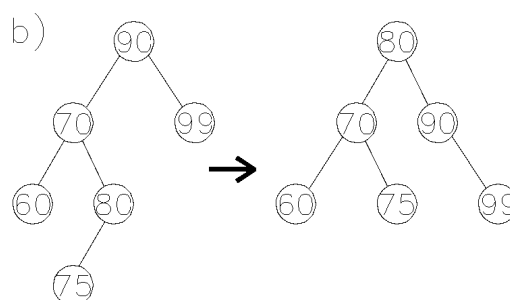
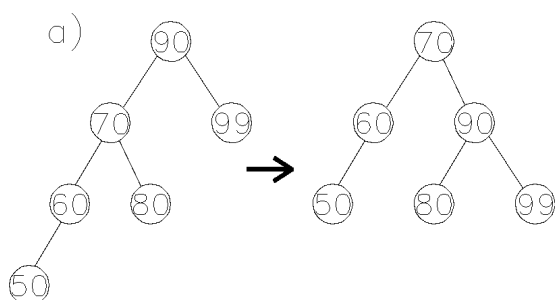


## Exercici 2 [2 punts]

Realitzeu les següents operacions, segons com heu vist a l'assignatura, sobre l'arbre AVL de la figura. El resultat de cada operació ha de seguir sent un arbre de cerca AVL. Realitzeu totes les operacions partint de l'AVL original, que es mostra a continuació:



- Afegir el 50
- Afegir el 75
- Afegir el 86
- Eliminar el 70
- Eliminar el 90
- Eliminar el 99



### Exercici 3 [2 punts]

Suposeu que tenim creat un directori d'enllaços a llocs interessants d'Internet, i volem afegir un diccionari dels termes o paraules clau més utilitzats en el títol i en la descripció d'aquests llocs.

El diccionari a crear traduirà cada terme en una llista encadenada de llocs que tenen aquest terme com a paraula clau. El nombre de termes a guardar és molt gran i anirà creixent constantment. A més, és requisit que es pugui recórrer alfabèticament la llista de termes.

Es demana el següent:

**Apartat 3.1)** Quina estructura de dades de les estudiades és la més adequada per implementar aquest diccionari? Raoneu la resposta.

Com que no es coneix el nombre de termes que contindrà el diccionari i es pot demanar un llistat ordenat d'aquests, a més, tenint en compte que la cerca en el mateix es pressuposa que ha de ser el més ràpida possible, es podria utilitzar un arbre AVL ordenat alfabèticament on la clau és el terme.

**Apartat 3.2)** Quina modificació es podria fer a l'estructura que permetés construir un iterador dels termes que comencen per una lletra determinada? Explica breument de quina manera s'implementaria aquest iterador.

Es podrien usar les lletres de l'alfabet com a termes emmagatzemats a l'AVL (amb valor nul). Aquests termes s'inseririen tots de cop en construir l'arbre. L'iterador a implementar recorreria l'arbre AVL en inordre, però a partir del terme corresponent a la lletra. L'iterador acabaria el seu recorregut en trobar el terme corresponent a la següent lletra.



## Exercici 4 [4 punts]

El Ministeri d'Educació publica diverses convocatòries l'any per finançar activitats de recerca. Les activitats poden ser de diversa índole: accions complementàries, xarxes d'especialització, projectes de R + D, etc. Un investigador pot participar simultàniament en diverses d'aquestes activitats, però només en un únic projecte d'R + D (podria participar en 3 xarxes i 1 projecte, per exemple). Els investigadors realitzen la seva activitat en el marc d'una universitat espanyola, i com a fruit de la seva recerca en l'activitat finançada realitzen publicacions científiques.

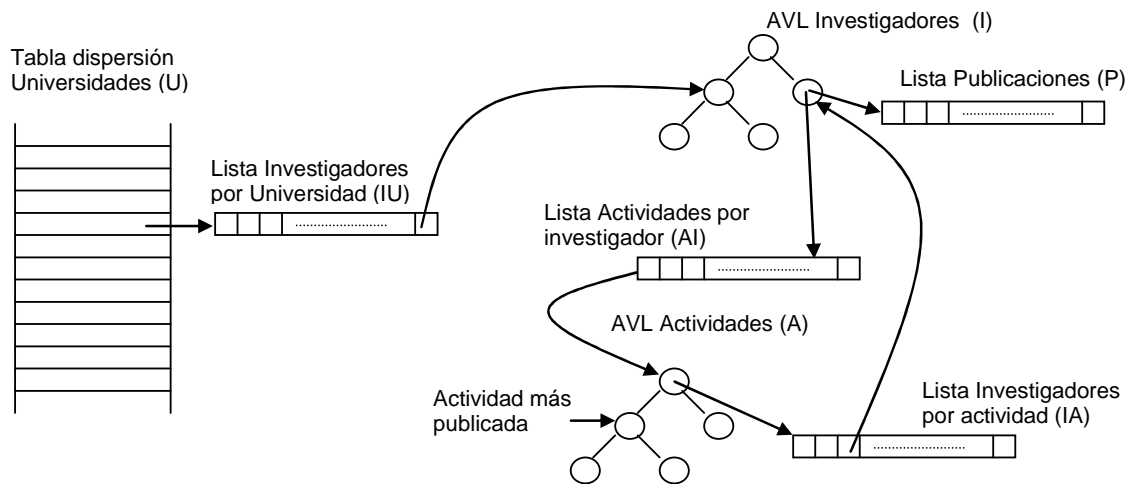
El nombre d'universitats és conegut i estable (diguem-li U). El nombre d'activitats que finança el Ministeri (diguem-li A) és gran i no es coneix a priori. El nombre d'activitats simultànies en què participa un investigador (diguem-li AI) tampoc es coneix, encara que s'estima que serà petit. El nombre de publicacions que un investigador pot realitzar en tots els projectes en què està actiu (diguem-li P) és petit (de l'ordre de les desenes), però tampoc es pot predir amb exactitud. El nombre d'investigadors que participen en una activitat (diguem-li IA) no és fix, però si es pot afirmar que no és molt gran, mentre que el nombre d'investigadors (els denominarem I) de les universitats espanyoles sí que és gran i variable. Cada universitat té un nombre desconegut d'investigadors (diguem IU) que depèn de la mida d'aquesta.

Es necessita dissenyar un TAD que reflecteixi aquesta situació i permeti realitzar les següents operacions:

- **finançarActivitat**(*llista Investigadors, llista universitats, activitat*). Insereix una nova activitat dins de les finançades pel Ministeri. En l'activitat participen els investigadors que es passen com a paràmetre. Si algun investigador no existeix a l'estructura, es donarà d'alta. La universitat a la qual pertany cada investigador figura a la llista que es passa com a segon paràmetre. Si l'activitat és un projecte de R + D, s'ha de comprovar que no hi ha entre els investigadors cap que participi en un altre projecte d'R + D en vigor. Si fos així, se l'exclouria de la llista d'investigadors del projecte. Pot suposar que les universitats dels investigadors ja existeixen en l'estructura.
- **consultarInvestigadorsUniversitat**(*universitat*): llista Investigadors. Retorna els investigadors d'una universitat que estan immersos actualment en alguna activitat de recerca finançada. Es pot suposar que la Universitat sempre està a l'estructura.
- **consultarInvestigadorsActivitat**(*activitat*): llista Investigadors. Retorna els investigadors que treballen en l'activitat que es passa com a paràmetre. Podeu suposar que l'activitat sempre existeix a l'estructura.
- **Publicar**(*publicació, investigador, activitat*). Dóna d'alta una publicació associada a una activitat feta per un investigador. Es pot suposar que tant l'investigador com l'activitat sempre són a l'estructura.
- **MesPublicada** (): activitat. Torna l'activitat que més vegades ha estat publicada.

Es demana dissenyar l'estructura de dades que suporti aquesta situació i permeti realitzar les operacions el més eficientment possible pel que fa al temps i sense malgastar espai. Feu un dibuix que expliqui gràficament l'estructura i indiqui l'eficiència de cada operació especificant les operacions internes que s'han de realitzar per dur-la a terme, juntament amb la seva eficiència (per exemple: inserir en AVL, buscar en llista enllaçada, etc).





### **finançarActivitat (llista Investigadors, llista universitats, activitat) $O(\log A + IA \cdot \log I)$**

Inserir a l'AVL l'activitat:  $O(\log A)$

Si la activitat al projecte de R+D: per a cada investigador de l'activitat  $O(IA)$

    Buscar a l'AVL investigadors:  $O(\log I)$ .

    Si no existeix,

        Inserir a l'AVL:  $O(\log I)$

        Buscar universitat a la taula:  $O(1)$

        Inserir investigador a la llista d'investigadors per universitat:  $O(1)$

        Inserir investigador a llista d'investigadors per activitat:  $O(1)$

        Inserir investigador a llista d'activitats investigador:  $O(1)$

    Sino

        Comprovar si a la llista d'activitats de l'investigador hi ha algun projecte R+D vigent:  $O(A)$

        Si no te projectes R+D vigents

            Inserir investigador a llista d'investigadores per activitat:

$O(1)$

        Inserir investigador a llista d'activitats investigador:  $O(1)$

### **consultarInvestigadorsUniversitat (universitat): llista investigadors: $O(1)$**

Buscar universitat en taula:  $O(1)$

Retornar la llista associada:  $O(1)$

### **consultarInvestigadorsActivitat (activitat): llista investigadors $O(\log A)$**

Buscar activitat a l'AVL:  $O(\log A)$

Retornar llista associada:  $O(1)$

### **Publicar (publicació, investigador, activitat) $O(\log I + \log A)$**

Buscar investigador a l'AVL:  $O(\log I)$

Buscar activitat a l'AVL:  $O(\log A)$

Inserir a la llista publicacions:  $O(1)$

Incrementar comptador publicacions activitat:  $O(1)$

Si comptador actual > comptador de la activitat més publicada, actualitzar punter a activitat més publicada.  $O(1)$

### **MesPublicada(): activitat. $O(1)$**

Retornar activitat apuntada por punter a activitat més publicada.  $O(1)$

