What does a fully associative cache store?	How does the CPU locate an item in a fully associative cache?
$What \ is \ temporal \ locality?$	What is spatial locality? 4
What are the three common cache replacement algorithms?	Explain the write-through cache write strategy.
Explain the copy-back cache write strategy.	Why does a direct mapped cache usually use static RAM ?

Hardware compares the input address with all stored addresses (in parallel) Addresses and their corresponding data If we get a match we have a hit If no match we must go to main memory 1 The principle that if you use an address once, you are The principle that if you use an address once, you also likely to use addresses nearby e.g. arrays may use it again soon e.g. loops Least Recently Used (LRU) Whenever a write is done to the cache, the write is Round Robin also done to main memory Random6 5

It is a lot faster than dynamic RAM

When a cache line is replaced, if the dirty bit is set,

the modified value is written to main memory

How many transistors do DRAM and SRAM use per bit?	Briefly explain what a set associative cache consists of.
What is the advantage of using a set associative cache?	What two control bits are usually used in cache entries?
Explain what a compulsory cache miss is?	Explain what a capacity cache miss is?
Explain what a conflict cache miss is?	What are the two types of virtualisation?

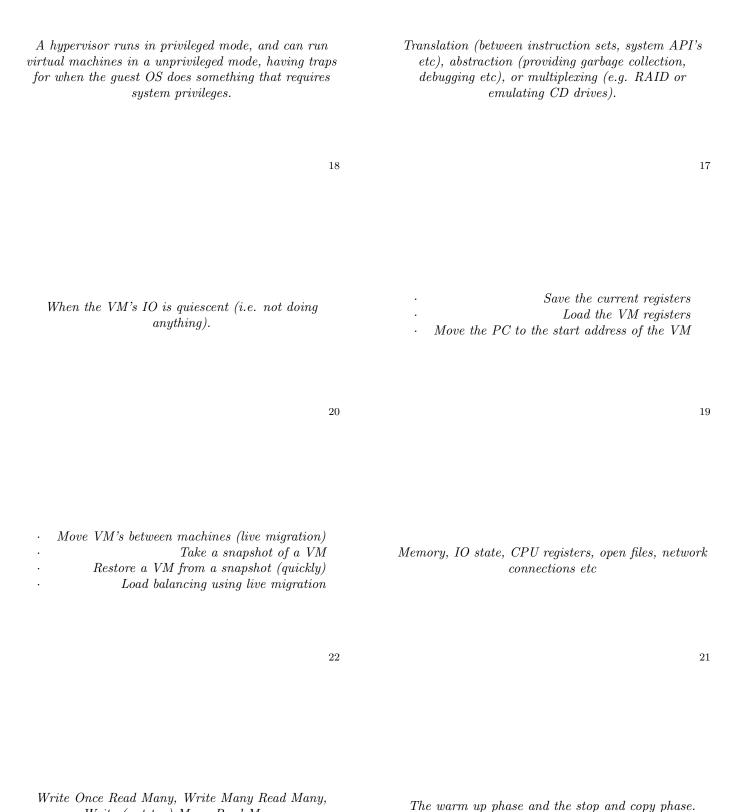


process under a control layer of software, e.g. JVM).

the cache. If the cache was fully associative, then

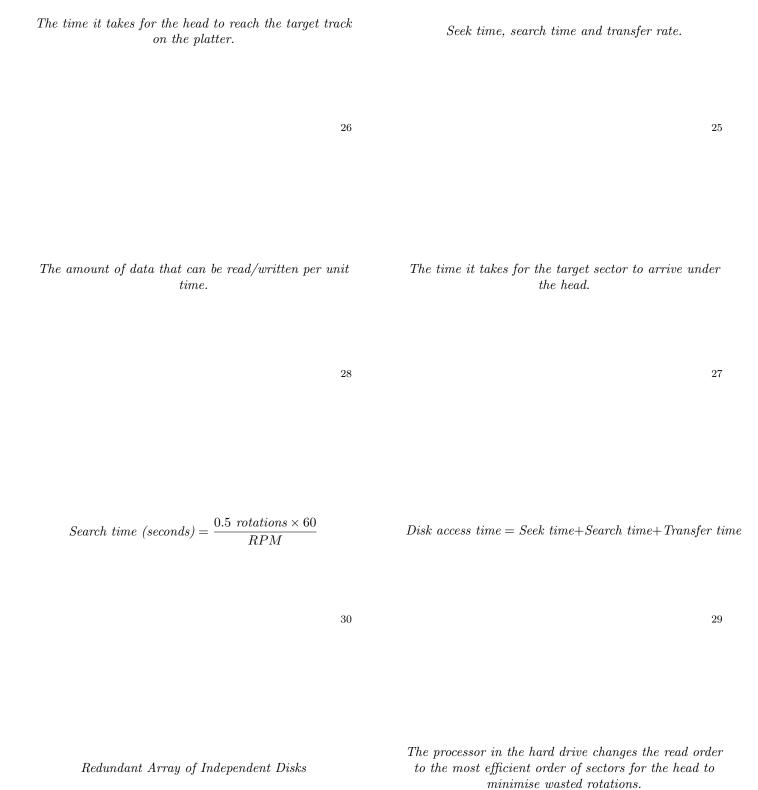
misses due to this wouldn't occur.

What are the three main advantages of virtualisation?	A hypervisor runs in mode, and can run virtual machines in a mode, having for when the guest OS does something that requires
What happens when you start a VM?	When is it best to stop a VM?
What is retained when a VM is stopped/paused? 21	What operations can we do on a VM?
What are the two phases in live migration?	What are the three main categories of permanent storage media?

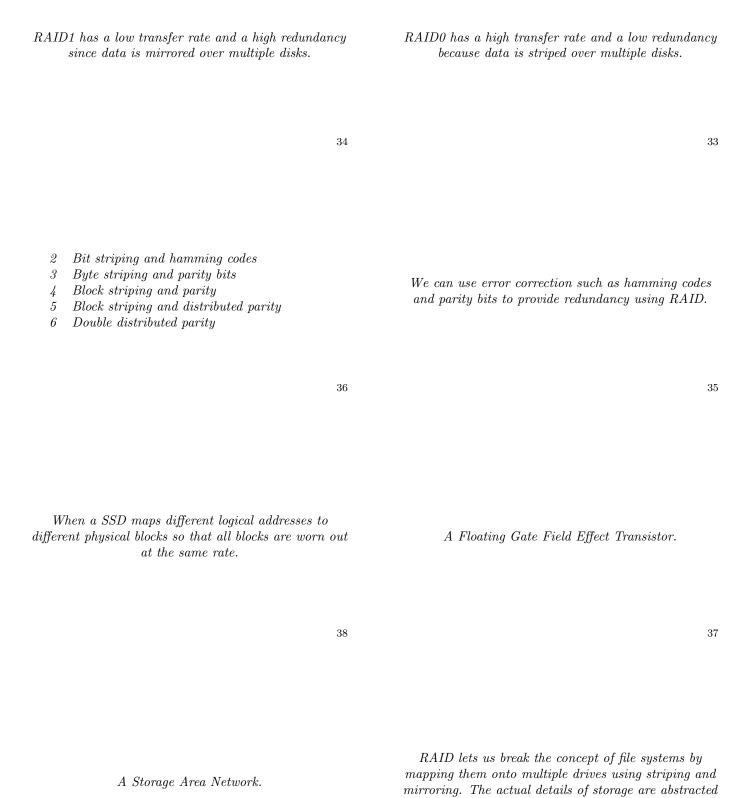


Write (not too) Many Read Many

What are the three terms used to quantify hard drive performance characteristics?	What is the seek time?
What is the search time?	What is the transfer rate?
Give the equation for disk access time.	How can we work out the search time from the RPM?
What happens if the OS wants to read a file that is split over multiple sectors on the hard drive?	What does RAID stand for? 32



RAID0 has a transfer rate and a redundancy because data is over multiple disks.	RAID1 has a transfer rate and a redundancy since data is over multiple disks.
We can use such as and to provide redundancy using RAID.	Define $RAID2,3,4,5,6$.
What is the transistor used in SSD's?	What is wear-levelling?
RAID lets us break the concept of by mapping them onto using The actual details of storage are	What is a SAN ?



away.

is a volume aware filesystem. It protects against losing files, running out of space, corruption of data etc by being very flexible and having lots of and implementing, simple , self and, sumchecking and more. 41	Explain each stage of a 5 stage pipeline	2
Briefly explain what pipelining is	$What \ is \ a \ control \ hazard?$	4
What are two ways of dealing with control hazards?	Briefly explain what branch prediction is	6
What is used to implement branch prediction and what does it do?	What is a data hazard?	8

IF - Fetch instruction from memory
ID - Decode instruction; select registers
EX - Perform an operation or calculate an address
MEM - Access an operand in memory
WB - Write to registers

ZFS is a volume aware filesystem. It protects against losing files, running out of space, corruption of data etc by being very flexible and having lots of ECC and implementing copy-on-write, simple rollback and recovery, wear leveling, self checking and healing, sumchecking and more.

42 41

If we have a branch at the ID stage, then the fetched instruction at the IF stage will have to be ignored all the way down the pipeline, wasting one full cycle and causing a bubble.

Where we get all the components of the CPU working at the same time, with buffers that are flushed every clock cycle inbetween each stage, so that we can overlap the execution of instructions to increase overall clock speed.

44

If we can remember what address a branch directed us to fetch next from what it did when we executed that branch previously, then we can pre-emptively load that instruction in the IF stage instead of fetching the instruction at the PC.

Pipeline bubbling (abort instructions that are incorrect) and branch prediction (guess what way to branch).

46

This is where we execute instructions that depend on each other in parallel or close together and the correct data might not be in the right place (e.g. registers).

48

A branch target buffer which maps the virtual address of one branch instruction onto the virtual address of the instruction that is branched to

47

What is forwarding?	How can we exploit instruction level parallelism?
What does VLIW stand for?	How can we implement an out of order processor?

Fetch multiple instructions per cycle
Have multiple ALU's to execute instructions in
parallel (superscalar)
Have common registers and caches, since the
instructions are operating on the same data

Where we add extra paths to the architecture to pass updated register values back to previous stages of the pipeline to avoid data hazards.

50 49

Have a buffer that instructions are fetched into A scheduler to choose which instructions to execute at what times

A cache to store memory and register accesses until all instructions have finished so that the application can execute normally as though instructions were executed in parallel Very Long Instruction Word

52