Embedded Linux:

* Toolchain
* Deployment on embedded device.

C/C++:

* C vs CPP: power of CPP over C?
* Try… Catch, Exception handling, multiple catch
* Type of allocated memory: Variable declared in main is allocated in which memory?
* Pass by value/ref.
* Copy Constructor, Assignment operator
* Write a Copy constructor to copy string
* Which are the Operators that cannot be overloaded? \*, ., ::
* Deep copy, Shallow Copy
* Do you know Sequence Initialization? When it is used? (Const Member, Dynamic Member)
* New vs Malloc? (New is function and can be overloaded, malloc is keyword) Can new be overloaded?
* Function Pointer
* DLL
* Storage classes
* Const ptr to const
* Array vs LL vs Vector
* Hashtable
* LL
* Binary Tree, Binary search

OOP in CPP:

* Abstraction, Encapsulation
* Inheritance : Type of inheritance, access specifier in inheritance, protected inheritance
* Diamond Problem
* Can we have Virtual Constructor?
* Can we have Virtual Destructor? What is use of it?
* STL
* Polymorphism: Static Polymorphism, dynamic
* Overloading
* Overriding
* Unary Vs Binary Operator Overloading
* VTable,
* Abstract Class
* Abstract class vs Private constructors

System/Application Development:

* Thread vs Process
* What are the thread synchronization methods?
* What is event, mutex and semaphores?
* Which are the IPC mechanisms?
* Design Patterns: Singleton, factory patterns, factory methods
* TCP, UDP example in Linux utilities?
* Which protocol Ping works on?
* What is Memory Leak, How to prevent/detect?
* How to optimize performance of code? How do you approach to find and fix any random crash?

Qt:

* Signal Slot, 2 connects
* Custom Control
* Embedded Qt: Keyboard, Input Methods
* QML
* Qt Controls,
* QTreeView
* QApplication, QCoreApplication, QUIApplication
* Qt.conf, plugins
* Qt Embedded Vs Desktop
* Deployment of Qt on device
* Library dependency handling in Qt.
* QThread, QProcess
* QtConcurrency

Current Project Architecture