

2015

Documentació de la Pràctica 5

ESTRUCTURA DE DADES

PAU SANCHEZ VALDIVIESO I ALBERT ESPÍN ROMÁN

| 28.05.2015

Índex

1.	Avaluació de les estructures <i>heap</i> i taula <i>hash</i> implementades	2
2.	Annexos.....	3

1. Avaluació de les estructures *heap* i taula *hash* implementades

L'estructura *heap* organitza els seus elements segons una clau de prioritat que cada un d'ells tenen assignada: els de major prioritat se situen més amunt, i el *heap* regula que no quedin buits en afegir mitjançant les operacions *d'upHeap* i *downHeap* perquè es respecti que cap element tingui un pare de menor prioritat, ja avaluï aquesta condició quan s'ha inserit un element o quan s'ha eliminat l'element de major prioritat.

El fet que el *heap* no deixi buits i vetlli per mantenir en tot moment una estructura anàloga a la de l'arbre complet o quasi-complet garanteix una profunditat de $O(\log_2 n)$, cosa que fa extremadament ràpida la construcció de l'estructura, ja que la inserció d' n elements és $O(n * \log_2 n)$, de manera que si n no es gaire gran resulta negligible (amb *smallText* el temps de construcció és proper a 0 milisegons).

A la cerca sobre el *heap*, en canvi, succeeix una altra cosa: el fet que al llarg de l'estructura hi hagi elements repetits impedeix acabar la cerca d'un valor amb una simple cerca binària; sinó que cal obtenir primerament una llista de valors ordenats i amb compactació dels repetits mitjançant un algorisme de tipus *heapSort*, sobre el qual sí es pot aplicar cerca estrictament binària i assolir la cerca dels elements en temps $O(\log_2 n)$.

La taula *hash* és una estructura amb caselles que tenen una clau i un valor, i està dissenyada de tal manera que indicant una clau l'estructura pugui, després d'una breu operació de *hash*, accedir i retornar l'índex a la taula o el propi valor o valors associats a la clau en la taula. Així, la complexitat de cerca per a cada element és propera a $O(1)$ més la suma del que es triga en realitzar la operació *hash* donada una clau per a obtenir un índex. La inserció d'elements és també molt ràpida perquè donat un valor en genera la clau corresponent i el posiciona en una cel·la de la taula; si ja està ocupada, com s'ha implementat *hash* obert, simplement es guarda seguidament del que ja és present en la cel·la, amb la puntualització que en fer la cerca es filtra dins dels valors de cada cel·la per trobar aquell amb el valor que volem; la mida d'una taula *hash*, per tant, pot oscil·lar entre molt menor que lineal però amb moltes col·lisions i cel·les amb molts elements fins a lineal amb algunes cel·les buides però poques col·lisions: tot depèn de la funció *hash* que es faci servir en un context.

Analitzant els temps de la pràctica, podem observar que tant el *heap* com la taula *hash* tenen un temps de construcció de l'estructura molt similar, en ambdós casos sublineal: al *heap* perquè es requereix poc més que comparacions en cerca binària per l'estructura, de profunditat sublineal; a la taula *hash* perquè només cal emprar la funció *hash* i una simple gestió de col·lisions.

Pel que fa a la cerca, podem veure que la taula *hash* és remarcablement més òptima que el *heap*, i això és gràcies a la seva complexitat relativament propera a la unitat, tenint en compte, però, el cost molt menor que lineal del procés de la funció *hash*; la cerca al *heap* també és ràpida; analitzada a fons, però, es pot veure que requereix una cerca binària sobre la llista ordenada dels elements del *heap*, de manera que cercar un element, quan aquesta llista ja està disponible, és inferior a $O(\log_2 n)$ o com a molt igual si no hi ha elements repetits al *heap*.

2. Annexos

Temps de construcció de l'estructura (inserció de totes les paraules)	smallText	largeText
Heap	1.11523056166 ms	62.9062647782 ms
HashTable	1.12640148548 ms	59.5205439495 ms

Figura 1

Temps de cerca sobre l'estructura	smallText	largeText
Heap	1654.02935222 ms	2594.05700647
HashTable	943.986815827 ms	984.237205879 ms

Figura 2

Profunditat del heap	smallText	largeText
Heap	8 nivells	14 nivells

Figura 3

Dades de la taula hash	smallText	largeText
Nombre de col·lisions	25	738
Nombre d'elements a la cel·la més plena	4	6
Percentatge de cel·les buides	52.7638190955%	75.8381960241%

Figura 4

Text	Nombre de paraules
smallText	199
largeText	10117

Figura 5