

# Problemes de Grafs

Algorísmica Avançada

Grau en Enginyeria Informàtica  
Primer Quadrimestre Curs 2014-2015  
Universitat de Barcelona

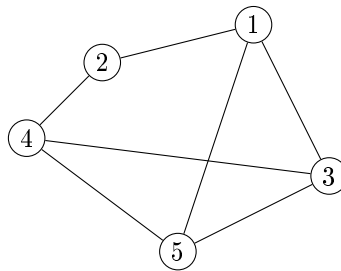
Dels problemes que s'enuncien a continuació, per als dos primers no es demana cap implementació en python, tan sols un document amb les explicacions pertinents, que pot ser aquest mateix, sobreescrit. El tercer problema s'ha de respondre en part en el mateix document, i també amb la codificació de la funció demanada en un arxiu de nom `aillats.py`. Els dos últims exercicis s'entregaran en fitxers anomenats `recorreguts.py`, i `articulacio.py`.

1. Dibuixeu el graf  $G(V,E)$  amb els conjunts

- $V = \{1, 2, 3, 4, 5\}$
- $E = \{\{1, 2\}, \{1, 4\}, \{2, 3\}, \{2, 5\}, \{3, 5\}\}$

Quantes components connexes té?

2. Descriuiu com seria el contingut de la matriu d'adjacències i de les llistes d'adjacència que representessin el graf de la figura.

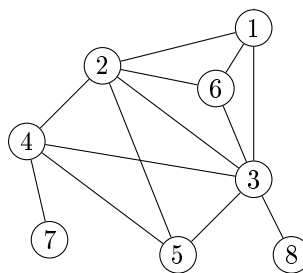


3. Expliqueu quina diferència hi ha entre l'ordre i la mida d'un graf. Raoneu-la. Esteu d'acord amb aquesta terminologia? Podeu imaginar quina característica se suposa al graf per justificar-la? Com a pista, la pàgina 155 del llibre. Si anomenem  $n$  a l'ordre d'un graf, codifiqueu en python un algorisme per saber el nombre de vèrtexos aïllats d'un graf, en  $\Theta(n)$ .
4. Codifiqueu en python un algorisme que recorri en amplada i en profunditat un graf i per cada node n'imprimeixi el seu número identificador, sense utilitzar funcions macroscòpiques. És a dir, usant tan sols el recorregut de nodes del graf, el recorregut dels veïns d'un node, i una llista pel cas del recorregut en amplada. Podeu suposar que les adjacències estan ordenades ascendentment per número de vèrtex. Tot seguit es mostra el contingut d'un arxiu amb un graf descrit en llistes d'adjacència, i la seva representació gràfica.

```

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

```



Podeu usar-lo com a joc de proves. El resultat hauria de ser

```

BFS: 1,2,3,6,4,5,7,8
DFS: 1,2,3,4,5,7,8,6

```

5. Un punt d'articulació d'un graf és un vèrtex tal que si es treu del graf el nombre de components connexes augmenta. En el graf de la següent figura el vèrtex 3 és un punt d'articulació. Implementeu un algorisme en python que donat un graf respongui si hi existeix algun punt d'articulació. Analitzeu l'eficiència del vostre algorisme.

