

Pràctica 4. Heaps

1. Introducció

Objectiu: Familiaritzar-se amb les estructures de dades de Heap.

Temes de teoria relacionats amb la pràctica: Heap.

2. Enunciat

L'aplicació que es vol implementar és una "base de dades" per guardar les puntuacions d'un conjunt de pel·lícules. Aquesta estructura haurà de mantenir informació sobre les dades de les pel·lícules i permetre accedir a aquesta informació de manera ràpida.

Per a cada pel·lícula es disposarà d'un ID únic, el títol i la seva puntuació (rating).

Un exemple de les dades podria ser:

1::	Toy Story (1995)::	3.87
2::	Jumanji (1995)::	3.40
3::	Grumpier Old Men (1995)::	3.16
4::	Waiting to Exhale (1995)::	2.38
5::	Father of the Bride Part II (1995)::	3.27
...		

I també caldria oferir la possibilitat de poder consultar el títol i la puntuació d'una pel·lícula a partir d'un ID.

En aquesta pràctica es demana implementar aquesta funcionalitat utilitzant una estructura de dades de tipus Heap.

Exercici 1. Heap

Implementeu el TAD **MinHeap** corresponent a un Heap amb una **representació en vector**. Com en el cas de la pràctica anterior, heu de tenir en compte que cada node del Heap estarà representat per un `movieID` (que serà la clau o *key*) i una pel·lícula (que serà el valor o *value*).

L'especificació amb el **mínim** d'operacions necessàries al TAD **MinHeap** és la següent:

- **constructor:** construeix el Heap buit
- **size:** retorna el nombre de nodes que hi ha en el Heap
- **empty:** retorna cert si el Heap està buit, fals en cas contrari
- **insert:** afegeix un nou element al Heap. Aquesta funció rep la clau i el valor/s d'aquesta clau.
- **min:** retorna la clau mínima del Heap
- **minValues:** retorna els valors de la clau mínima del Heap

Estructura de Dades: Pràctica 4

- **removeMin**: elimina el node mínim del Heap
- **printHeap**: imprimeix per consola tot el Heap
- **search**: Busca la clau donada al Heap i retorna el vector de valors associat;

Es poden implementar altres mètodes que siguin necessaris pel desenvolupament de la pràctica, **tot i justificant a la memòria el seu ús** i el seu cost computacional teòric. A la memòria també heu de justificar la representació del TAD MinHeap i el cost computacional teòric de les funcions del TAD MinHeap.

Decidiu vosaltres el TAD per representar dels nodes del Heap i justifiqueu la vostra decisió a la memòria. Es valorarà l'encapsulament de les dades, el cost d'emmagatzemament, i el cost computacional teòric de les operacions associades al TAD definit. Podeu reaprofitar/optimitzar codi de pràctiques anteriors sempre que ho comenteu adientment a la memòria. A més a més, comenteu a la memòria si creieu que una implementació del heap amb enllaços hagués sigut més eficient a la implementació realitzada amb vectors.

Llenceu les excepcions oportunes a cada TAD. El codi s'ha de comentar obligatòriament.

Exercici 2. Cercador de pel·lícules amb un MinHeap

A) Especificació del Cercador de pel·lícules

Escriviu la capçalera d'una classe anomenada **HeapMovieFinder** que es cridarà des del main i ha de realitzar les següents operacions:

- `appendMovies(filename)`: Aquest mètode rep el nom d'un fitxer i emmagatzema el seu contingut a una estructura de dades.
- `insertMovie(movieID, title, rating)`: Aquest mètode rep les dades d'una pel·lícula i fa la inserció a l'estructura de dades corresponent.
- `showMovie(movieID)`: Aquest mètode rep un `movieID` i retorna un sol string amb les dades associades a la pel·lícula.
- `findMovie(movieID)`: Aquest mètode mostra per pantalla el títol de `movieID`, junt amb tota la informació de la pel·lícula.

B) Implementació del Cercador de pel·lícules amb un Heap

Implementeu un cercador de pel·lícules **HeapMovieFinder** que tingui un objecte de tipus **MinHeap**. En aquest objecte **MinHeap** cada node contindrà elements consistents en una `clau` i una dada associada (Movie). La clau serà el `movieID`.

C) Programació del main

A l'algorisme principal s'ha de fer:

- Demanar a l'usuari el nom del fitxer que volem utilitzar amb la pregunta: "Quin fitxer vols (P/G)?" (per petit o gran). Al campus virtual trobareu dos fitxers de text que podeu utilitzar per fer les proves: `movie_rating_small.txt` i `movie_rating.txt`. Aquests fitxers s'hauran d'incorporar a la carpeta del projecte i incloure'ls com a recursos en el projecte.

Estructura de Dades: Pràctica 4

Avaluar el temps d'inserció i mostrar-lo al final de tot. En aquest apartat, a més de demanar el nom del fitxer i llegir-lo, cal:

- Crear un `HeapMovieFinder`.
- Inicialitzar el `HeapMovieFinder` a partir del contingut del fitxer.
- Mostrar el temps de creació del `HeapMovieFinder`.
- Mostrar el MinHeap en amplada. Fer un comptador que demani confirmació cada 40 pel·lícules per seguir mostrant l'índex. NOTA: Es demana un llistat complet del contingut del heap en ordre creixent de `movieID`, junt amb tots els altres camps que hi han associats.
- Llegir el fitxer `cercaPelicles.txt` que us proporcionem i per a cada `movieID` contingut en el fitxer, fer una cerca en el heap del `HeapMovieFinder`. Avaluar el temps de cerca de tots els elements de `cercaPelicles.txt` i comptar el nombre d'elements que estant a `HeapMovieFinder`. Mostrar les dues dades per pantalla.
- Visualitzar per pantalla la profunditat de l'arbre equivalent al Heap.

Implementeu aquestes funcionalitats en forma d'un menú al main que permeti realitzar les 4 accions descrites anteriorment i l'opció 5 per sortir del menú.

Atenció: En aquest exercici sols es demana fer amb templates els TAD MinHeap. El TAD Movie o HeapMovieFinder no s'han de fer amb templates.

Exercici 3. Avaluació d'estructures

Feu una avaluació del rendiment de la implementació anterior (**Heap**):

1. Compteu el temps de generació de l'estructura per les dos llistes de pel·lícules de diferent grandària. Féu les dues proves en el mateix ordinador i en condicions similars.
2. Compteu el temps d'accés (cerca) de les pel·lícules de `cercaPelicles.txt` amb el heap inicialitzat en base als dos fitxers donats.

Raoneu els resultats de temps obtinguts.

Indiqueu quin és el cost computacional teòric de les operacions d'inserció i cerca en el heap.

Memòria

La memòria ha de recollir tots els punts de valoració i raonament indicats anteriorment.

3. Lliurament

A partir de la descripció del problema, es demana:

- Escriure una memòria explicativa que inclogui els següents punts:
 - Portada amb: número de pràctica i títol complet, nom, cognom, NIUB de cada membre del grup, professor/a de pràctiques, número de parella assignat i grup de pràctiques al que assistiu (A, B, C, F)
 - Comentaris de cada exercici: observacions, decisions preses i resposta a les preguntes plantejades, si n'hi ha.
- La memòria s'entregarà en un document **memoriaPX.pdf** on X és el número de la pràctica.
- Implementar els exercicis en C++ . Lliurar el codi C++ corresponent als vostres exercicis en una carpeta anomenada `codi`, amb una subcarpeta per a cada exercici.

Estructura de Dades: Pràctica 4

Com a màxim el dia del lliurament es penjarà en el campus virtual un fitxer comprimit **en format ZIP** amb el nom dels dos membres del grup, el nom del grup (A, B, C o F), el número de parella (ParX) i el número de la pràctica com a nom de fitxer, **GrupA_Par2_BartSimpsonLisaSimpson_P4.zip**, on Par2 indica que és la “parella 2” i P4 indica que és la “pràctica 4”. El fitxer ZIP inclourà: la memòria i la carpeta amb tot el codi.

Els criteris per acceptar la pràctica són:

- La pràctica ha de funcionar en la seva totalitat.
- La pràctica ha de ser orientada a objectes.
- El codi ha d'estar comentat.

IMPORTANT: La còpia de la implementació de la pràctica implica un zero a la nota de pràctiques de l'assignatura per les parelles implicades (tant la que ha copiat com la que ha deixat copiar).

4. Planificació

Lab 6 i 7, Maig de 2018

Per aquesta pràctica els professors proposen la següent planificació:

- **Setmana 1** (*Classe de Laboratori 6*)
 - 1. Implementació del TAD de l'exercici 1 i comentar el codi
- **Setmana 2** (*No hi ha classe presencial*)
 - 1. Implementació del programa principal l'exercici 1 i comentar el codi
- **Setmana 3** (*Classe de Laboratori 7*)
 - 1. Implementació de l'exercici 2 i comentar el codi
- **Setmana 4** (*No hi ha classe presencial*)
 - 1. Implementació de l'exercici 3 i comentar el codi
 - 2. Realització de la memòria completa de la pràctica

La setmana 2 i la setmana 4 s'obrirà una tasca al campus virtual per dipositar la feina feta fins aquell moment. **Recordeu que la setmana 4 s'ha de lliurar la totalitat de la pràctica (tots els exercicis).**

Lliurament: dia 3 de Juny de 2018