XUNTA DE GALICIA
CONSELLERÍA DE CULTURA, EDUCACIÓN E UNIVERSIDADE

Nome e apelidos	DNI/NIE	OPCIÓN EXAME

ORDE DO 24 de febreiro de 2020 (DOG do 13 de marzo de 2020). SERVIZO SECUNDARIA.
CORPO: 590 - Profesores de ensino secundario
ESPECIALIDADE: 107 - Informática

PRIMEIRA PROBA – PARTE A

INSTRUCIÓNS

- A duración total desta parte é de 3 horas.
- Entrégase un caderno con dúas opcións de exame.
- O/a aspirante escollerá e realizará unha das dúas opcións de exame propostas, que deberá indicar no recadro correspondente na cabeceira desta páxina e na do caderno de respostas, xunto co seu nome, apelidos e o seu DNI/NIE.
- As follas grampadas non se poden separar.
- De precisar algún folio adicional, este será solicitado a un membro do tribunal, quen o grampará ao resto dos folios.
- As respostas deben realizarse con bolígrafo de tinta negra ou azul, que non se poida borrar. Non se permite o uso de cintas ou líquidos correctores. Non se corrixirá nada a lapis ou con tinta que se poida borrar.
- Non se admiten preguntas. Calquera aclaración ou interpretación do enunciado que o/a aspirante considere oportuno para a súa valoración, deberá indicala no caderno de respostas.
- Non se pode utilizar ningún material auxiliar, e están prohibidos os móbiles, reloxos, mans libres ou calquera outro dispositivo electrónico, que deberán estar **completamente apagados** e gardados nos bolsos debaixo do propio asento.

OPCIÓN A

EXERCICIO 1

Consideracións xerais:

A fin de simplificar no posible a tarefa de realizar en papel exercicios de programación establécese o seguinte:

- Nos exercicios nos que se pide unha única función estática Java non é necesario completar a sintaxe da clase na que debería estar incluída, basta escribir exclusivamente o código da función en si.
- Se nun destes exercicios o aspirante escribe funcións auxiliares coas mesmas características de accesibilidade que a función pedida, pode proceder de igual maneira.
- Se se indica nalgún exercicio a posibilidade de utilizar funcións existentes sen ter que escribilas explicitamente estas funcións considerararanse accesibles no contexto no que están sendo utilizadas sen necesidade de que o aspirante indique ***import*** ou calquera outra condición necesaria para prover esa accesibilidade para a súa execución en contorno real.

EXERCICIO 1.1

Escribir unha función na linguaxe de programación Java chamada ***superficieSolapada*** coa seguinte firma:

```
private static int superficieSolapada(Rectangulo r1, Rectangulo r2)
```

que reciba como parámetros dous obxectos da clase ***Rectangulo*** representando cada un deles un rectángulo de lados paralelos ós eixos de coordenadas, dado por catro valores enteiros correspondentes ós seus lados superior, dereito, inferior e esquerdo e devolva a superficie do rectángulo intersección de ambos.

- A clase ***Rectangulo*** ven dada e é a que se indica na *Figura 1*.
- Úsanse coordenadas cartesianas. As coordenadas horizontais medran cara a dereita e as coordenadas verticais medran cara arriba.
- As coordenadas poden ser negativas.
- Se os dous rectángulos non se solapan, a función devolverá 0.
- Considerarase que os dous rectángulos recibidos en cada un dos parámetros son sempre correctos. En particular o lado superior sempre terá un valor maior que o inferior e o lado dereito sempre terá un valor maior que o esquerdo.
- Considerarase que o valor de superficie solapada que se debe calcular é sempre suficientemente pequena para poder almacenala nun ***int***.
- Non se permite a utilización da clase estándar de Java ***Rectangle*** nin de calquera outra similar de librería que poida ter incluída unha funcionalidade que facilite a realización do exercicio. O algoritmo debe realizarse utilizando exclusivamente as operacións aritméticas básicas *, +, / e – e as estruturas de control de programación bifurcación condicional e bucles.

A modo de exemplo, na *Figura 2* mostrase o rectángulo que corresponde co obxecto ***new Rectangulo(3, 5, -2, -1)***.

Rectangulo.java

```
1
2  public class Rectangulo {
3      private int arriba;
4      private int dereita;
5      private int abaixo;
6      private int esquerda;
7
8      public Rectangulo(int arriba, int dereita, int abaixo, int esquerda) {
9          super();
10         this.arriba = arriba;
11         this.dereita = dereita;
12         this.abaixo = abaixo;
13         this.esquerda = esquerda;
14     }
15
16     public int getArriba() {
17         return arriba;
18     }
19     public int getDereita() {
20         return dereita;
21     }
22     public int getAbaixo() {
23         return abaixo;
24     }
25     public int getEsquerda() {
26         return esquerda;
27     }
28
29 }
```

Figura 1

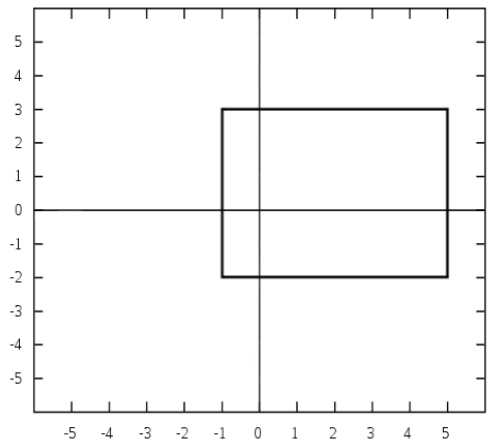


Figura 2

EXERCICIO 1.2

Escribir unha función na linguaxe de programación Java chamada ***anagrama*** coa seguinte firma:

```
private static boolean anagrama(String palabra1, String palabra2)
```

que reciba como parámetros dous ***String*** contendo dúas palabras e devolva un ***boolean*** indicando se unha das palabras é ou non un anagrama da outra, é dicir, que se pode obter a partir das letras da outra sen máis que reordenalas.

- Considerarase que as palabras están escritas en minúsculas, sen til nin outros signos diacríticos e sen espazos en branco ou outros caracteres distintos dos alfabéticos da "A" á "Z".
- Se as dúas palabras son iguais tamén se considera que son anagrama.

EXERCICIO 1.3

O cifrado por substitución consiste en designar unha letra que substitúa a cada unha das letras do alfabeto. Así, neste tipo de cifrado, unha clave consiste nunha permutación do alfabeto, de xeito que a primeira letra que aparece nesa permutación será a que substitúe á "a", a segunda será a que substitúe á "b" e así sucesivamente.

Escriba unha función na linguaxe de programación Java chamada ***cifrar*** coa seguinte firma:

```
private static String cifrar(String texto, String clave)
```

que reciba como parámetros un ***String*** contendo un texto a cifrar e outro ***String*** contendo a clave de cifrado e devolva o resultado de cifrar o texto do primeiro parámetro segundo o sistema de cifrado por substitución coa clave subministrada no segundo parámetro.

Por exemplo, a palabra "abade" cifrada coa clave "zaxbyc...." quedaría "zazby".

Teña en conta as seguintes consideracións:

- O texto está escrito en minúsculas, sen til, eñes, nin outros signos diacríticos e sen espazos en branco ou outros caracteres distintos dos alfabéticos da "A" á "Z".

- A clave subministrada é correcta, contén as 26 letras da "A" á "Z", aparecendo cada letra unha soa vez.

EXERCICIO 2

EXERCICIO 2.1

Desenvolva un programa na linguaxe de programación **Python** para determinar se unha cadea de caracteres (palabra/frase) é palíndromo, é dicir, se se pode ler igual de esquerda a dereita que de dereita a esquerda. Un exemplo de palíndromo é a palabra "radar".

Desenvolva a seguinte función:

Función ***palindromo***: cos parámetros de entrada que considere preciso para resolver de **forma recursiva** se unha palabra/frase é palíndromo. O valor de retorno será ***True*** ou ***False*** en función do caso. (Exemplo: "radar" é palíndromo, polo tanto, a función devolverá ***True***).

O arquivo co programa será gardado como "***palindromo.py***" e será executado como un ***script*** ao que se lle pasará como primeiro e único parámetro, a palabra/frase a consultar.

No corpo principal do programa chamarase a función ***palindromo*** coa palabra/frase que se pasa por parámetro ao ***script*** e deberase mostrar por pantalla como resultado as mensaxes "É un palíndromo" ou "NON é un palíndromo" en función do resultado.

Exemplo de execución do ***script*** e resultado que debe dar:

```
>python palindromo.py "radar"
É un palíndromo
```

Complete o seguinte código do arquivo "***palindromo.py***" con todo o que sexa preciso:

```
def palindromo # A COMPLETAR

if __name__ == "__main__":
        import sys
        # A COMPLETAR
```

EXERCICIO 2.2

Desenvolva un programa na linguaxe de programación **Python** facendo uso da estrutura de datos **dicionario** para gardar, asociado a cada DNI do/a opositor/a, a nota acumulada dos exercicios deste exame práctico. O DNI estará composto por oito díxitos e un carácter alfabético de control que será preciso validar. O algoritmo de verificación do DNI consiste en dividir o valor numérico do DNI entre 23. O resto desa operación validarase coa seguinte táboa para saber a letra que lle corresponde:

Resto	0	1	2	3	4	5	6	7	8	9	10	11
Letra	T	R	W	A	G	M	Y	F	P	D	X	B

Resto	12	13	14	15	16	17	18	19	20	21	22
Letra	N	J	Z	S	Q	V	H	L	C	K	E

Dita táboa debe ser gardada no programa como un dicionario. Soamente será preciso indicar o comezo da declaración incluíndo os dous primeiros elementos 0 e 1 e os dous últimos 21 e 22. O resto da declaración indicarase por puntos suspensivos (…) aínda que sintacticamente non é correcto.

Desenvolve as seguintes funcións:

- Función ***comprobarDNI***: devolverá ***True*** ou ***False*** se a letra de verificación é correcta. Non é necesario validar que a entrada é unha cadea de 8 díxitos e 1 letra.
- Función ***gardarNota***: tendo como parámetros de entrada o dicionario, o DNI do/a opositor/a e a puntuación a acumular. Se no dicionario non existe unha nota asociada a dito DNI, gardarase a nova nota. Se xa existe unha nota asociada, a nova nota acumularase ao valor xa existente.
- Función ***consultarNota***: tendo como parámetros de entrada o dicionario e o DNI do/a opositor/a. Se no dicionario non existe unha nota asociada a dito DNI, mostrarase unha mensaxe "DNI de opositor/a incorrecto". Se está rexistrada a nota dese/a opositor/a, mostrarase por pantalla "A nota acumulada do DNI XXXXXXXX é: YY".

Complete o seguinte código con todo o que considere preciso:

```
def comprobarDNI(dni) # A COMPLETAR

def gardarNota(diccionario, dni, nota) # A COMPLETAR

def consultarNota(diccionario, dni) # A COMPLETAR

if __name__ == "__main__":
	# A COMPLETAR
	gardarNota(dicionario,'11111111H',1.5)
	gardarNota(dicionario,'11111111H',2.5)
	gardarNota(dicionario,'22222222B',2.1)
	consultarNota(dicionario,'11111111H')
```

No corpo principal do programa chamarase a función de ***gardarNota*** tres veces cos seguintes datos: asociado ao DNI 11111111H gardar a nota 1.5, asociado de novo ao mesmo DNI 11111111H a nota 2.5 e asociado ao DNI 22222222B a nota 2.1. Deberase imprimir por pantalla como resultado a nota do DNI 11111111H coa función ***consultarNota***.

O resultado da execución do programa cos datos anteriores debe ser o seguinte:

```
A nota acumulada do DNI 11111111H é:  4.0
```

EXERCICIO 2.3

Partindo do seguinte documento XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<reservas xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="reservas.xsd">
	<reserva_habitación cliente="C_53454123X" opciones="AD" tipoHabitación="Doble">
		<fechaEntrada año="2021" mes="Junio" día="18"></fechaEntrada>
		<fechaSalida año="2021" mes="Junio" día="19"></fechaSalida>
		<observaciones>LLegan sobre las 20:30</observaciones>
	</reserva_habitación>
	<reserva_habitación cliente="C_32555111A" opciones="MP"
tipoHabitación="Individual">
		<fechaEntrada año="2021" mes="Agosto" día="2"></fechaEntrada>
		<fechaSalida año="2021" mes="Agosto" día="4"></fechaSalida>
	</reserva_habitación>
	<reserva_salón cliente="C_53454123X">
```

```
        <fechaEntrada año="2021" mes="Octubre" día="25"/>
        <fechaSalida año="2021" mes="Octubre" día="25"/>
    </reserva_salón>
    <reserva_salón cliente="C_44545123A">
        <fechaEntrada año="2021" mes="Agosto" día="15"/>
        <fechaSalida año="2021" mes="Agosto" día="15"/>
        <observaciones>Servicio de catering</observaciones>
    </reserva_salón>
    <cliente id="C_53454123X">
        <nombre>Clara</nombre>
        <apellidos>Lago Grau</apellidos>
        <móvil>655656777</móvil>
    </cliente>
    <cliente id="C_44545123A">
        <nombre>Fernando</nombre>
        <apellidos>Simón Soria</apellidos>
        <móvil>785567811</móvil>
    </cliente>
    <cliente id="C_32555111A">
        <nombre>Francisco</nombre>
        <apellidos>Castro Ramos</apellidos>
        <móvil>649567811</móvil>
    </cliente>
    <cliente id="C_37545123L">
        <empleado>E2_4</empleado>
    </cliente>
</reservas>
```

Cree os elementos do XML Schema necesarios para realizar as seguintes tarefas:

a) Defina o elemento ***reserva_habitación*** creando un grupo que se poida reutilizar para definir o elemento ***reserva_salón***. Defina tamén o grupo creado.

b) Defina un tipo de dato chamado ***tipoIdCliente*** para reutilizar cos atributos **cliente** e **id,** tendo en conta que sempre seguen o mesmo formato: a letra C, un guión e o DNI (8 díxitos e unha letra maiúscula), por exemplo *C_53454123X.*

Tomando como base o documento XML anterior, empregue expresións XPath para obter os seguintes datos:

c) O nome dos clientes que reservaron algunha habitación para entrar en Agosto.

d) O identificador (id) dos clientes que teñan un nome que empece por F.

Tomando como base o documento XML anterior:

e) Xere como saída das transformacións XSLT un documento HTML que conteña unha táboa como a seguinte. Observe que as reservas están ordenadas por mes. O texto das cabeceiras debe estar centrado. O color de fondo das celas sombreadas é #ddd

Reserva de salones		
Fecha entrada	Cliente	Observaciones
15-Agosto-2021	C_44545123A	Servicio de catering
25-Octubre-2021	C_53454123X	

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
A COMPLETAR
</xsl:template>
</xsl:stylesheet>
```

EXERCICIO 3

EXERCICIO 3.1

Dada a seguinte relación:

idServicio	Servicio	idEmpl	nomEmpl	dirEmpl	cpDir	fecha	horas	codcliente	Impuesto	tipoCont
1	Laboral	1	Julián	Rúa San Pedro	15289	01/11/2021	70	112	15%	Por horas
1	Laboral	2	María	Plaza de Galicia	15003	01/11/2021	51	112	15%	Por horas
1	Laboral	1	Julián	Rúa San Pedro	15289	02/11/2021	83	112	15%	Por horas
2	Penal	2	María	Plaza de Galicia	15003	01/11/2021	34	234	21%	Comisión
2	Penal	4	Alexandra	Plaza España	15235	02/11/2021	82	234	21%	Comisión
3	Laboral	4	Alexandra	Plaza España	15642	03/11/2021	23	123	15%	Por horas

Tendo en conta que:

- Cada empregado/a poderá traballar en varios servizos e en cada servizo poden traballar varios empregados/as.
- Nunha data concreta un empregado/a só poderá ter un reporte de horas por cada servizo.
- O imposto dependerá do tipo de contrato.
- Cada cliente só poderá ter un tipo de contrato.

a) Indica as dependencias funcionais.

b) Que atributos forman parte do peche de ***idServicio***?

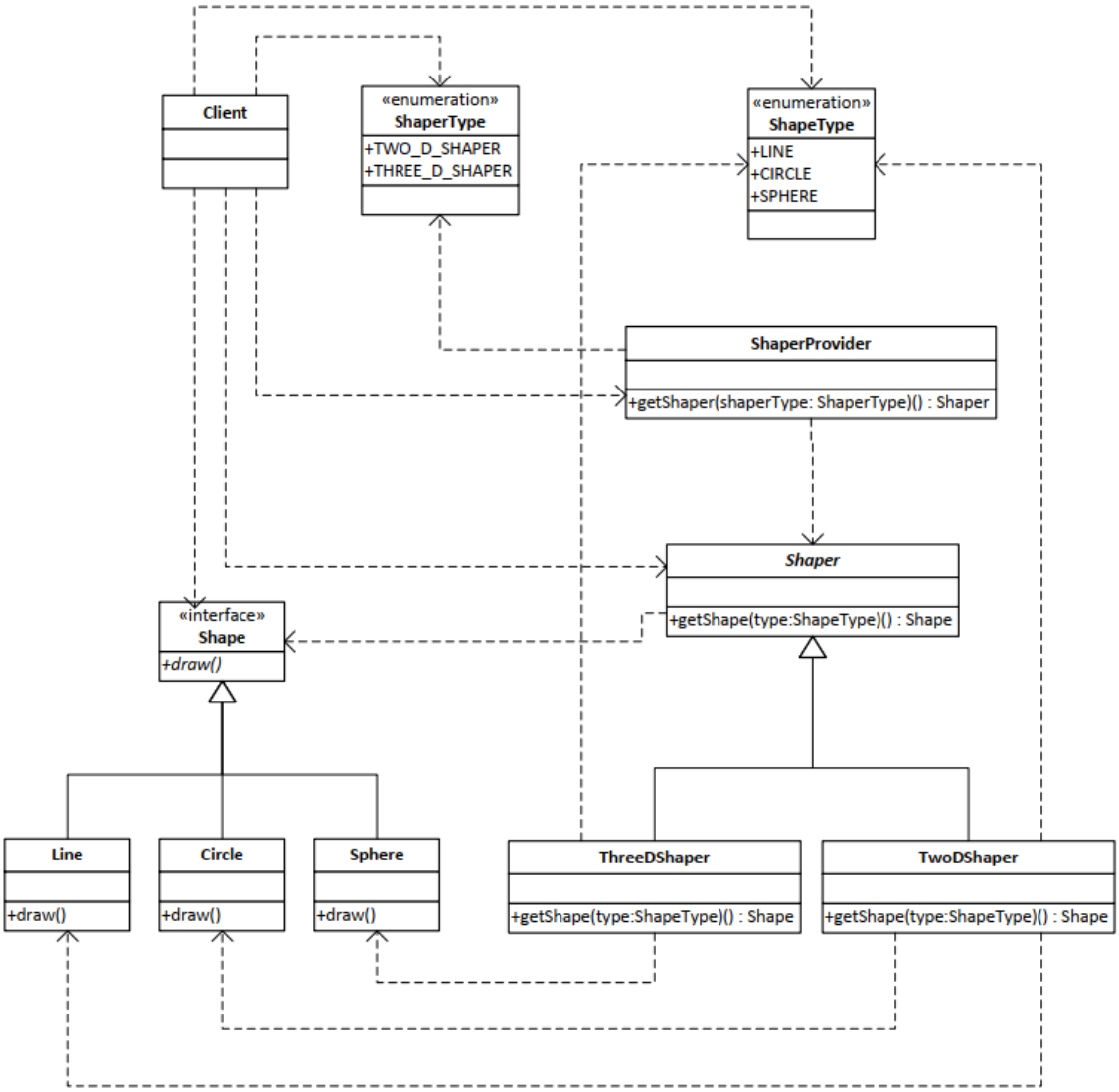
c) Indica que atributos forman parte da clave.

d) En que forma normal está a relación? Xustifique a resposta.

e) Normalícea ata 3FN se é posible.

EXERCICIO 3.2

Dado o seguinte diagrama de clases UML:



a) Identifique os patróns de deseño empregados.

b) Implemente en Java as clases ***TwoDShaper***, ***ShaperProvider***, ***Circle***.

EXERCICIO 3.3

Todos os exercicios débense facer sobre MariaDB/MySQL.

As consultas deben facerse de forma xenérica, non de forma específica para o contido concreto das táboas de exemplo.

A base de datos ***Hospital***, que se describe a continuación, contén información sobre médicos, habitacións e pacientes dun hospital.

A táboa ***Medicos*** contén o listado dos médicos que traballan no hospital.

- A columna ***Id*** é o identificador.

- A columna ***Nombre*** contén o nome e primeiro apelido dos médicos.
- Na columna ***Supervisor*** indícase o identificador do supervisor, que é outro médico. Se figura ***null*** quere dicir que o médico non ten supervisor.
- A columna ***Salario*** indica o salario anual do médico.
- A columna ***Promocion*** indica o ano no que se incorporou o médico ao hospital.
- A columna ***Especialidad*** indica a especialidade médica. O supervisor non ten por que ser da mesma especialidade que o médico ao que supervisa.

Id	Nombre	Supervisor	Salario	Promocion	Especialidad
1	Bibiana Diaz	6	46000	2012	Cardiologia
2	Angel Cervantes	6	44000	2000	Cardiologia
3	Brenda Palau	9	35600	2003	Dermatologia
4	Belen Romera	6	54000	2010	Cardiologia
5	Araceli Fraga	NULL	75800	1995	Geriatria
6	Mikel Zamora	NULL	84900	1996	Cardiologia
7	Humberto Ruiz	4	46900	2006	Geriatria
8	Joaquin Pineda	6	28900	2010	Cardiologia
9	Anselmo Santana	5	41600	2002	Dermatologia
10	Teodora Tello	3	41200	2002	Geriatria

A táboa ***Habitaciones*** almacena as habitacións do hospital.

- A columna ***Id*** é o identificador.
- A columna ***Tipo*** indica se a habitación é individual ou dobre.

Id	Tipo
1	Individual
2	Individual
3	Individual
4	Individual
5	Doble
6	Doble
7	Doble
8	Doble

A táboa ***Pacientes*** contén os pacientes.

- A columna ***Id*** é o identificador.
- A columna ***Nombre*** contén o nome e primeiro apelido dos pacientes.
- A columna ***IdMedico*** indica o identificador do médico que trata ao paciente.
- Na columna ***IdHabitacion*** indícase o identificador da habitación na que se aloxa o paciente.

Id	Nombre	IdMedico	IdHabitacion
1	Esther Godoy	9	5
2	Ines Urena	5	5
3	Raimundo Rosado	1	3
4	Antonia Carrero	10	7
5	Xoel Llamas	9	6
6	Esperanza de la Torre	2	4
7	Jose Fernandez	4	6
8	Manuel Oliveira	10	2
9	Emilia Aguilera	1	7
10	Luis Buendia	4	1

a) Na base de datos ***Hospital***, cree unha nova táboa, chamada ***Cardiologos***, que estea disponible unicamente na sesión actual, que conteña toda a información dos médicos da especialidade de ***Cardiologia***.

b) Sobre a base de datos ***Hospital***, obteña os nomes de tódolos médicos xunto cos nomes dos seus supervisores, ordeado alfabéticamente polos nomes dos médicos. Se un médico non ten supervisor, deberá aparecer ***null*** no nome deste. Utilice un alias de columna para indicar cal é a columna dos médicos e cal a dos supervisores

c) Sobre a base de datos ***Hospital***, obteña o nome dos médicos que non son supervisores de ningún outro.

d) Sobre a base de datos ***Hospital***, calcule o salario medio dos médicos en función do tipo de habitación na que se aloxan os seus pacientes. Considere unicamente aos médicos que teñan supervisor. Descarte do resultado os salarios medios inferiores a 45000 .

e) Sobre a base de datos ***Hospital***, seleccione todas as parellas posibles de médicos da mesma especialidade. Débense evitar as parellas triviales (o mesmo médico dúas veces) e as simétricas (mesma parella en distinto orden). O resultado da consulta podería ser este (sen importar a orde):

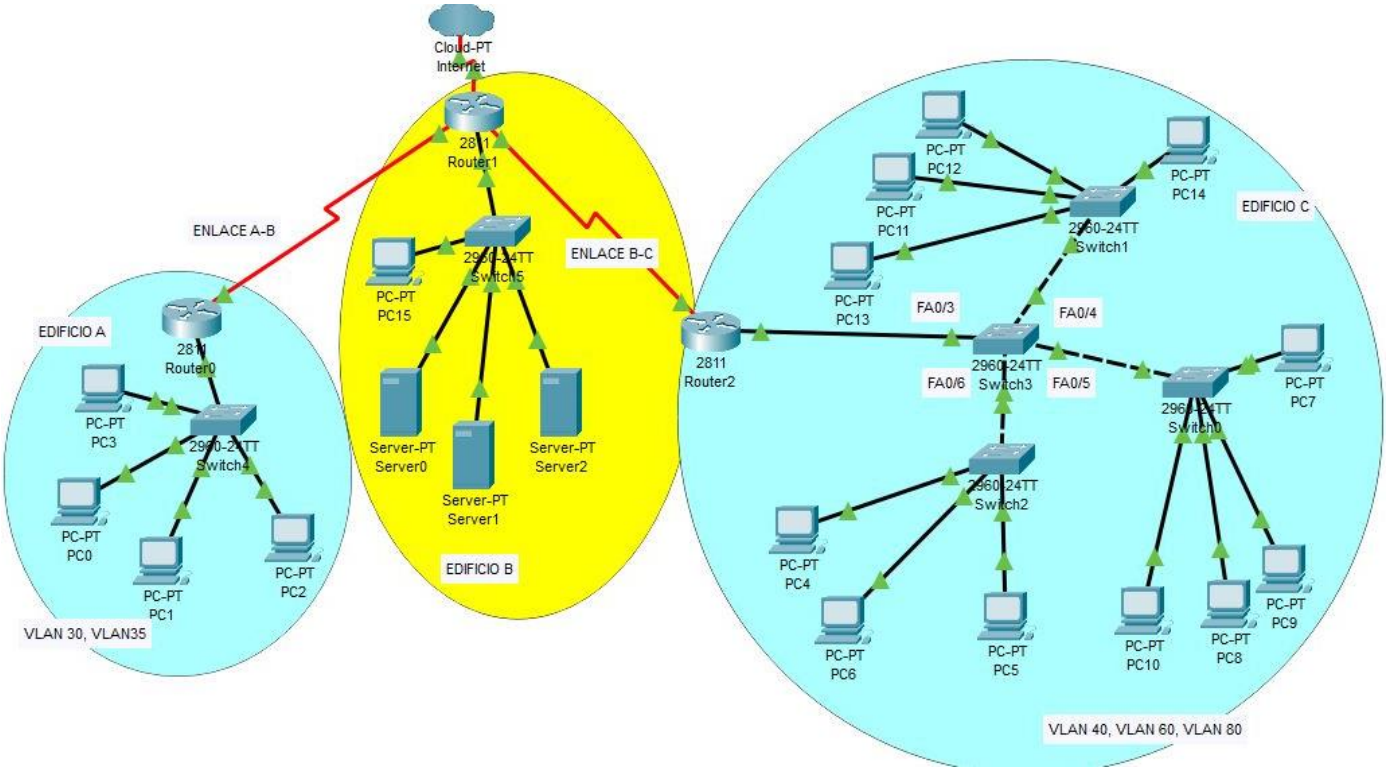
Medico1	Medico2	Especialidad
Angel Cervantes	Belen Romera	Cardiologia
Angel Cervantes	Joaquin Pineda	Cardiologia
Angel Cervantes	Mikel Zamora	Cardiologia
Belen Romera	Joaquin Pineda	Cardiologia
Belen Romera	Mikel Zamora	Cardiologia
Bibiana Diaz	Angel Cervantes	Cardiologia
Bibiana Diaz	Belen Romera	Cardiologia
Bibiana Diaz	Joaquin Pineda	Cardiologia
Bibiana Diaz	Mikel Zamora	Cardiologia
Mikel Zamora	Joaquin Pineda	Cardiologia
Brenda Palau	Anselmo Santana	Dermatologia
Araceli Fraga	Humberto Ruiz	Geriatria
Araceli Fraga	Teodora Tello	Geriatria
Humberto Ruiz	Teodora Tello	Geriatria

f) Sobre a base de datos ***Hospital***, obteña, para cada médico, o número de médicos que teñen maior antigüidade ca el. O médico de antigüidade máis alta debe figurar tamén no resultado, indicando que o número de médicos que teñen maior antigüidade é cero.

g) Cree un usuario chamado ***u1***, con contrasinal ***abc123.***, que teña soamente acceso local. Conceda ao usuario ***u1*** permiso de consulta e inserción sobre unha táboa ***t1*** (con columnas ***c1, c2*** y ***c3***), nunha base de datos chamada ***d1***, e permiso de actualización sobre as columnas ***c1*** e ***c2*** da mesma táboa.

EXERCICIO 4

A Consellería de Educación desexa realizar unha nova instalación nun centro educativo que dispón de 3 edificios. Plantéxase o seguinte esquema de rede:



O edificio A terá 2 VLANs (VLAN30 e VLAN35) con capacidade para 420 e 510 equipos. O edificio B terá un capacidade de 2020 equipos e o edificio C terá 3 VLANS (VLAN40, VLAN 60 e VLAN80) con capacidade para 130, 45 e 32 equipos respectivamente.

Os tres edificios disporán dun router. Por medio do router do edificio B conectaranse a Internet. A Consellería dispón, para todo o direccionamento dos tres edificios e os enlaces entre os routers, da rede IPv4 160.0.160.0/20.

a) Realice a división en subredes para o direccionamento de tódalas redes tendo en conta os seguintes criterios:

- O rango de direccións que queden libres despois de asignar os direccionamentos debe ser o maior posible.
- Débese axustar o tamaño da rede o máis pequeno posible para o número de equipos empregados.
- As portas de enlace precisan dunha IP.
- Os enlaces entre os routers soamente deben permitir dúas IPs e deben empregar o rango de direccións seguinte o da última subrede calculada dos edificios, comezando polo enlace A-B e logo polo B-C.
- Empregue unha táboa cos seguintes campos para resolver o exercicio:

Empregue comandos CLI sen abreviar para a realización dos apartados b, c e d:

Id Subrede	Dirección de rede	Prefixo	Máscara	Primeira IP asignable	Última IP asignable	Broadcast

b) Configure ssh version2 no ***Router1*** para un usuario que se chame ***usuario1*** con contrasinal ***exame*** e que non permita acceso por telnet. Supoña que o resto da configuración do ***Router1*** xa é correcta.

Router1>

c) Configure o ***Switch3*** para que os equipos das VLANs poidan chegar ao router e os equipos das distintas VLANs poidan verse entre eles, a excepción da VLAN 40. Supoña o resto do esquema configurado correctamente.

Switch3>

d) Configure unha IP para poder administrar o ***Switch5*** do Edificio B de forma remota desde calquera rede e habilite telnet no mesmo. Presupoña como porta de enlace a última IP asignable da subrede na que se atope o switch. No caso de que precise un contrasinal configure o contrasinal ***exame.*** De ser así, o contrasinal ten que gardarse cifrado.

Switch5>

e) No edificio B plantéxase instalar 3 servizos en cada un dos servidores. DHCPv6, NTP e RTSP. O administrador de rede quere coñecer os portos nos que van a traballar eses servizos no caso da súa implantación. Indique os números de portos por defecto de traballo e tipo de porto de cada un deles.

f) No edificio A óptase por instalar un servidor ***DHCPv4*** en Ubuntu na ***VLAN30***. Na ***VLAN35*** preténdese que as direccións IPs sexan adquiridas por medio dun axente de retransmisión en Ubuntu. Supoña para este exercicio que as VLAN están nas seguintes dirección de rede. VLAN30 (160.0.160.0/21) e VLAN35(160.0.168.0/21). O servidor DHCP atópase na IP 160.0.160.5. Indique:

   i. O comando para instalar ese axente de retransmisión.

   ii. Indique o nome do arquivo de configuración do mesmo.

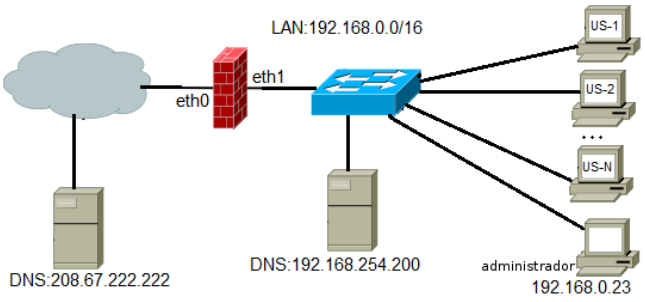
   iii. Cal é o parámetro principal deste arquivo e que valor debe tomar?

   iv. As liñas necesarias do arquivo de configuración ***dhcpd.conf*** para crear os ámbitos que permitan que os hosts de ambas VLANs se poidan conectar a Internet. A porta de enlace será a primeira IP asignable. O rango de direccións dispoñible de cada rede será dende a 100 á 200 do último byte da dirección de rede. O servidor DNS ao que se conectarán será o 8.8.8.8. Presupoña o resto do arquivo ***dhcpd.conf*** preconfigurado.

g) Decídese instalar un servidor DNS no edificio A na VLAN30 (suponse para dita VLAN, a dirección de rede 160.0.160.0/21) cunha zona de búsqueda directa que se chama edificioA.es. Supoña o arquivo de zona configurado correctamente e escriba as liñas necesarias de dito arquivo para crear unha delegación sobre unha zona chamada planta1.edificioA.es que se atopará no servidor con IP 160.0.168.40. Este servidor chamarase dns2.

EXERCICIO 5

Nunha organización vaise a instalar un ***Firewall*** con ***Iptables*** sobre un servidor Debian 10. Este ***Firewall*** estará instalado entre a LAN e Internet, tal como se amosa na imaxe.



Inicialmente, o ***Firewall*** ten a seguinte configuración:

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
```

Configure este ***Firewall*** para que cumpra coas funcionalidades que se piden a continuación (*Nota: débese controlar o estado das conexións*):

a) Os ordenadores da LAN deben poder comunicarse cos servidores WEB en Internet.

b) O servidor DNS que está na LAN deberá poder conectarse, a través do ***Firewall***, ao seu DNS ***forwarder*** que está en Internet, coa IP: *208.67.222.222.*

c) Realice a configuración necesaria para que o ordenador do administrador na LAN (192.168.0.23) poida establecer comunicación co servizo SSH instalado no ***Firewall***.

d) Débense eliminar tódolos paquetes de información que entran pola tarxeta eth0 (dende Internet) e cuxos bits SYN e FIN da cabeceira TCP sexan igual a 1 e o resto de bits iguais a 0.

e) Débese rexistrar (facer un ***log***) cando algún ordenador da LAN, distinto do ordenador do administrador, intente conectarse ao porto ssh (22) do ***Firewall***. Débese engadir o texto "SSH-Proh: " ao rexistro. (*Nota: o servidor de logs é o que está instalado por defecto en Debian 10: syslog*).

f) Non se permitirá máis dun paquete ***icmp echo-request*** (***ping***) por segundo dende a LAN cara o ***Firewall*** e ningún ***ping*** desde Internet cara o ***Firewall***.

OPCIÓN B

EXERCICIO 1

Deséxase realizar a tódolos membros do corpo docente da Consellería de Cultura, Educación e Universidade, unha enquisa onde se puntúen e valoren algúns dos pilares básicos do sistema educativo. Para iso, e para axilizar todo o proceso, elaborarase un sitio Web. O primeiro paso será desenvolver, a partir da maquetación dunha páxina en HTML, unha aplicación con scripts PHP que permita realizar dita enquisa. O sitio Web conterá os seguintes campos:

- DNI
- Nome completo
- Ciclo no que se imparte docencia
- Varios campos de selección de resposta única ás preguntas plantexadas

```
<html>
    <head>
        <title>Enquisa de opinión</title>
        <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
    </head>
    <body>
            <form action="<?php echo $_SERVER['PHP_SELF']; ?>" method="post">
                <fieldset>
                    <h1>Datos persoais</h1>
                    <h3>Introduza o seu DNI</h3>
                    <input type="text" name="dni">

                    <h3>Introduza o seu nome completo</h3>
                    <input type="text" name="nome">

                    <h3>Seleccione o ciclo no que imparte docencia</h3>
                    <select name="ciclo" value="ciclo">
                        <option>SMR</option>
                        <option>ASIR</option>
                        <option>DAM</option>
                        <option>DAW</option>
                    </select>

                    <h1>Enquisa</h1>
                    <h3>Qué opinas da LOMLOE?</h3>

                    <h3>Qué opinas acerca da supresión dos itinerarios curriculares e a
obtención do título único da ESO?</h3>
                    <input type="radio" name="reply1" value="3">
                    3. Estou de acordo.<br>
                    <input type="radio" name="reply1" value="2">
                    2. Indiferente.<br>
                    <input type="radio" name="reply1" value="1">
                    1. En desacordo.
                    <br> <br>
```

```
                    <h3>Qué opinas acerca da flexibilización da evaluación do
alumnado?</h3>
                    <input type="radio" name="reply2" value="3">
                    3. Estou de acordo.<br>
                    <input type="radio" name="reply2" value="2">
                    2. Indiferente.<br>
                    <input type="radio" name="reply2" value="1">
                    1. En desacordo.
                    <br> <br>

                    <h3>Qué opinas acerca da incorporación da educación dixital?</h3>
                    <input type="radio" name="reply3" value="3">
                    3. Estou de acordo.<br>
                    <input type="radio" name="reply3" value="2">
                    2. Indiferente.<br>
                    <input type="radio" name="reply3" value="1">
                    1. En desacordo.
                    <br> <br>

                    <h3>Cal é a túa opinión de forma xenérica acerca da LOMLOE?</h3>
                    <input type="radio" name="reply4" value="3">
                    3. Estou de acordo.<br>
                    <input type="radio" name="reply4" value="2">
                    2. Indiferente.<br>
                    <input type="radio" name="reply4" value="1">
                    1. En desacordo.
                    <br> <br>

                    <input name="submit" type="submit" value="Enviar resultados">
            </fieldset>
        </form>
    </body>
</html>
```

Á hora de desenvolver a lóxica do sitio Web, deberanse ter en conta as seguintes consideracións e implementar as correspondentes comprobacións e cálculos:

a) O DNI do usuario que realiza a enquisa debe estar rexistrado na base de datos ***empleados*** da Consellería, para asegurar que a persoa que realiza a enquisa é realmente un docente. Dita base de datos atópase aloxada no porto 3306 do servidor ***edu.xunta.gal***, cuxo usuario administrador é ***useradmin*** e o contrasinal ***bdadministration***. En caso de que o usuario non se atope dado de alta na base de datos como docente, a enquisa non será enviada e a aplicación mostrará unha mensaxe de erro.

b) A partir de tódalas puntuacións asignadas a cada unha das cuestións, deberase calcular a súa media empregando as funcións de conteo e suma propias de PHP.

c) A continuación, deberase redondear dita media de puntuacións empregando unha función propia de PHP destinada para iso, que devolva un número con só 2 cifras decimais e onde se redondee á alza cada cifra decimal superior a 5.

d) Almacenaranse os datos introducidos, e tamén a media redondeada das respostas, nun ficheiro de texto chamado ***resultados.txt***, ubicado na carpeta raíz da aplicación, que deberá ser creado en caso de que non exista, e que será ampliable según se vaian realizando máis participacións en dita enquisa.

e) Para evitar rexistros redundantes, deberase comprobar, mediante o uso de ***cookies***, que a enquisa non foi cuberta xa polo usuario. No caso de que xa fose enviada, non se mostrarán os campos a cubrir, senón unha mensaxe que advirta das circunstancias.

EXERCICIO 2

Implemente un programa en ***Java*** que xere un novo ficheiro XML a partires dun ficheiro de texto (.txt).

O ficheiro de texto contén o DNI e un ou máis correos electrónicos de opositores.

Cada liña do ficheiro contén os datos dun único opositor.

Os datos veñen nesa orde sempre separados por coma (,). O DNI non pode aparecer repetido pero o correo electrónico sí, é dicir, pode haber varios emails nunha mesma liña por opositor.

Tanto o DNI como os correos poden ou non vir entre comiñas dobres.

Exemplo de ficheiro de entrada:

```
"36223344a", marcosantonio@gmail.com, MARCOSLOU@EDU.XUNTA.GAL
44112233B, MARTAPENA@edu.xunta.es, "martapena@edu.xunta.es"
```

Os datos deben ser pasados a un ficheiro XML na mesma orde na que veñen.

Exemplo de ficheiro de saída:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<opositores>
    <opositor>
        <dni>36223344A</dni>
        <correo>marcosantonio@gmail.com</correo>
        <correo>marcoslou@edu.xunta.gal</correo>
    </opositor>
    <opositor>
        <dni>44112233B</dni>
        <correo>martapena@edu.xunta.es</correo>
    </opositor>
</opositores>
```

Comprobe, mediante expresións regulares, que os datos son correctos:

- O DNI ten 7 ou 8 números e unha letra maiúscula ou minúscula (sen guión entre os números e a letra), coa letra inmediatamente a continuación dos números. No ficheiro XML, a letra do DNI debe aparecer en maiúscula.
- Os correos electrónicos poden ter letras, puntos (.), números e guións baixos (_), pero un correo só pode comezar por letra, número ou guión baixo (_) continuando por letras, puntos (.), números ou guións baixos (_) ata a arroba (@). Despois da arroba (@) só pode haber letras e números (alomenos un carácter) e logo letras, números, guións baixos (_) e puntos (.), ata un punto seguido de 2 ou máis letras (o dominio).

- Os correos deben ser gardados no ficheiro XML completamente en minúsculas. Os correos poden vir duplicados no ficheiro de texto pero no XML non pode haber correos repetidos para o mesmo opositor (non se debe ter en conta maiúsculas e minúsculas, é dicir, o carácter 'a' é igual ó carácter 'A').
- No caso de que o ficheiro de texto teña algún erro nalgún dos datos, o programa debe abortar indicando o lugar e causa do erro, lanzando unha nova excepción dunha nova clase ***OpositorException***.

A clase principal do programa é a seguinte:

```
package com.opos2021.exameopos2021;
public class XerarFicheiroXML {
     private static final String TXT_FILE = "/home/opositores.txt";
  private static final String XML_FILE = "/home/xml_dom_opositores.xml";
  public static void main(String[] args)
  {
   try
   {
         XmlFileGenerator xmlFG = new XmlFileGenerator();
         xmlFG.createXMLFile(TXT_FILE, XML_FILE);
         System.out.println("Ficheiro xerado correctamente: " + XML_FILE);
   }
   catch (Exception e)
   {
              System.out.println("ERROR Fatal: " + e.getMessage());
   }
    }
}
```

Considere que existen dous métodos implementados y funcionais na clase ***XmlFileGenerator*** que poden ser empregados sen necesidade de ser desenvolvidos:

- ***openFile***: recibe unha cadea de texto coa ruta ao ficheiro de texto e ábreo.
- ***closeFile***: pecha o ficheiro de texto.

a) Cree a clase ***Opositor.java*** e ***OpositorException.java***. No primeiro caso, a clase contén os datos de cada opositor e os métodos ***getter*** e ***setter*** das propiedades da clase. A clase ***OpositorException*** debe extender a clase ***Exception*** e amosar o texto "Excepción opositor: " e despois a mensaxe da propia excepción da clase ***Exception***.

b) Desenvolva un método da clase ***XmlFileGenerator*** chamado ***splitTokensFromString*** que, dada unha liña do ficheiro, devolva unha lista de ***string*** cun dato en cada posición da mesma. Indique a firma completa do método, incluído o modificador de acceso, os parámetros recibidos, o valor devolto cos seus tipos correspondentes e, de ser o caso, a excepción que pode lanzar.

c) Desenvolva un método da clase ***XmlFileGenerator*** chamado ***createOpositorXmlNode*** que reciba unha lista de ***string*** cos datos dun opositor e devolva un obxecto de tipo Opositor cos datos recibidos na lista. Comprobe que os datos sexan correctos. Indique a firma completa do método, incluído o modificador de acceso, os parámetros recibidos, o valor devolto cos seus tipos correspondentes e, de ser o caso, a excepción que pode lanzar.

d) Complete o método ***createXMLFile*** desenvolvendo o código que sería necesario incluír nos subapartados seguintes (marcados coa indicación ***A DESENVOLVER***. Teña en conta que o

comentario anterior indica o que hai que facer). Presupoña tódolos ***import*** necesarios para a resolución do apartado, sen necesidade de engadilos á resposta.

A declaración da propiedade empregada para ler o ficheiro de texto é a seguinte: ***private Scanner lectorFicheiro = null;***

```
import java.io.*;;
import java.util.*;
import javax.xml.parsers.*;
import javax.xml.transform.*;
import org.w3c.dom.*;

public void createXMLFile(String txtFile, String xmlFile) throws OpositorException
{
  try
        {
      // Ler o ficheiro de texto e gardar os datos nunha lista de obxectos de //
tipo Opositor:
      A DESENVOLVER APARTADO D[...]
       // Create DOM:
       DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
       DocumentBuilder builder = factory.newDocumentBuilder();
       DOMImplementation implementation = builder.getDOMImplementation();
       // Create document (root of the XML file) and set version:
       Document doc = implementation.createDocument(null, "opositores", null);
       doc.setXmlVersion("1.0");
       // Get the root element:
       Element root = doc.getDocumentElement();
       // Recorrer a lista de opositores e ir xerando a estrutura XML
       // que logo se almacenará no ficheiro:
       A DESENVOLVER APARTADO D[...]
       // Write to file:
       Source source = new DOMSource(doc);
       Result result = new StreamResult(new java.io.File(xmlFile));
       Transformer trans = TransformerFactory.newInstance().newTransformer();
       trans.setOutputProperty(OutputKeys.INDENT, "yes");
       trans.setOutputProperty("{http://xml.apache.org/xslt}indent-amount", "4");
            trans.setOutputProperty(OutputKeys.ENCODING, "UTF-8");
            trans.transform(source, result);
        }
        catch (Exception e)
        {
      A DESENVOLVER APARTADO D[…]
        }
          }
```

EXERCICIO 3

EXERCICIO 3.1

Dada a seguinte relación:

IdLibro	tituloLibro	autor	genero	codCopia	Estado	Cod Socio	NomSocio	fechaPrestamo	fechaDevol
1	La sombra del viento	Carlos Ruíz Zafón	Novela	CP_1	Correcto	1	Julia	01/08/2021	01/09/2021
1	La sombra de viento	Carlos Ruíz Zafón	Novela	CP_4	Deficiente	2	Alejandro	01/08/2021	05/08/2021
2	Aquitania	Eva Saenz	Histórica	CP_13	Segunda mano	1	Julia	05/02/2021	05/08/2021
2	Aquitania	Eva Saenz	Histórica	CP_12	Segunda mano	1	Julia	05/01/2021	05/07/2021
3	Ulises	James Joices	Ficción	CP_1	Deficiente	3	Carmen	23/09/2021	15/10/2021

Tendo en conta que:

- Un libro só pode ter un determinado xénero.
- O estado dun libro depende da copia do libro que se prestase.
- A data de devolución dun libro, calcularase a partir da data de préstamo, do socio e da copia do libro que se prestase.
- Un socio pode alquilar varias veces un mesmo libro, pero non na mesma data.

a) Indique as dependencias funcionais.
b) Que atributos forman parte do peche de ***idLibro***?
c) Indique que atributos forman parte da clave.
d) En que forma normal está a relación?
e) Normalícea ata 3FN se é posible.

EXERCICIO 3.2

Dada a base de datos ***meteo*** con información meteorolóxica existente nun servidor MySQL coas seguintes táboas:

provincia *(cada fila representa unha provincia)*

columna	tipo	null	clave	observacións
`id`	`int`	N	P	Id da provincia
`nome`	`varchar(30)`	N		Nome da provincia

concello *(cada fila representa un concello)*

columna	tipo	null	clave	observacións
`id`	`int`	N	P	Id autoincremental do concello
`nome`	`varchar(30)`	N		Nome do concello
`id_provincia`	`int`	N	FK	Id da provincia á que pertence o concello

estacion *(cada fila representa unha estación meteorolóxica)*

columna	tipo	null	clave	observacións
`id`	`int`	N	P	Id autoincremental da estación
`nome`	`varchar(30)`	N		Nome da estación
`latitude`	`double`	S		Latitude á que se atopa a estación
`lonxitude`	`double`	S		Lonxitude á que se atopa a estación
`cota`	`float`	S		Cota sobre o nivel do mar da estación
`id_concello`	`id`	N	FK	Id do concello onde se sitúa a estación.

variable *(unha fila por cada da variable meteorolóxica da que se toman medidas)*

columna	tipo	null	clave	observacións
`id`	`int`	N	P	Id autoincremental da variable
`nome`	`varchar(30)`	N		Nome da variable
`unidade`	`varchar(10)`	S		Unidade na que se mide a variable

medida *(unha fila por cada valor medido dunha variable nunha estación e data determinadas)*

columna	tipo	null	clave	observacións
`id`	`int`	N	P	Id autoincremental da medida
`valor`	`double`	N		Valor rexistrado da medida
`data`	`date`	N		Data na que se tomou a medida
`id_variable`	`int`	N	FK	Id da variable á que corresponde a medida
`id_estacion`	`int`	N	FK	Id da estación na que se tomou a medida
Existe un índice único sobre as columnas *`id_variable, id_estacion, data`*				

Escriba as instrucións SQL necesarias para responder ás seguintes consultas:

a) Obteña para cada estación cantas estacións hai a unha cota maior que a súa na mesma provincia. Incluír a estación situada a maior altura de cada provincia, en cuxo caso o número de estacións a maior altura debe ser 0. Mostrar o id e o nome da provincia, e o id e o nome da estación.

b) Obteña as parellas de estacións que rexistraron, no mesmo día, o mesmo valor nalgunha variable. Mostrar o nome de cada estación, o nome da variable na que coincidiron, o valor coincidente e a data na que isto sucedeu.

   Débense evitar as parellas triviais, formadas por coincidencia dos valores dunha estación cos seus propios valores e as parellas simétricas, formadas pola repetición noutra orde dunha parella xa amosada.

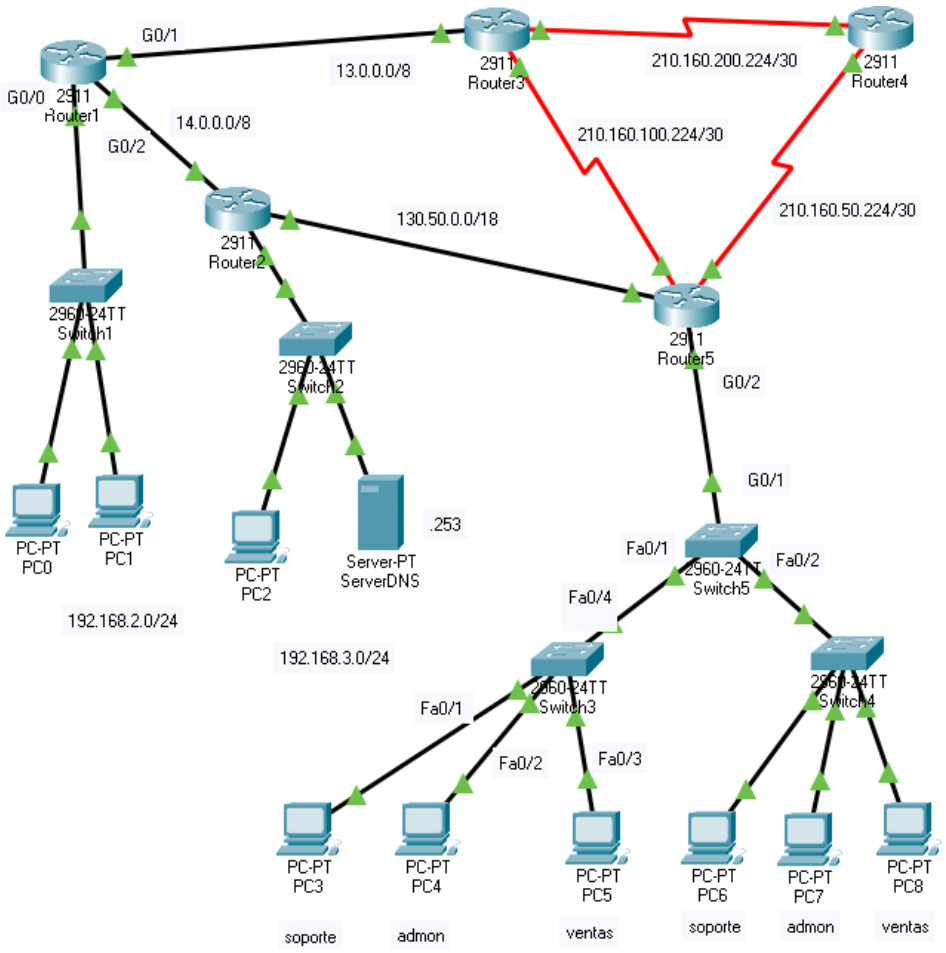
c) Para os municipios nos que hai estacións, amosar o id da estación, o id do municipio e cantas estacións hai en cada un deles, amosando tamén o total de estacións de cada provincia e o total de estacións de tódalas provincias.

d) Obteña as estacións que tiveron unha media das temperaturas mínimas diarias durante o mes de xaneiro de 2020 maior que 2. A temperatura mínima diaria é a variable que ten por nome 'Temperatura mínima'.

EXERCICIO 4

EXERCICIO 4.1

Dada a topoloxía da seguinte imaxe, na que o routing vaise facer con OSPF, responda ás seguintes cuestións indicando tódolos comandos CLI completos (sen abreviar) necesarios dende que arranca cada un dos dispositivos implicados para poder realizar a configuración que se pide en cada caso.



a) Cubra a seguinte táboa de direccións según a topoloxía e empregue esas interfaces para a configuración nos exercicios, asumindo que para direccionar os routers emprégase a primeira das direccións dispoñibles da rede e sempre numeramos de esquerda a dereita (e de arriba a abaixo).

TÁBOA DE TOPOLOXÍA

Dispositivo	Interface	Dirección IP	Máscara de subrede
R1	G0/0	192.	
	G0/1	13.	
	G0/2	14.	
R2	G0/0	14.	
	G0/1	192.	
	G0/2	130.	
R3	G0/0	13.0.0.2	
	S0/3/0	210.160.200.	
	S0/3/1	210.160.100.	
R4	S0/3/0	210.160.200.	
	S0/3/1	210.160.50.	
R5	G0/1	130.	
	S0/3/0	210.160.100.	
	S0/3/1	210.160.50.	

b) Configure o ***Router1*** para que funcione como servidor DHCP das redes 192.168.2.0/24 á que chamaremos LAN1 e 192.168.3.0/24 a que chamaremos LAN2. Teña en conta a dirección do servidor DNS indicada na topoloxía (só o será para estas dúas redes e non hai que configurar ese servizo). Indique a configuración de tódolos dispositivos implicados para conseguir esa configuración DHCP, incluíndo as interfaces implicadas.

c) Configure as VLAN presentadas na topoloxía de xeito que **soporte** e **ventas** ademais poidan comunicarse co resto da rede pero **admon** so cos da súa vlan.
As direccións para as distintas redes VLAN:

Nome	Número	Rede
soporte	10	192.168.4.0/24
admon	20	192.168.5.0/24
ventas	30	192.168.6.0/24

No caso da configuración do ***Switch3*** e o ***Switch4*** configure unicamente a do ***Switch3***. Non é necesario neste caso crear interfaces de administración nos switch.

d) Configure os routers con OSPFv2 de área única, tendo en conta as seguintes condicións:
- Configure explícitamente o id do router para a súa participación en OSPF. Empregarase o mesmo número que o número de router (1,2,...)
- Configure as redes para o enrutamento OSPF mediante comandos de rede. Non se admiten máscaras de cuádruple 0.
- So anuncie OSPF a través das interfaces conectadas a outros routers pero conseguindo que haxa conectividade entre tódalas redes.
- Configure as interfaces dos routers que non estivesen direccionadas.

Realice unicamente a configuración dos seguintes routers: ***Router1***, ***Router3***, ***Router5***

e) Substitúa o ***Switch5*** e o ***Router5*** por un único dispositivo que permita realizar as mesmas funcionalidades dos apartados c) e d), e indique a súa configuración completa.

f) No caso do ***Switch3*** restrinxa as direccións MAC que se poden conectar na ***interface Fa0/1*** realizando as seguintes tarefas, habilitando previamente todo o necesario:
- Indique explicitamente que a cantidade máxima de direccións permitidas para acceder será 1.
- Para non ter que especificar manualmente a dirección MAC permitida habilite a autodetección de MAC (aprendizaxe) e logo gárdea no arquivo de configuración.
- Se hai un intento de conexión á interface por unha dirección non permitida, enviarase unha notificación SNMP ao administrador e descartarase ese tráfico.

EXERCICIO 4.2

Para o direccionamiento IPv6, foi asignada a dirección de rede 2001:1DB8:CAFE::/48. Decídese asignar identificadores de rede correlativos empregando o bit máis significativo do hexteto identificador da subrede, comezando por 0 para a rede A (0000 para a rede A, 1000 para a rede B, 2000 para a rede C…). Complete a seguinte táboa:

Subrede	IPv6 rede/prefixo	Rango de direccións
C		

D		
H		

EXERCICIO 5

EXERCICIO 5.1

Nun servidor Debian 10 instálanse 7 discos de 500GB de capacidade cada un (aparte dos que use o sistema). Estos discos son: ***/dev/sdb, /dev/sdc, /dev/sdd, /dev/sde, /dev/sdf, /dev/sdg*** e ***/dev/sdh*** e preténdese facer un RAID 6 cun disco en spare empregando estes 7 discos.

Conteste as seguintes preguntas:

a) Escriba a instrución en Linux para crear este RAID con ***mdadm***

b) Que capacidade terá para datos o RAID resultante?

c) Cantos discos poderían deixar de funcionar simultaneamente, sen tempo de recuperación entre erros?

d) Cantos discos poderían deixar de funcionar, dando tempo para a recuperación entre os erros e o RAID seguiría funcionando?

EXERCICIO 5.2

Unha empresa desexa xestionar as conexións a Internet a través dun servidor proxy con ***Squid***. Dita empresa conta cos seguintes departamentos, cada un cun rango de direccións IP da rede interna:

- O departamento de produción, rango de IPs 192.168.100.10-150.
- O departamento de administración, rango de IPs 192.168.100.200-210.

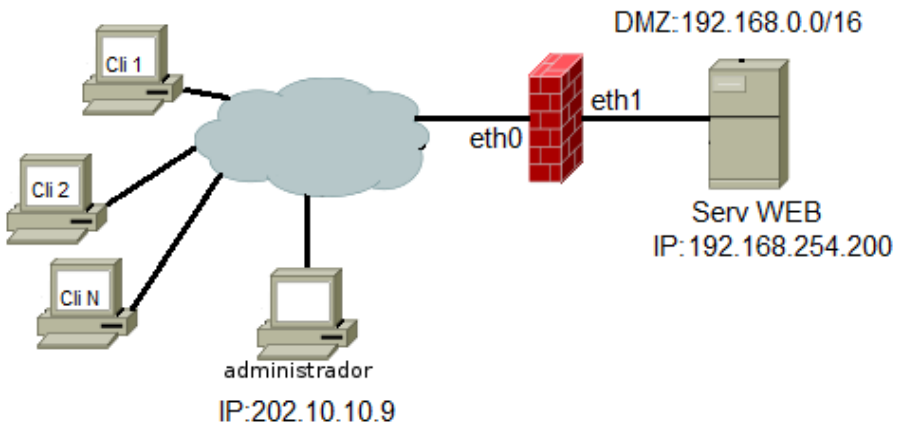
Xestionaranse as conexións ao exterior en base ás seguintes premisas:

- O departamento de produción ten acceso á rede externa de luns a venres, dende as 8:00h ata as 17:00h pero ten restrinxida a descarga dos tipos de arquivos definidos no arquivo ***/etc/squid/extensiones***.
- Fora do horario pedirase a autenticación de usuarios e non se permitirá o acceso ás webs definidas no arquivo ***/etc/squid/webprohibidas***.
- O departamento de administración sempre ten autenticación de usuarios, non ten restrición de horarios pero ten restrinxido o acceso ás webs definidas no arquivo ***/etc/squid/webprohibidas***.

Cree as ACL necesarias para o enunciado, así como as regras de acceso para as premisas anteriores no arquivo de configuración ***squid.conf***. Non é necesario especificar o programa de autenticación, suponse que xa está definido por defecto.

EXERCICIO 5.3

Nunha organización van instalar un ***Firewall*** con ***Iptables*** sobre un servidor Debian 10. Este ***Firewall*** estará instalado entre unha rede DMZ e Internet, tal como se amosa na seguinte imaxe.



Configure este ***Firewall*** para que cumpra coas funcionalidades que se piden a continuación: (*Nota: débese controlar o estado das conexións*):

a) Débese permitir que o administrador se poida conectar dende a súa casa (IP: 202.10.10.9) ao servizo SSH (porto 22) do ***Firewall*** e o servizo SSH do servidor WEB que está na DMZ.

b) Os clientes de Internet poderanse conectar ao servidor WEB.

c) Débese pór un límite de 25 conexións por segundo aos portos 80 e 443 dende Internet.

d) Rexistrarase (facer log) calquera intento de conexión que non estea permitido e débese engadir o texto "ConxProh:" ao rexistro. (*Nota: o servidor de logs é o que está instalado por defecto en Debian 10: syslog*)

e) Débense eliminar tódolos paquetes de información que chegan pola tarxeta eth0 (dende Internet) e que teñan tódolos bits da cabeceira TCP igual a 0.