Solution 1 - Unix fsck (File System Checker)

When system boot, check system looking for inconsistencies
e.g. inode pointers and bitmaps, directory entries and inode reference counts

➡ Try to fix errors automatically

◉ Cannot fix all crash scenarios

◉ Poor performance

- Sometimes takes hours to run on large disk volumes
- Does fsck have to run upon every reboot?

◉ Not well-defined consistency

Solution 2 - Log Structure File System (LFS) or (Copy-On-Write Logging)

Idea - treat disk like a tape-drive

1. Buffer all data (including inode) in memory segment

2. Write buffered data to new segment on disk in a sequential log

➡ Existing data is not overwritten
Segment is always written in free location

✓ Best performance from disk for sequential access

				b[0]:A0 b[1]:A1 b[2]:A2 b[3]:A3	$D_{k,0}$	b[0]:A5	
$D_{j,0}$	$D_{j,1}$	$D_{j,2}$	$D_{j,3}$				
A0	A1	A2	A3	Inode j A5		Inode k	