Kafli 1

Instruction cycle is fetch, decode, execute
Cache er small, very fast memory on processor for recently used memory content for quick access
Memory Hierarchy. Fast, small, expensive on top, slow, large, cheap on bottom

Kafli 2

Objectives of OS: Conveniance, efficiency, evolution
Multiprogramming: Processes use the same Processor
Batch processing: One program executed before the next one is started
Processes: A program in execution
Resource management: Fairness, Differential responsiveness, effiency

Kafli 3

Instruction trace : Sequence of instructions executed for a process
Process states : ready, running, block, suspend.
Process control block: Contains information for OS to control processes
PCB Stack = Identification, State, priority, pc, memory pointers, context data, io status, accounting information.
Kernel mode has access to all, user should be restricted to protect the os.

Kafli 4

Process can have one or more threads. Threads are execution units within processes while processes are binaries. Process has one process control block while threads gets it's own space on the stack and control block. Threads share the state and resources of that process

kaflar 11 oog 12

Single buffer - process writes to buffer than has to wait for I/O to write and reverse. Double buffer - Process can write to one buffer and read another, don't have to wait. Circular buffer uses more than 2 buffers.
Blocking i/o - process continues in non blocking, os takes over if it is blocking

Kafli 5

Race condition : "loser" makes the last change, affects all.
Mutual exclusion: only one process can access code or data.
Producer/Consumer
Producer puts data in buffer, consumer takes data from buffer.
If there is no data consumer sleeps and producer wakes him when he adds data.
If buffer is full, producer sleeps and consumer wakes him when he takes data.
Reader/Writer problem
The conditions that must be satisfied are as follows
1.Any number of readers may simultaneously read the file
2.Only one writer at a time may write to the file, no reader may read it
Monitors must be invoked with a call to a monitor procedure. uses wait and signal, if no process is waiting the signal is lost.
Message passing: direct and indirect.

Kafli 6

Deadlocks	
Causes	Mutual exclusion, no preemption, hold and wait, circular wait
Prevention	Eliminate on of the conditions. Direct eliminates circular waiting, indirect the other 3. Very conservative
Avoidance	Dynamic decision if current resource request will cause deadlock. Bankers algo
Detection	Tests for deadlocks, consumes cosiderable cpu time, liberal with resources.
Starvation	Where process is overlooked by scheduler even though it's ready to run.

Kafli 7

Partitioning	
Fixed equal	Program to big, use overlays, inefficient, internal fragmentation
Fixed unequal	Chooses the smalles space it can fit in, might lead to swapping,memory might go unused, internal fragmentation
Dynamic	Each process gets exactly what it needs, small holes, external fragmentation, compact
Best fit	Chooses the block closest in size
first fit	Chooses the next block which can hold the process. scan from beginning
next fit	Starts for current position, finds block next available large enough block.

Kafli 8

Translation lookaside buffer: checks the TLB table and if it's a hit it gets the frame number immediately and can add the offset and find the data it needs else it has to look up the address in the page table.
Page replacement: FIFO, LRU, Optimal, Clock.

kafli 9

Levels of scheduling	Long term, medium term, short term.
Long term	Controls what programs are admitted to the system for processing
medium term	if process starts in swap out it is added to medium term scheduler
Short term	when program becomes a process it is added to the short term scheduler
Preemptive and non preemptive	Preemtive can take resources, non preemptive can not