Q.1.1.

a) 11731

b) 5115

c) 7936

Q.1.2.

RAID 1: Financial records/Databases.

RAID 5: Large media files, documents(non-critical)

Q2.1. a)

Unified address space: Since device registers are mapped into the system's memory address space, the CPU can access them using the same instructions it uses for regular memory access (like 'load' and 'store' instructions)

Increased Speed: Faster access through caching.

More Flexible and Scalable: No limit on addressable devices.

Compatibility with Modern Software Architectures: Easier to integrate with operating systems.

Q2.1. b)

Consumes Address Space: Limited memory address space.

Possible Performance Degradation: No caching for I/O operations.

Complexity in Memory Management: Requires careful management of memory regions.

Security Risks: Potential for accidental overwriting.

Q2.2.

Direct blocks size=12 blocks×8 KB/block=96 KB

Single indirect size=2048 pointers×8 KB/block=16384 KB=16 MB

Double indirect size=4194304 blocks×8 KB/block=33554432 KB=32 GB

Triple indirect size=8589934592 blocks×8 KB/block=68719476768 KB=64 GB

= 96.015715 GB

Q2.3.

Logging ensures that updates are atomic, meaning that they either complete fully or not at all.

If the file system crashes, the log can be replayed to bring the file system back to a consistent state.

Q.3.1.

Automatically mounting the root file system at boot time is essential because it provides the necessary foundation for the operating system to load and operate. It ensures access to critical system files, allows for proper initialization of other file systems.

Q3.2.

Pros:

Can provide an additional layer of security.

Security policies related to file access can be managed centrally in one place.

Cons:

Each file access request involves invoking the watchdog program, which introduces additional processing time.

Adds complexity to the system's security infrastructure.

Q.4.

Strengths:

Capabilities allow for quick, direct access checks, as they are tokens given to domains representing rights to objects.

Capabilities can precisely define the type of access (e.g., read, write, execute) a domain has over each object.

Weaknesses:

Capabilities can be transferred or copied between domains. This can lead to unauthorized access.

Once capabilities are distributed, revoking them becomes challenging.