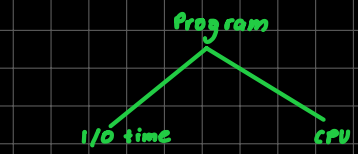
# Lec 1

- uniprogramming & multiprogramming
- batch OS

- multiprocessing — 1 core = 1 cpu
  → multiple processors

- Tightly coupled system
- Distributed OS
- Multicore systems
- Clusterred operating system
-




→ CPU idle

→ Drawbacks of I/O time processes in uniprogramming

→ To solve the issue, multiprogramming was introduced

(responsive) (banking)

→ multitasking & multiprogramming
↳ premptive algos used ↳ non-premptive algo used

- * P2P, master-slave system

# Lec 2

- Von Neumann Architecture
- Program Counter
- function calling in pipelining using stacks
- All Alevels registers
- memory structure in hierarchy
  → Cache
  → speed
  → size
  → votality

- memory size ↑
  access ↓
  contradictory

- * cache sizes of same architecture

- HDD is an I/O device

Cache
RAM ↑ access time decreese, storage decrease
primary memory

- Cache coherency / Data consistency
- caching
- I/O structure
  → Role of DMA
- DMA when reading, and CPU schedules other processes?
  → cycle steeling*
- with and without DMA execution

- synchronous communication → CPU waits for I/O to complete
- Asynchronous → CPU runs other tasks simultaneously

- Interrupts
  → Traps, Software Interrupt, Hardware Interrupt

# Lec 3

- Interrupts
  - Software
  - Hardware

- Traps → infinitely running programs
- Hardware → Ethernet, keystrokes
- ISR
- printf() → write system call
- Interrupt Timeline
  → minute dip initially for software interrupt , —√—
- locking & unlocking events
- services provided by OS.
- Multisharing / Multiprogram
- OS operations → mode bit
- transition from user mode to kernel mode // seperate areas
- I/O subsystem
  → spooling : simultaneous peripheral operations
- Protection & security