

1. Programación multi-procesador o multi-core con Python o Javascript.

Si es ejecución sincrónica o asincrónica y asociar con teoría vista.

La programación multi-procesador o multi-core se refiere al uso de múltiples procesadores o núcleos de procesamiento en un sistema para ejecutar tareas de forma simultánea. Tanto Python como JavaScript admiten la programación multi-procesador o multi-core a través de bibliotecas específicas.

La ejecución sincrónica implica que las tareas se ejecutan en secuencia, una después de la otra, mientras que la ejecución asincrónica permite que las tareas se ejecuten de forma independiente y en paralelo. En Python, una biblioteca común para la programación multi-procesador o multi-core es `multiprocessing`. Permite crear procesos múltiples que se ejecutan en paralelo y aprovechan los núcleos de procesamiento disponibles.

Mostrar ejemplo práctico.

Ejemplo en PYTHON

```
python Copy code

import multiprocessing

def square(n):
    return n ** 2

if __name__ == '__main__':
    numbers = [1, 2, 3, 4, 5]

    # Crear un grupo de procesos
    pool = multiprocessing.Pool()

    # Aplicar la función a los números en paralelo
    results = pool.map(square, numbers)

    # Imprimir los resultados
    print(results)
```

2. Cómo implementar un clúster de computadoras en Windows o Linux.

Qué tecnologías y protocolos subyacentes se utilizan.

Para implementar un clúster se utilizan diversas tecnologías y protocolos subyacentes.

En Windows, puedes usar Windows Server, mientras que en Linux, puedes optar por distribuciones como Ubuntu Server, CentOS o Red Hat Enterprise Linux.

En el caso de hardware se necesita un conjunto de computadoras interconectadas para formar el clúster. Estas computadoras, conocidas como nodos del clúster, pueden ser servidores dedicados o máquinas virtuales que se ejecutan en un hipervisor.

Además, en la red se podría utilizar los nodos del clúster se comuniquen entre sí, se utiliza una red de área local (LAN) o una red de área amplia (WAN) según los requisitos. Los protocolos de red estándar, como TCP/IP, se utilizan para la comunicación entre los nodos.

Es importante tener en cuenta que la implementación de un clúster puede variar según los requisitos específicos y las tecnologías utilizadas. Los detalles exactos de la configuración y la administración del clúster dependerán del sistema operativo y las herramientas seleccionadas.

3. Cómo diseñaría un clúster de computadoras en Windows o Linux.

Qué tecnologías, protocolos o lenguajes utilizaría.

Se debe realizarse considerando varios aspectos, como los requisitos de rendimiento, la disponibilidad, la escalabilidad y la confiabilidad.

Antes de diseñar un clúster, es importante comprender los objetivos y requisitos del sistema. Esto incluye considerar aspectos como la carga de trabajo esperada, la disponibilidad requerida, las necesidades de almacenamiento compartido y la escalabilidad futura.

También elegir el sistema operativo adecuado para el clúster. Tanto Windows Server como varias distribuciones de Linux (como Ubuntu Server, CentOS o Red Hat Enterprise Linux) son opciones comunes.

En la parte del hardware se debe elegir el adecuado para los nodos del clúster. Esto puede incluir servidores dedicados o máquinas virtuales que se ejecutan en un hipervisor. Considera aspectos como el rendimiento, la capacidad de almacenamiento, la capacidad de red y la redundancia.

Para la parte de configurar una red adecuada para el clúster. Utiliza protocolos estándar como TCP/IP para la comunicación entre los nodos. Puedes utilizar tecnologías como Ethernet, InfiniBand o Fibre Channel, según tus necesidades de rendimiento y latencia.

Para la administración se debe utilizar herramientas de administración adecuadas para configurar y administrar el clúster. En Windows, puedes utilizar la herramienta Failover Cluster Manager. En Linux, puedes utilizar herramientas como PCS, DRBD o herramientas específicas proporcionadas por la distribución elegida.