2

**Spin()**, check time, ret once run for 1 second

**Virtualize CPU** (turning a single or small set) into seemingly many CPUs, allow programs to seemingly run at once

**Virtualize memory**, as if each running program has its own private memory; each process accesses its own private **virtual address space**

**Concurrency**,instructions do not execute atomically, strange things happen

**Persistency**, DRAM store values in a volatile manner, data in memory lost

**Abstractions** makes it possible to write a large program by dividing it into small and understandable pieces

Early OS: just **libraries** of commonly used functions, programs handled by human operator

Syscall transfers control into the OS and raising the **hardware privilege level**: in **kernel mode**, OS has full access to the hardware

4

**Process** = running program

At a time, process can be described by its state

(state = contents of memory in address space, CPU registers, I/O)

**Address space** = the memory that the process can address

**Process API** = calls that programs can make about processes (create, destroy…)

**Process states**: running, ready, blocked

Events: **scheduled**, **descheduled**

**Process list =** info about all processes in system,

each entry = **process control block** = struct of process info

26

**Critical section** is a piece of code that accesses a shared resource, usually

a variable or data structure