Projecte
LP 2019-20

Idea General

Xarxes socials: generen grafs de connexió entre les persones:

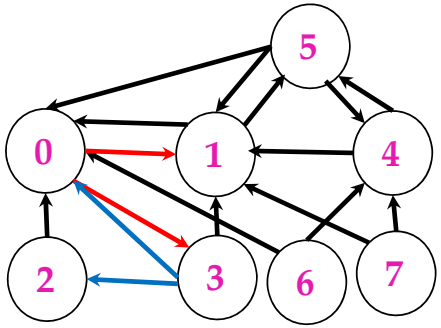
- Qui és amic de qui
- Qui confia en qui etc.

Volem implementar graf:

- Dispers. I no guardar dades innecessàries.
- Amb milers de nodes connectats entre si, i fer consultes sobre ell.
- Utilitzant la seva matriu d'adjacència.

Graf de la xarxa

Graf representa la confiança d'usuaris en la opinió dels altres.



confiança	0	1	2	3	4	5	6	7
0		X		X				
1	X					X		
2	X							
3	X	X	X					
4		X				X		
5	X	X			X			
6	X				X			
7		X			X			

Així l'usuari 0 confiaria en l'usuari 3, i 1, mentre l'usuari 3 confiaria en l'usuari 0 i el 2. Això ho veiem representat en el graf i a la matriu d'adjacència que el representa.

Sparse Matrix

Donada la següent matriu

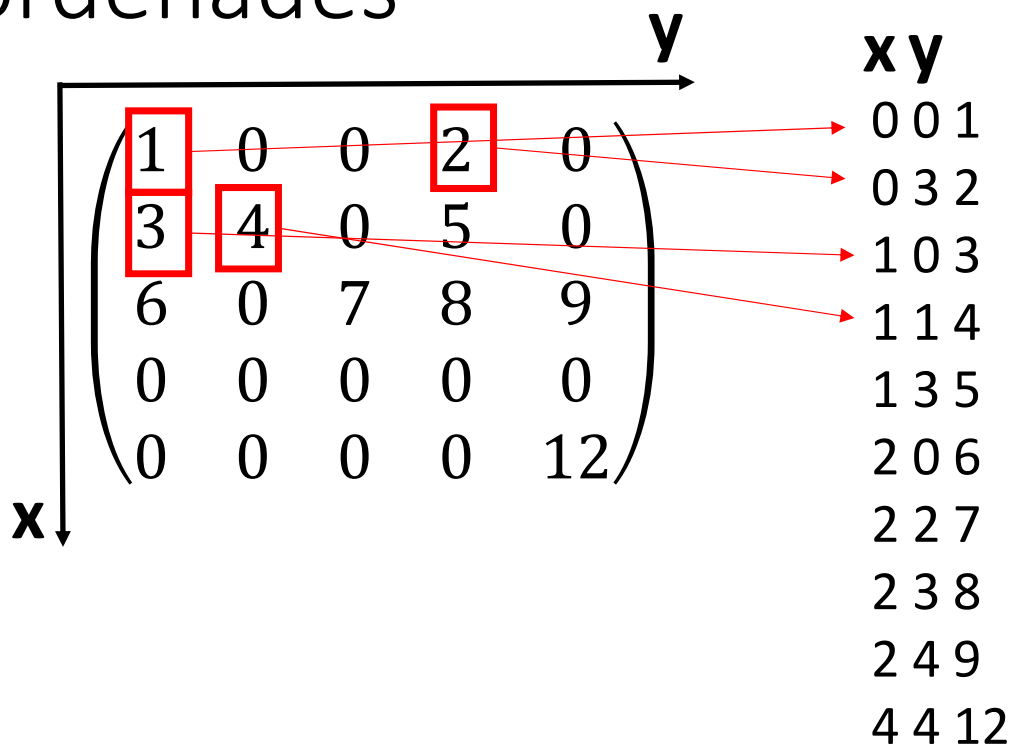
$$\begin{pmatrix} 1 & 0 & 0 & 2 & 0 \\ 3 & 4 & 0 & 5 & 0 \\ 6 & 0 & 7 & 8 & 9 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12 \end{pmatrix}$$

La podem representar:

- Com una matriu de 5 files per 5 columnes i guardar tots els seus valors (0s inclosos)
- Utilitzar altres representacions per guardar només els valors diferents de 0.

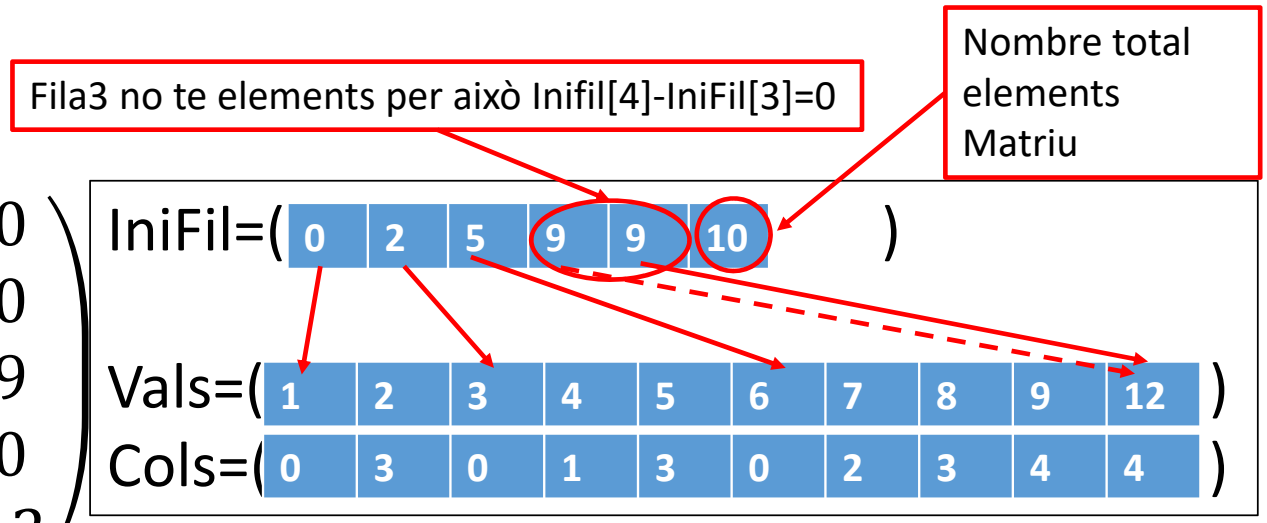
Matriu de Coordenades

Format coordenades



Sparse Matrix

Format CSR $\begin{pmatrix} 1 & 0 & 0 & 2 & 0 \\ 3 & 4 & 0 & 5 & 0 \\ 6 & 0 & 7 & 8 & 9 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12 \end{pmatrix}$



- Vals: Valors no nuls de la matriu ordenats per files.
- Cols: Índex de columna a la matriu dels elements de Vals.
- IniFil: Índex de posició d'inici de cada fila sobre vector Vals.

Implementació

- Volem que analitzeu què passa si utilitzem la matriu implementada a l'exercici 6 i carreguem els dos fitxers de relacions que tenim, mireu la quantitat de memòria que es necessita:
  - Primer el Xarxa1.txt
  - Segon el Epinions.txt
- Volem que implementeu una matriu Sparse tenint en compte les característiques d'aquestes matrius.
- Mireu enunciat per més detalls.

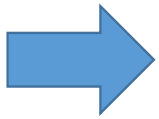
Sparse Matrix Xarxa1.txt

	0	1	2	3	4	5	6	7
0	0	1	0	1	0	0	0	0
1	1	0	0	1	0	1	0	0
2	1	1	0	0	0	0	0	0
3	1	0	1	0	0	0	0	0
4	0	1	0	0	0	1	1	1
5	1	1	0	0	1	0	0	0
6	1	0	0	0	1	0	0	0
7	0	1	0	0	1	0	0	0

Sparse Matrix

Les Matrius sempre serán quadrades: si afegim valor a [9,0] será de 10x10

	0	1	2	3	4	5	6	7
0	0	1	0	1	0	0	0	0
1	1	0	0	1	0	1	0	0
2	1	1	0	0	0	0	0	0
3	1	0	1	0	0	0	0	0
4	0	1	0	0	0	1	1	1
5	1	1	0	0	1	0	0	0
6	1	0	0	0	1	0	0	0
7	0	1	0	0	1	0	0	0



	0	1	2	3	4	5	6	7	8	9
0	0	1	0	1	0	0	0	0	0	0
1	1	0	0	1	0	1	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0
3	1	0	1	0	0	0	0	0	0	0
4	0	1	0	0	0	1	1	1	0	0
5	1	1	0	0	1	0	0	0	0	0
6	1	0	0	0	1	0	0	0	0	0
7	0	1	0	0	1	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	2	0	0	0	0	0	0	0	0	0

Sparse Matrix

- Què passa si una matriu és buida (0 files i 0 columnas)?

  Format CSR: Vals i Cols: No tindrem res; IniFil: | 0 |

  Format coordenades: No tindrem res.

- Què passa si una matriu no té valors diferents a 0?

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Format coordenades no tindrem res

Format CSR:

Vals: No tindrem res

Cols: No tindrem res

IniFil: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Sparse Matrix

- Què passa si les ultimes files estan a 0?

	0	1	2	3	4	5	6	7
0	1	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0
3	0	0	0	0	0	0	0	1
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Format coordenades: no es especial

Format CSR:

Vals:

1	1	1	1

Cols:

0	1	2	7

Inifil:

0	1	2	3	4	4	4	4	4

Hauríem de vigilar a l'accedir a la matriu de no sortir de la mida del vector

Excepcions

- **Excepció:** "succes excepcional". És un event que es dona durant l'execució del programa que interromp el fluxe normal de les sentencies. Són el mecanisme per gestionar els errors d'execució que es produeixen en un programa, i serveixen per:
  - Donar un avis de que es pot produir un error.
  - Transferir la gestió dels errors a fragments de codi específicament destinats a això.
  - Quan apareix una condició excepcional es crea un objecte *Throwable que s'envia al mètode que l'ha generat*
- La gestió de les excepcions:
  - Permet detecció i correcció d'errors en execució
  - Simplifiquen els programes donat que es diferencia entre el codi normal i el de tractament d'errors
  - Es creen programes més robustos donat que (en molts casos) el programa no compila sense codi de tractament d'excepcions.
  - Només s'han d'utilitzar quan no es poden resoldre les situacions anòmales directament en el context, s'ha de seguir fent el control d'errors habitual

Excepcions a C++

Exemple alocatació de memòria

```
int main ()
{ ...
  try {  ...
          //crides a diferents mètodes que alocaten memòria
         //i a mètode que llença una excepció en forma de texte.
  }
  catch (std::bad_alloc) {
             cout << “Error allocatar memòria”<< endl;
  }
  catch (const char* msg) {
             cout << msg << endl;
  }
```

Excepcions a C++

Exemple divisió nombres double:

```
#include <iostream>
using namespace std;

double division(int a, int b)
{   if( b == 0 )
    {throw "Division by zero condition!"; }
    return (a/b);
}
```

```
int main ()
{  int x = 50;
   int y = 0;
   double z = 0;
   try
   {   z = division(x, y);
       cout << z << endl;
   }
   catch (const char* msg)
   {   cout << msg << endl;
   }
```

Excepcions a C++

```
int main()
{ try
  {  MatriuSparse m(nomFitxer);
     cout << m;
  }
  catch (std::bad_alloc)
  {  cout <<"Comment ==> error allocatant memoria:" << endl;
  }
  catch (const char* msg)
  {  cout << msg << endl;
  }
}
```

```
MatriuSparse::MatriuSparse(string nomFitxer)
{  ifstream fitxer;
   fitxer.open(nomFitxer);
   if (fitxer.is_open())
   {
   }
   else
   {   throw "ERROR LECTURA FITXER";
   }
}
```

Observacions

- Treballarem amb els fitxers d'entrada ordenats per tal de reduir el temps d'execució i la complexitat dels algorismes. Per això els fitxers utilitzats seran:
  - Xarxa1.txt: ja estava ordenat
  - SubEpinions2Ordenat.txt
  - EpinionsOrdenat.txt
- Per tal de testejar que la matriu està ben construïda escriurem a un fitxer les files i columnes diferents de 0. Això ho farem a partir de l'operador << que teniu definit. Per poder fer les comparacions pertinents haureu de donar la sortida en un format determinat que us passarem a l'enunciat.
- Donat que tenim dues possibles implementacions de la matriu tindrem dos lliuraments diferents; un per Coordenades i un per CSR.