

Breu Descripció de les estructures de dades i algorismes utilitzats per a implementar les funcionalitats principals.

Per tal de satisfer totes les nostres necessitats per implementar el projecte, em generat diverses estructures de dades. A continuació us exposem de forma breu la descripció de cada una d'elles ordenades per classes.

Llistat de les classes que contenen les estructures de dades utilitzades:

- Edge
- Node
- Graph
- Pair
- Ponderation
- Reproduction
- User

CLASS EDGE:

Aquesta classe és molt simple degut a que és la representació d'una relació entre nodes. Utilitzarem aquesta estructura de dades per tal de poder separar els nodes en comunitats i així saber quins nodes tenen relació amb quins altres depenen del "pes" que li posem.

CLASS NODE:

Aquesta classe és molt simple degut a que és la representació d'un node dins d'un graf. Utilitzarem aquesta estructura de dades per tal de poder representar o bé una canço o bé una reproducció.

Aquesta estructura conté només un identificador per diferenciar nodes i una funció per tal de saber si un node és igual a un altre node.

CLASS GRAF:

Aquesta classe és més complicada degut a que és la representació o bé de totes les cançons o bé de totes les reproduccions, on els nodes són les cançons o les reproduccions i les arestes (Edge) la relació que hi ha entre elles. La utilitzarem per tal de poder establir les comunitats desitjades amb les ponderacions establertes per l'usuari. No és si més no, una estructura de dades en forma de graf (no binari) per tal d'utilitzar l'algoritme d'ordenació de comunitats que volgum.

Al ser una de les estructures de dades principals, conté moltes funcions, entre elles de control per tal de gestionar-la bé.

CLASS PAIR:

Aquesta classe és necessària degut a que en java no existeix l'estructura pair. La utilitzem sobretot per saber quins dos nodes connecta una aresta. És una classe molt simple que retorna o bé la parella de valors que conté o bé el valor del primer paràmetre o el valor del segon.

CLASS PONDERATION:

Aquesta estructura de dades conté tot el que considerem com rellevant per establir les relacions entre nodes, és a dir, l'autor, l'estil, l'any... etc. A més cada ún dels paràmetres té un factor d'importància que defineix l'usuari, per exemple, l'usuari pot decidir que l'estil és molt més important que no pas l'any (per defecte tots els paràmetres tenen el mateix factor d'importància).

CLASS REPRODUCTION:

Aquesta estructura de dades representa una reproducció. La utilitzem per tal de saber en quin moment un usuari a escoltat una canço. La reproducció conté data i hora, autor i títol. També se sap si ha escoltat en un mateix dia la mateixa canço ja que una reproducció s'identifica per la data i hora.

CLASS USER:

Aquesta classe és una estructura que representa un usuari. L'usuari conté un nom, una edat i una llista de reproduccions. Com el graf és una estructura molt important no només perquè ens identifica els usuaris sinó perquè conté per cada usuari una llista de reproduccions. Aquestes ens serviran per definir relacions entre reproduccions depenent el moment en que s'han reproduït. També el fet de l'usuari en si, és a dir, el seu identificador, és un paràmetre per definir relacions entre cançons i entre reproduccions.

Algorismes utilitzats per a implementar les funcionalitats principals.