

# Pràctica de Grafs

## Algorísmica Avançada

Grau en Enginyeria Informàtica  
Primer Quadrimestre Curs 2012-2013  
Universitat de Barcelona

## 1 Ordenació Topològica

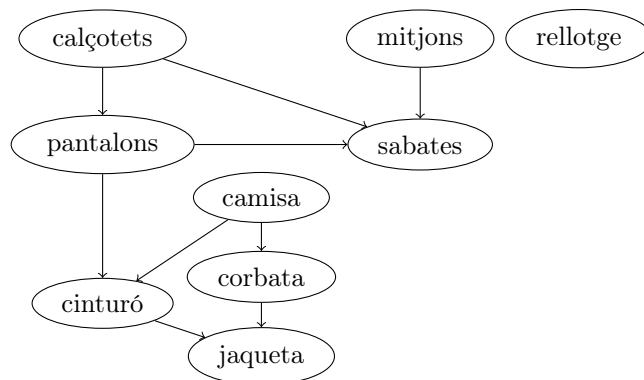
Tal com s'ha explicat a classe de teoria, un graf és acíclic quan no hi existeix cap cicle, és a dir, no hi ha un camí que porti d'un node a ell mateix, llevat del camí buit.

Un graf dirigit també s'anomena *dígraf*. En un dígraf, les arestes són parelles ordenades de nodes. És a dir, tenen *direcció*. Les arestes comencen en un node *cua* i acaben en un altre node *cap*. O sigui, van de la cua al cap. I enlloc de línies es representen per fletxes. El fet que les arestes tinguin direcció és fonamental. Entren a escena conceptes com nodes *antecessors* i nodes *successors*. Tan gran és la transcendència d'aquesta condició, que enlloc d'arestes les anomenem *arcs*. Així doncs, quan parlem de dígrafs o grafs dirigits utilitzem el terme arcs per anomenar les arestes.

Els grafs dirigits acíclics són un tipus de graf especialment freqüent. Sovint s'anomenen amb les sigles *dag*. Entre altres àmbits, apareixen com a grafs que serveixen per expressar precedències entre accions en el temps. És a dir, grafs on cada node representa una acció, i un arc  $a = (u, v)$  que va d'un node  $u$  a un node  $v$ , significa que l'acció  $u$  ha de ser efectuada abans que l'acció  $v$ .

Per altra banda, *topo* vol dir punt. I lògica, de logia, vol dir estudi, percepció calculable, manera d'entendre, model humà del fenomen. Topologia, doncs, vol dir lògica del punts. I una ordenació topològica, com ja podeu suposar, és una seqüenciació dels nodes d'un *graf de precedències* tal que cap node successor aparegui abans que els seus nodes antecessors.

El funcionament de tot plegat s'il·lustra amb un exemple extret del llibre de R. Sedgewick (Sedgewick, 1992). Es tracta d'analitzar el procés habitual de vestir-se per un ésser humà de gènere masculí. Un home. El graf que podríem obtenir, fruit de l'anàlisi, podria ser el següent,



I a partir d'aquest graf, una possible ordenació topològica seria una possible manera de vestir-se per un home: *mitjons, calçotets, pantalons, sabates, rellotge, camisa, cinturó, sabates i jaqueta*.

## 2 Enunciat

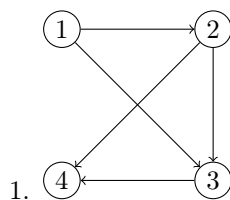
Donat un graf acíclic dirigit, implementeu un recorregut tal que ens proporcioni una ordenació topològica dels nodes. Feu una anàlisi d'eficiència, que podeu posar com a comentari dins el mateix arxiu font.

Els grafs d'entrada que s'utilitzaran per provar el vostre codi seran grafs on els nodes vinguin representats per enters. Utilitzeu la rutina `llegir_graf(G)` que es mostra a continuació per fer la lectura a partir d'un fitxer de text pla on es llistin els successors de cada node.

```
def llegir_graf():  
    global G  
  
    nom = raw_input("Doneu el nom del graf: ")  
    G = nx.read_adjlist(nom,create_using=nx.DiGraph(),nodetype = int)
```

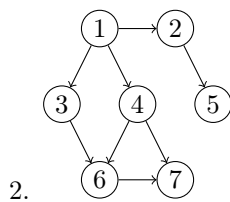
### entrades i sortides

Seguidament es mostren dos exemples. A la primera columna hi ha el graf, a la segona el contingut del fitxer de text que el representa, i en la tercera, la sortida que hauria de donar l'exercici.



```
1 2 3  
2 3 4  
3 4
```

```
[1,2,3,4]
```



```
1 2 3 4  
2 5  
3 6  
4 6 7  
6 7
```

```
[1,4,3,6,2,5,7]
```

### 3 Lliurament

El lliurament d'aquest exercici s'haurà de fer penjant un arxiu de nom amb el format

*G\_nom\_cognom.py*

en el campus virtual. Poseu el vostre grup de pràctiques enlloc de la G.

Poseu també el vostre nom en la primera línia d'aquest fitxer.

El termini pel lliurament finalitza el diumenge 4 de novembre de 2012 a les 23.59 hores.

### References

Sedgewick, R. (1992). *Algorithms in C++. Part 5. Graph Algorithms*. Addison-Wesley.