

1. Describa qué es una dirección IP, cuál es su estructura y qué rol juega a la hora de comunicar dispositivos electrónicos dentro de una red.

Una IP es una etiqueta numérica que identifica a un dispositivo en la red informática, se usa para identificar y comunicarse entre otros dispositivos y/o servidores en red.

Existen dos tipos de IP (IPv4(32 bits) e IPv6(128 bits)) la IPv4 es la más comúnmente usada, va desde el "0.0.0.0" hasta el "255.255.255.255" aunque algunas de estas direcciones de están reservadas a usos específicos.

Por otro lado la IPv6 se divide en dos partes, los primeros 64 bits son de componente en la red, son para el enrutamiento y los últimos 64 bits son los componentes del nodo, que se utilizan para identificar la dirección de la interfaz.

La IP es única en la red, con la IP se puede saber donde estás para facilitar el paso de información deseada.

2. Describa detalladamente cuáles son y en qué consisten las distintas etapas de construcción de software (programa) en C++, desde los archivos de código fuente hasta obtener un archivo ejecutable. Ponga un ejemplo ilustrativo de cada una de estas etapas.

Escritura del código fuente: los desarrolladores escriben el código fuente del programa en C++. El código fuente es un conjunto de instrucciones que se utilizan para definir el comportamiento del programa.

```
#include <iostream>

int main() {
    std::cout << "Hola, mundo!" << std::endl;
    return 0;
}
```

Procesado: sustituir las librerías por su código asociado a estas.

Compilado: convierte el código fuente en código ensamblador (.s)

Ensamblado: el código fuente se procesa por un compilador que lo traduce a código máquina, dependiendo del compilador y de las opciones de configuración. (genera un código objeto .o u .obj)

```
g++ -c MiPrograma.cc -> MiPrograma.o
```

Enlazado: Enlaza todos los códigos objetos generados en un único archivo binario al que se le llama ejecutable.

```
g++ -o MiPrograma MiPrograma.o
```

```
./MiPrograma
```

5. Responda Verdadero (V) o Falso (F) o elegir la/s respuesta/s correcta/s para cada una de las siguientes preguntas teniendo en cuenta las siguientes observaciones:

- Por cada respuesta correcta se sumará 0.2 puntos.
- Por cada respuesta incorrecta se restará 0.2 puntos.
- Las preguntas sin responder se considerarán incorrectas.
- La puntuación mínima de la pregunta es de 0 puntos.

1.- El Gestor de Red es un componente habitual en un Sistema Operativo. **Verdadero**

2.- En la shell de Linux el directorio "home" del usuario se representa mediante el carácter / **Falso**

3.- De acuerdo a la Guía de Estilo de código que se sigue en la asignatura, los identificadores de funciones ordinarias (no métodos de una clase) deben escribirse comenzando con letra minúscula y usando mayúscula para cada palabra. **Falso**

4.- En una relación de composición de objetos la parte (miembro) sólo puede pertenecer a un objeto (clase) a la vez. **Verdadero**

5.- El método `emplace_back()` de la clase `std::vector` se utiliza para eliminar el último elemento de un vector. **Falso**

6.- Al leer caracteres desde un flujo de entrada (`cin`) la lectura se detiene cuando se detecta un espacio en blanco o bien un retorno de carro (enter). **Verdadero**

7.- Una función declarada "friend" de una clase puede acceder para escritura los miembros de datos (atributos) de esa clase. **Verdadero**

8.- Los operadores unarios debieran sobrecargarse usando métodos de la clase. **Falso**

9.- El espacio de direcciones que permite IPv4 es mayor que el que utiliza IPv6. **Falso**

10.- El lenguaje SQL se utiliza para realizar operaciones CRUD en bases de datos no relacionales. **Verdadero**

1. Describa las arquitecturas de red que conoce, características de cada una así como sus ventajas e inconvenientes.

Redes PAN (Red de Área Personal):

Las redes de Área Personal (PAN) están diseñadas para la comunicación entre dispositivos personales a distancias cortas, generalmente dentro de un rango de pocos metros. Tecnologías como Bluetooth y USB se utilizan para la conexión inalámbrica, facilitando la interacción entre dispositivos como teléfonos móviles, computadoras portátiles y periféricos. Aunque su tasa de transferencia de datos es baja en comparación con redes de mayor alcance, las PAN ofrecen ventajas como un consumo de energía relativamente bajo, implementación simple y costos bajos. Sin embargo, su principal inconveniente radica en su alcance limitado, siendo inadecuadas para la interconexión de dispositivos a larga distancia.

Ventajas: Ideal para conectar dispositivos personales cercanos. Consumo de energía relativamente bajo. Implementación simple y costos bajos.	Inconvenientes: Alcance limitado. Baja velocidad de transferencia de datos en comparación con redes de mayor alcance. No es adecuada para la interconexión de dispositivos a larga distancia.
--	---

Redes LAN (Red de Área Local):

Las redes de Área Local (LAN) tienen un alcance limitado, generalmente dentro de un edificio o campus, conectando dispositivos como computadoras, impresoras y servidores en un área geográfica cercana. Utilizan tecnologías como Ethernet y Wi-Fi, ofreciendo una mayor velocidad y capacidad de transferencia de datos en comparación con las PAN. Aunque facilitan la comunicación y compartición de recursos en un área local, presentan desafíos en configuración y mantenimiento en redes más grandes.

Ventajas: Alta velocidad de transferencia de datos. Facilita la comunicación y compartición de recursos en un área local. Costos relativamente bajos para implementaciones en pequeña escala.	Inconvenientes: Alcance limitado a una ubicación geográfica específica. Configuración y mantenimiento pueden ser más complejos en redes más grandes.
---	---

Redes WLAN (Red de Área Local Inalámbrica):

Similar a las LAN, las redes de Área Local Inalámbrica (WLAN) prescinden de cables físicos mediante el uso de tecnologías como Wi-Fi. Esto permite la movilidad de dispositivos dentro del área de cobertura, siendo ampliamente utilizadas en entornos empresariales, hogares y lugares públicos. Aunque ofrecen mayor flexibilidad y movilidad, pueden experimentar interferencia de otras redes inalámbricas y presentan menor velocidad y ancho de banda en comparación con algunas redes cableadas.

Ventajas: Mayor flexibilidad y movilidad debido a la conexión inalámbrica. Fácil expansión y acceso para dispositivos móviles. Conveniente para entornos donde el tendido de cables es difícil.	Inconvenientes: Posible interferencia de otras redes inalámbricas. Menor velocidad y ancho de banda en comparación con algunas redes cableadas.
---	--

Redes MAN (Red de Área Metropolitana):

Las redes de Área Metropolitana (MAN) tienen un alcance geográfico mayor que las LAN, conectando redes en una ciudad o área metropolitana. Utilizan tecnologías como fibra óptica y enlaces de microondas para proporcionar conectividad de alta velocidad en un área geográfica más extensa. Aunque son adecuadas para la interconexión de sucursales empresariales en una ciudad, los costos de implementación y mantenimiento pueden ser significativos, y la velocidad y capacidad son menores en comparación con las WAN.

Ventajas: Proporciona conectividad de alta velocidad en un área geográfica más extensa. Adecuada para la interconexión de sucursales empresariales en una ciudad.	Inconvenientes: Costos de implementación y mantenimiento pueden ser significativos. Menor velocidad y capacidad en comparación con las WAN.
--	--

Redes WAN (Red de Área Amplia):

Las redes de Área Amplia (WAN) tienen un alcance geográfico mayor que otras redes, pudiendo cubrir un país o incluso regiones globales. Utilizan tecnologías como conexiones de fibra óptica, satélites y líneas de transmisión de datos para interconectar LAN y MAN distantes. Aunque proporcionan conectividad a larga distancia con alta velocidad y capacidad de transferencia de datos, los costos de implementación y mantenimiento son significativos. Puede haber latencia en la transmisión de datos a larga distancia, y requieren un nivel más alto de configuración y mantenimiento.

Ventajas: Proporciona conectividad a larga distancia. Alta velocidad y capacidad de transferencia de datos.	Inconvenientes: Costos de implementación y mantenimiento significativos. Posible latencia en la transmisión de datos a larga distancia. Requiere un nivel más alto de configuración y mantenimiento.
--	--

2. Describa detalladamente qué es la directiva `#ifndef`, qué uso tiene e ilustre su exposición con un ejemplo representativo.

La directiva `#ifndef <Type_Definition>` se usa para saber si no está definida, definición la con `#define <Type_Definition>`, suele usarse junto con el `#endif` para indicar que acabó la indirectiva, también de puede usar `#pragma once`, pero no se usará en IB. es bastante útil en programas muy largos o que sin querer hayas añadido dos veces la misma librería local o de programa, pues así te evitas un error de compilación de tipo redefinición.

<code>#ifndef EMEPLO_H #define EMEPLO_H #include <iostream> //resto de cabeceras necesarias // Funcio o clase de EJEMPLO #endif</code>	<code>#include "EJEMPLO.h" #include "EJEMPLO.h" // Otras cabeceras que no esten en el fichero de EJEMPLO.h // código de desarrollo del .h</code>	En esta ejemplo no se produciría ningún error de redefinición, pues tenemos la directiva <code>#ifndef</code> (if not define) por lo que solo se haría caso al primer <code>#include</code> el segundo se ignoraba pues ya está definido.
<code>#include <iostream> //resto de cabeceras necesarias // Funcio o clase de EJEMPLO</code>	<code>#include "EJEMPLO.h" #include "EJEMPLO.h" // código de desarrollo del .h</code>	En esta ejemplo se produciría un error de redefinición, pues no tenemos la directiva <code>#ifndef</code> (if not define) por lo que solo se haría caso al primer <code>#include</code> el segundo volvería a definido. generando así este error.

5. Responda Verdadero (V) o Falso (F) o elegir la/s respuesta/s correcta/s para cada una de las siguientes preguntas teniendo en cuenta las siguientes observaciones:

- Por cada respuesta correcta se sumará 0.2 puntos.
- Por cada respuesta incorrecta se restará 0.2 puntos.
- Las preguntas sin responder se considerarán incorrectas.
- La puntuación mínima de la pregunta es de 0 puntos.

- 1.- El directorio /bin/ de un sistema Linux contiene ficheros de configuración del sistema **Falso**
- 2.- De acuerdo a la guía de estilo de programación de la asignatura, los identificadores de atributos de clases (datos miembros) se escriben igual que los de cualquier otra variable **Verdadero**
- 3.- La siguiente declaración es correcta: `boolean IsOdd(int x);` **Verdadero**
- 4.- A un `std::vector` no se le puede cambiar el número de elementos que almacena **Falso**
- 5.- El valor que el siguiente programa imprime en pantalla es 00011011 **Verdadero**

```
#include <iostream>
int main() {
    int i{0};
    while (i < 2) {
        for (int j{0}; j < 2; ++j) {
            std::cout << i << j;
        }
        ++i;
    }
    return 0;
}
```

- 6.- Las clases correctamente encapsuladas deben utilizar el nivel de acceso "protected" **Falso**
- 7.- Cualquier operador puede ser sobrecargado mediante una función friend **Falso**
- 8.- En una agregación las partes de la misma pueden pertenecer a más de un objeto simultáneamente **Verdadero**
- 9.- Las direcciones IP versión IPv6 utilizan 256 bits **Falso**
- 10.- El lenguaje SQL se utiliza para para realizar operaciones CRUD en bases de datos no relacionales **Verdadero**

1. Bases de datos no relacionales. Explique (no más de un folio escrito por ambas caras) (a) Qué son este tipo de bases y cuales son sus características definitorias. (b) Exponga cuáles son los tipos de bases de datos no relacionales más habituales.

Bases de datos no relacionales (también conocidas como bases de datos NoSQL) se diferencian de las bases de datos relacionales tradicionales en que almacenan sus datos en forma no tabular, es decir no utiliza el esquema tabular de filas y columnas. A menudo se utilizan cuando es necesario organizar grandes cantidades de datos complejos y diversos. Su modelo de almacenamiento está optimizado para el tipo de datos que almacena:

Bases de datos orientadas a documentos: Los datos se guardan en colecciones de documentos, normalmente utilizando los formatos JSON, XML o BSON. Un registro puede contener tantos datos como sea necesario, en cualquier tipo (o tipos) de datos; no hay restricciones Hay una estructura interna dentro de un solo documento. Sin embargo, puede diferir de un documento a otro.

Bases de datos de gráficos: Muestran conexiones entre diferentes puntos de datos. Se utilizan principalmente cuando es necesario analizar diferentes tipos de datos y su relación entre sí. Estos se representan en forma de una red de objetos o nodos que están relacionados

Bases de datos de valores clave: Este tipo solo almacena y proporciona conocimiento rápido y sencillo sobre pares clave-valor. Esta es una forma sencilla y fácil de almacenar y acceder a los datos.

Bases de datos de columna ancha: Almacena y gestiona grandes cantidades de datos en tablas o múltiples columnas, cada columna de las cuales puede actuar como un registro.

2. Describa detalladamente qué es la sobrecarga de funciones, qué uso tiene e ilustre su exposición con un ejemplo representativo.

En C++, la sobrecarga de funciones se refiere a la posibilidad de definir múltiples funciones con el mismo nombre en el mismo ámbito, pero con diferentes conjuntos de parámetros. Esto permite a un programador utilizar el mismo nombre de función para operaciones similares que actúan sobre diferentes tipos de datos o con diferentes números y tipos de parámetros.

La sobrecarga de funciones facilita la escritura de código más intuitivo y expresivo, ya que permite utilizar el mismo nombre de función para realizar acciones similares en diferentes contextos. La elección de la función específica a ejecutar se basa en los parámetros proporcionados al llamar a la función.

```
// Overload as normal (global) function
Complex operator+(const Complex& complex, const double real_number) {
    return Complex(complex.real() + real_number, complex.imag());
}

// Overload as member function (method)
Complex Complex::operator+(const double real_number) {
    return Complex(real() + real_number, imag());
}

// This can only be accomplished with global function
Complex operator+(const double real_number, const Complex& complex) {
    return Complex(real_number + complex.real(), complex.imag());
}
```

5. Responda en este mismo folio Verdadero (V) o Falso (F) para cada una de las siguientes preguntas teniendo en cuenta las siguientes observaciones:

- Por cada respuesta correcta se sumará 0.2 puntos.
- Por cada respuesta incorrecta se restará 0.2 puntos.
- Las preguntas sin responder se considerarán incorrectas.
- La puntuación mínima de la pregunta es de 0 puntos.

1.- El comando de Linux que permite mostrar en pantalla su nombre de usuario es uname **Verdadero**

2.- El preprocesador de C++ elimina del código los comentarios que el usuario puede haber escrito **Falso**

3.- De acuerdo a la Guía de Estilo de Google para C++ los identificadores de funciones ordinarias (no métodos de una clase) deben escribirse comenzando con letra mayúscula y usando mayúscula para cada palabra. **Falso**

4.- La salida que el siguiente programa imprime en pantalla es: 2 11 **Falso**

```
#include <iostream>
void Sum(int value_a, int value_b, int& value_c) {
    value_a = value_b + value_c;
    value_b = value_a + value_c;
    value_c = value_a + value_b;
}
int main() {
    int first{2}, second{3};
    Sum(first, second, second);
    std::cout << first << " " << second;
    return 0;
}
```

5.- El método push_back() permite añadir elementos a un objeto de la clase std::vector **Verdadero**

6.- Al leer caracteres desde un flujo de entrada (cin), la lectura se detiene al detectar un punto **Falso**

7.- Los atributos de una clase sólo pueden ser públicos o privados **Falso**

8.- Cualquier operador puede ser sobrecargado mediante una función friend **Falso**

9.- Una función se puede declarar friend de más de una clase. **Verdadero**

10.- En una relación de composición (entre objetos de diferentes clases), la parte (miembro) desconoce la existencia del objeto **Verdadero**