CS 537: CPU Virtualization

1) Mechanisms : How

2) Policies : Scheduling (choosing which process to run)

Mechanisms: ("at once")

⇒ Run N processes, while OS maintains control (security, protection, robustness) + efficiently

key technique:

$$\underbrace{\text{Limited}}_{\text{protection}}\ \underbrace{\text{Direct Execution}}_{\text{efficiency}}$$

Steps

@ Boot (Start up)

OS runs first, establish control of machine

=> sets up trap handlers

(code in OS to handle various service requests)

e.g. read a file

app issues a "system call"

<-> issue a trap (special inst)

key thing on trap: (App -> OS)

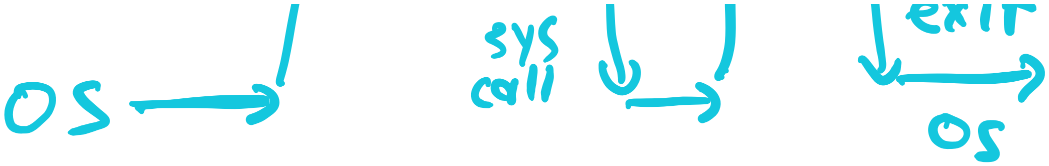
switch modes

(from user -> kernel)

non-privileged   privileged

when done: return-from-trap

Timeline



How to run many processes "at once"?

P_A .... P_B .... P_A .... P_B

"time sharing"

How? Problem:

A Process may run for a long time

Q: how to OS regain control of CPU?

A: timer interrupt

@ boot: set this up

interrupt the CPU every X milliseconds

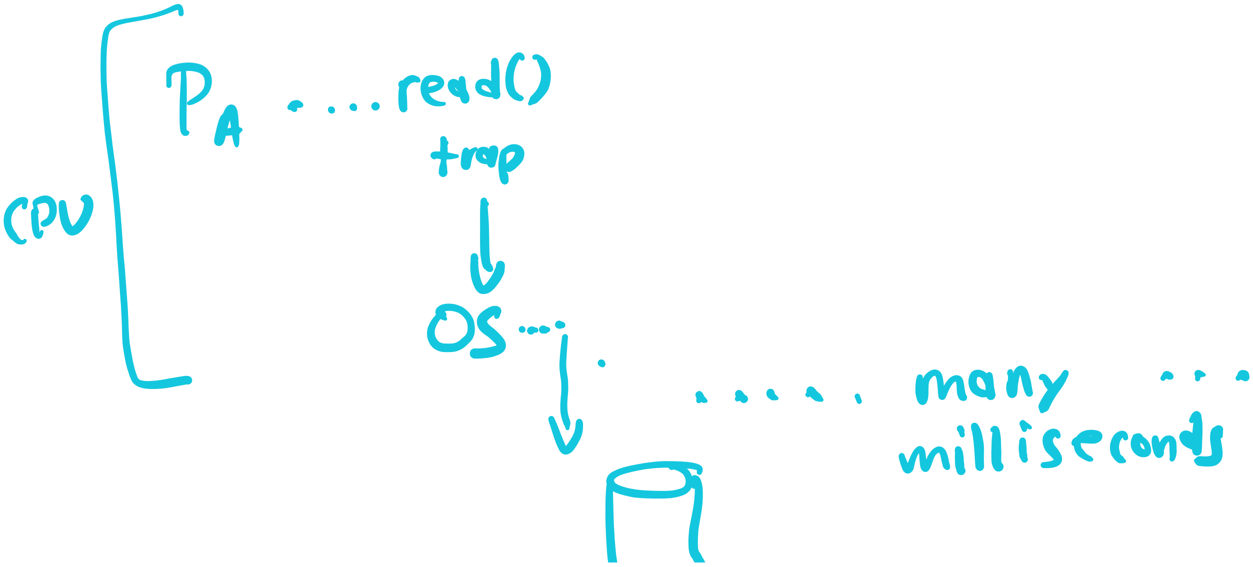
int 1)(save register state of P_A)
↓ .... 2)(restore state of P_B)
OS

"context switch"

OS responsibility:
track for each process: state

→ Running
→ Ready (not running, could be)

what if Process does something "slow"?
(disk I/O)



desire:

switch to some other process

when I/O is issued:

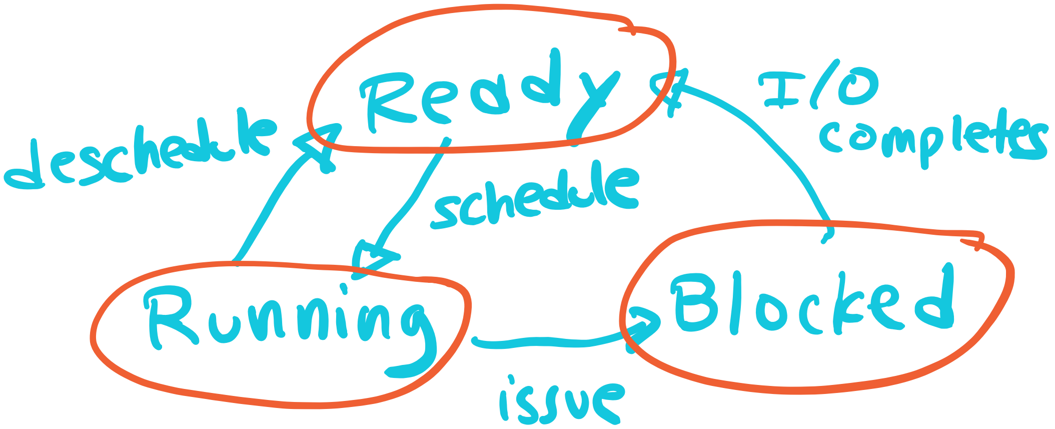
→ mark P_A as <u>Blocked</u>
(waiting for I/O)

→ switch to P_B

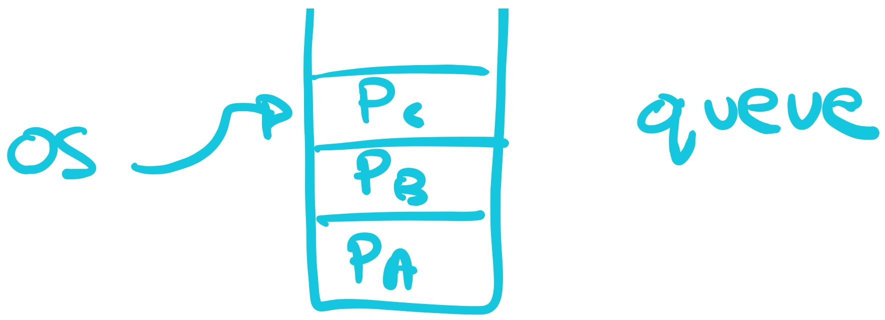
⋮

I/O completes

→ mark P_A as <u>Ready</u>



OS ⇒ CPU
Scheduler:

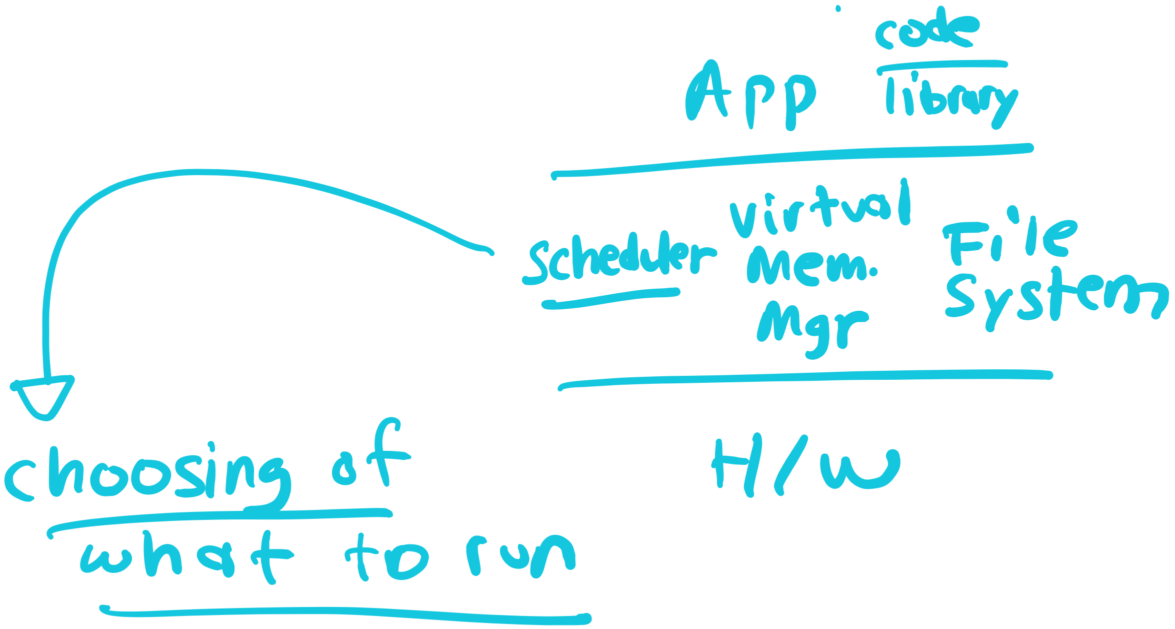


Back to work!

=> ~~Discussion~~ thanks

=> Piazza

=> P1b : light intro to xv6



Sched. Policy : how ?

simplifying
Assumptions about "workload" (set of processes that OS needs to run)

process ⟺ job

Assumptions:

=> start: all jobs arrive @ once

=> just use CPU (no I/O)

=> all fixed length (runs for time T)

=> time is known

=> metric: turnaround time

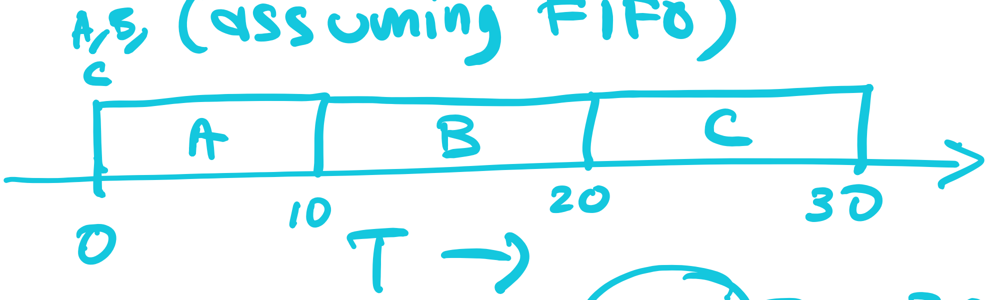
$$T = T_{completes} - T_{arrives}$$

Algorithm: #1 : FIFO, FCFS

A, B, C arrive @ t=0
run time: 10

Avg Turnaround:

(assuming FIFO)

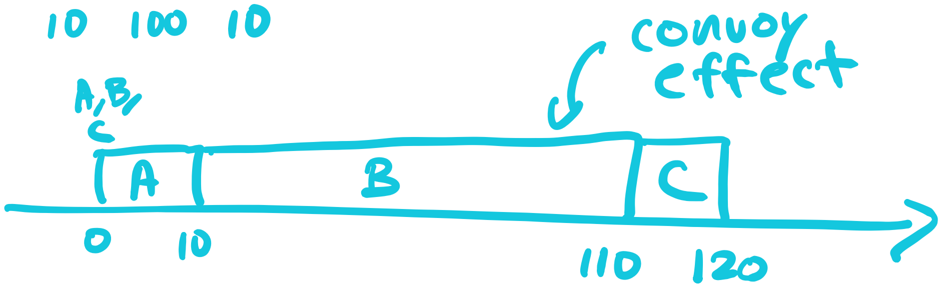


$T_A = 10$ $T_B = 20$ $T_C = 30$

Relax Assumption #3:

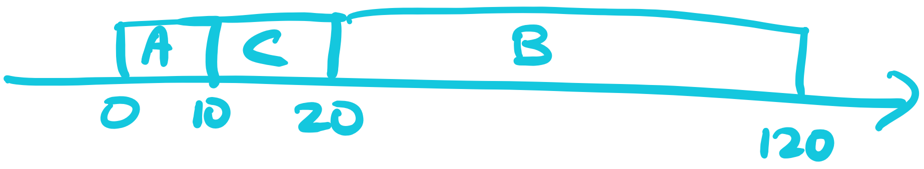
A, B, C

10 100 10



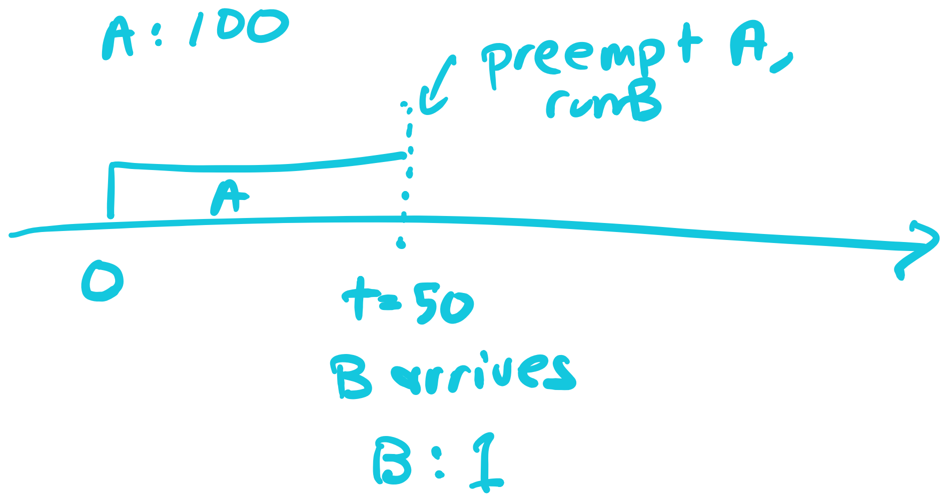
$T_{Avg} = 80$

New: SJF
shortest job first



$T_{AVG} = 50$

All jobs don't arrive at once



Shortest Time to Completion First (STCF)