



Model	Prova pràctica
B	Pàgina 1 de 7

INSTRUCCIONS:

- Per garantir l'anonimat de la prova s'invalidarà qualsevol exercici escrit que inclogui noms, marques o qualsevol senyal que pugui identificar l'aspirant.
- L'aspirant ha d'entregar grapats, ordenats i numerats (Núm. Pàgina / Total Pàgines) tots els fulls emprats durant la prova. A la primera pàgina entregada ha d'enganxar el codi. Les darreres pàgines lliurades corresponen als fulls emprats com a esborranys.
- Una vegada acabada la prova s'ha de signar el full de registre de la part B i retornar els enunciats de la prova.
- Es disposa de 3 hores per realitzar la prova.

ACLARIMENTS:

- En cas que els enunciats no fixin clarament alguns aspectes, cal argumentar les decisions preses.

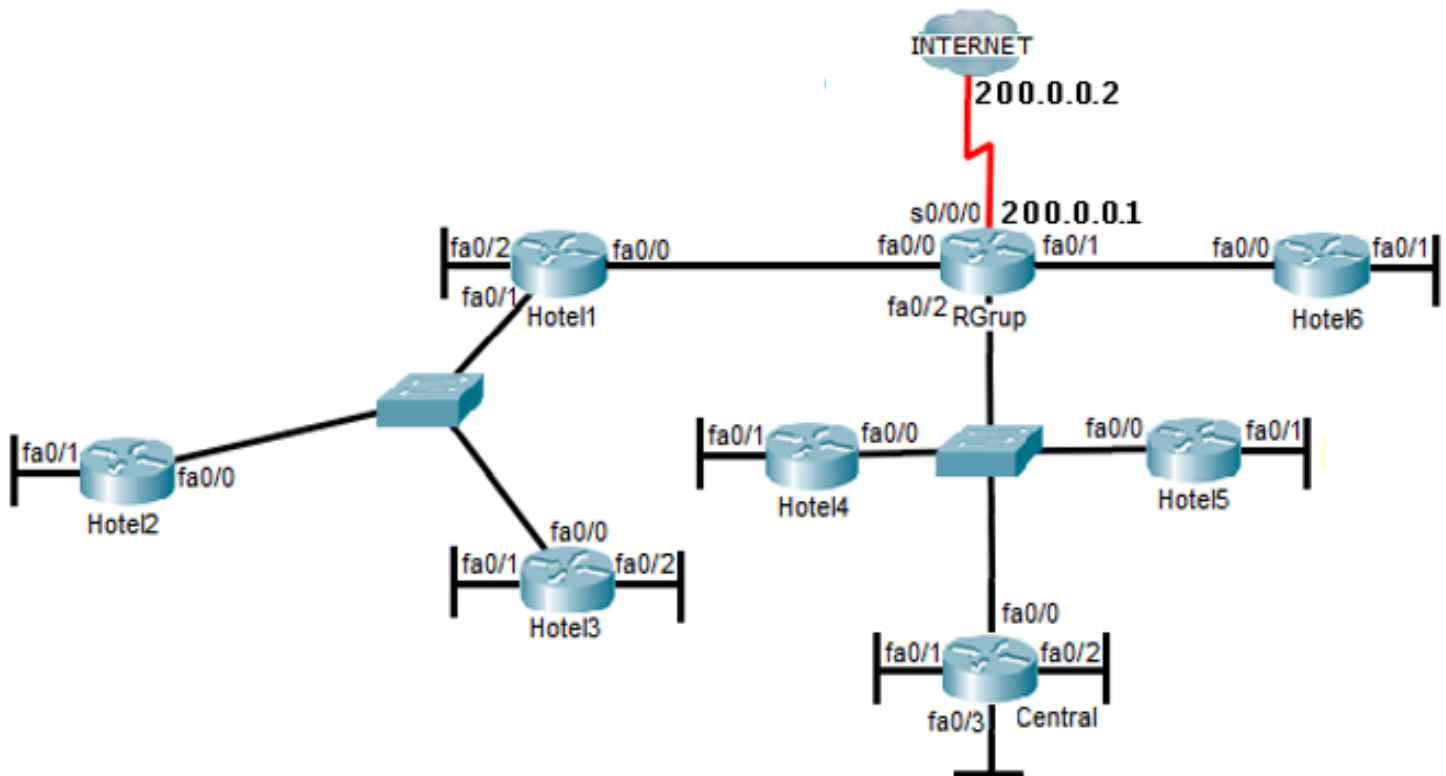
NO ES TINDRÀ EN COMPTE:

- Les parts tatxades o il·legibles de l'exercici.



EXERCICI 1

El següent diagrama mostra la topologia d'una xarxa WAN (entre encaminadors) que interconnecta cada un dels hotels d'un grup hoteler. A més, es mostren les xarxes LAN en cada un dels hotels així com les de les oficines centrals.



SIMBOLS:



Xarxa LAN



Commutador



Encaminador

Model B	Prova pràctica
	Pàgina 3 de 7

→ ES DEMANA:

1. (50%) Adreça les xarxes LAN i WAN presents a la topologia, tenint en compte que:
 - Els dispositius sols permeten l'adreçament IPv4.
 - Cada xarxa LAN té un màxim de 60 nodes i aquest nombre no està previst que es modifiqui en el futur.
 - Cal adreçar les xarxes WAN a partir del bloc 100.0.0.128 /25. Degut a que les adreces són públiques, cal aplicar la tècnica d'adreçament que estalviï més IP.

2. (40%) Implementa l'encaminament estàtic en cada encaminador per tal que totes les xarxes LAN es puguin connectar entre si i, a més, totes es puguin connectar a INTERNET. Cal realitzar la introducció de cada ruta segons la següent especificació:
 - *ip route destinationNetwork IP_nextHop*
 Exemple: `ip route 192.168.0.0 255.255.255.0 100.0.0.1`

 - o bé:
 - *ip route destinationNetwork interfaceToExit*
 Exemple: `ip route 192.168.0.0 255.255.255.0 fa0/0`

Una taula com la següent per a cada encaminador pot ajudar a organitzar la resposta dels apartats anteriors:

ENCAMINADOR	INTERFÍCIE	ADREÇA IP / MÀSCARA XARXA
<i>Nom_encaminador</i>	<i>Interfície1</i>	
	...	
RUTES ESTÀTIQUES APLICADES		
ip route ...		
...		

3. (10%) Tenint en compte que cada adreça pública emprada ens ha costat 1 EUR, quant ens ha costat l'adreçament ?

→ ES VALORARÀ POSITIVAMENT:

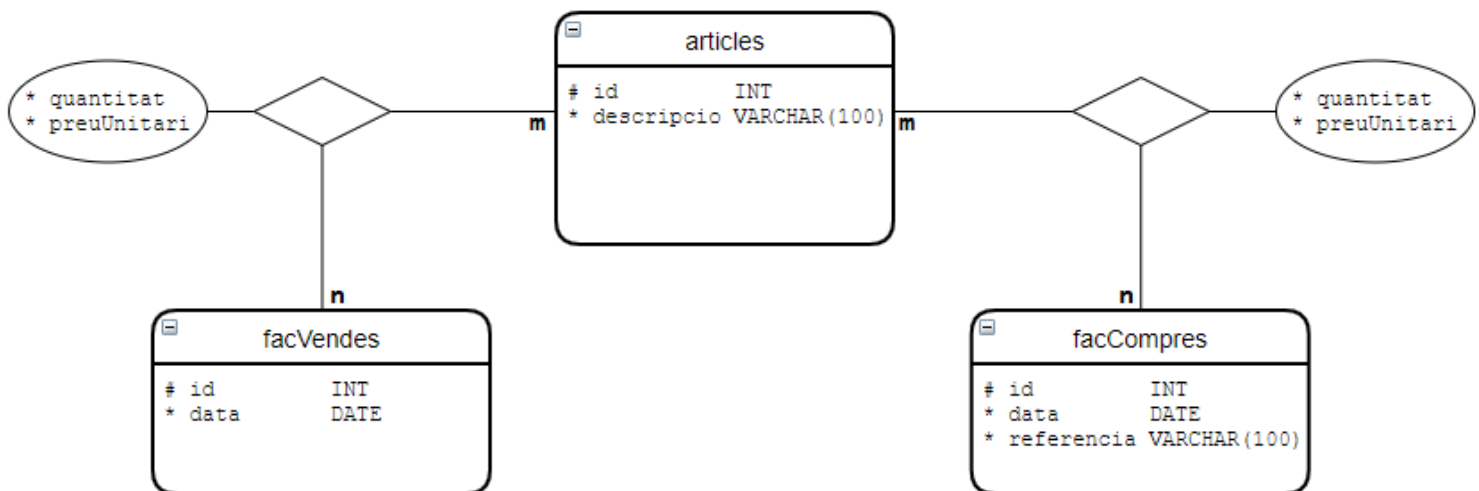
1. L'estratègia seguida en l'adreçament LAN per tal d'afavorir la màxima sumarització de les rutes en l'encaminament estàtic.
2. La simplicitat en l'adreçament de les xarxes.
3. La correcció en l'adreçament LAN, WAN i en l'encaminament estàtic.



EXERCICI 2

El següent model relacional representa una part del model de dades d'una empresa que es dedica a la compra i venda d'articles de tot tipus. La modelització que es mostra correspon a una part de la gestió del magatzem.

El model de negoci d'aquesta empresa consisteix, essencialment, en comprar articles a l'engròs a Proveïdors per, més endavant, posar-los a la venda a la menuda a diferents clients. Cada una d'aquestes transaccions s'explicita en el sistema mitjançant una factura.



Per tal de millorar el rendiment de la BBDD, s'ha decidit afegir un atribut 'totalArticles' a la taula 'articles' amb la intenció que sigui una columna calculada. El càlcul que ha de contenir ha de ser el següent:

- Les compres associades a un article han d'incrementar 'totalArticles' segons l'atribut 'quantitat' de la relació entre 'articles' i 'facCompres'.
- Les vendes associades a un article han de decrementar 'totalArticles' segons l'atribut 'quantitat' de la relació entre 'articles' i 'facVendes'.

Model	Prova pràctica
B	Pàgina 5 de 7

→ ES DEMANA:

1. (5%) Passa el diagrama E/R que es mostra a l'enunciat al diagrama relacional que se'n deriva. La finalitat d'aquest apartat és anomenar els objectes per tal que puguin ser emprats en el segon apartat.
2. (70%) Cal saber que la BBDD està creada a partir dels objectes esmentats en l'apartat anterior i a més està en producció. Per tant, les taules contenen dades reals. Implementa els canvis, escrivint un script en SQL estàndard (Oracle o Mysql) en el que cal combinar DDL i DML que permeti solucionar el problema en una sola execució.

NOTA: cal deixar actualitzat el valor del nou atribut 'totalArticles' segons les dades existents actualment a la base de dades.

3. (25%) Una vegada acabada l'execució de l'script, mostra els articles que no tenen cap venda associada.

→ ES VALORARÀ POSITIVAMENT:

1. L'estratègia seguida per tal de resoldre el problema d'una manera lògica.
2. La sintaxi de les instruccions proposades.

→ ES PENALITZARÀ:

1. El model relacional obtingut no correspon al model E/R que es mostra.



Model B	Prova pràctica
	Pàgina 6 de 7

EXERCICI 3

Es vol crear un script que, utilitzant una API proporcionada Github, ens permeti trobar l'amistat més propera que sàpiga programar en un llenguatge determinat.

Per fer això, suposarem les següents condicions:

- Mitjançant l'API de GitHub podem consultar la llista d'amistats de qualsevol usuari.
- Per llista d'amistat s'entén els usuaris que un usuari està seguint (*following users*).
- Mitjançant aquesta API de GitHub també podem consultar la llista de llenguatges de programació que utilitza qualsevol usuari.
- A partir d'un usuari determinat, volem trobar un usuari qualsevol que l'usuari segueixi directament (amistat de grau 1) que compleixi la condició que utilitza un llenguatge de programació determinat.
- Si no hi ha cap usuari de grau 1 que compleixi aquesta condició, volem trobar qualsevol amistat de les nostres amistats (amistats de grau 2) que compleixin la condició.
- Si no hi ha cap usuari de grau 2, seguirem pels usuaris de grau 3, 4, etc... fins a trobar el primer usuari que compleixi la condició desitjada.
- Hem de tenir en compte que el nombre d'usuaris de GitHub és finit.
- En cas de que no trobi cap usuari ha de mostrar un missatge d'error.

Model	Prova pràctica
B	Pàgina 7 de 7

→ ES DEMANA:

1. (60%) Crea l'algoritme que realitzi aquesta cerca. L'algoritme pot estar escrit en pseudocodi, Python, Java o JavaScript.
2. (30%) Especifica les estructures de dades utilitzades per l'algoritme.
3. (10%) Calcula el cost d'execució de l'algoritme.

→ ES VALORARÀ POSITIVAMENT:

1. L'estratègia seguida per tal de resoldre el problema d'una manera lògica.
2. L'eficiència de les estructures de dades i l'algoritme proposat.
3. La sintaxi de les instruccions proposades. Si s'utilitza pseudocodi es valorarà la coherència entre les diferents instruccions.
4. Que no se produeixin situacions no desitjades, com ara, bucles infinits o que el programa s'executi indefinidament.

→ ES PENALITZARÀ:

1. Que es produeixin situacions no desitjades, com per exemple, bucles infinits o que el programa s'executi indefinidament.
2. Que el cost d'execució no sigui òptim.

→ NO ES TINDRA EN COMPTE:

1. No és necessari implementar les consultes de l'API de GitHub. Suposarem que existeixen unes funcions que ja ens retornen aquesta informació.